Probabilistic Dependency Graphs and Inconsistency

How to model, measure, and mitigate internal conflict

Oliver Richardson

Cornell University
Department of Computer Science

September 2021

Outline for Section 1

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

The standard way of modeling an agent with uncertainty:

The standard way of modeling an agent with uncertainty:

- a probability distribution $p : \Delta W$ over worlds W,

$$\xrightarrow{p} \boxed{W}$$

The standard way of modeling an agent with uncertainty:

- a probability distribution $p : \Delta W$ over worlds W,
- a utility function $u : \Omega \to \mathbb{R}$

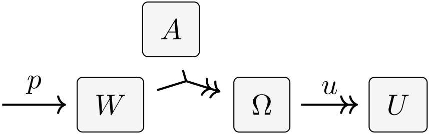
A utility function is *not* part of the standard way of modeling an agent with uncertainty. I would cut utility. It's a distraction.

$$\xrightarrow{p} \boxed{W} \qquad \boxed{\Omega} \overset{u}{\twoheadrightarrow} \boxed{U}$$

Don't overwhelm the reader with notation. I would cut this.

The standard way of modeling an agent with uncertainty:

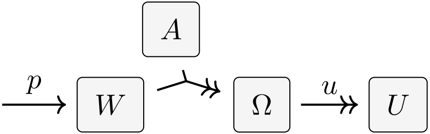
- a probability distribution $p : \Delta W$ over worlds W,
- a utility function $u : \Omega \to \mathbb{R}$
- some actions A.



The standard way of modeling an agent with uncertainty:

- a probability distribution $p : \Delta W$ over worlds W,
- a utility function $u : \Omega \to \mathbb{R}$
- some actions A.

cut acdtions; it's a distraction



Such agents cannot have internal conflict;

by construction, they have consistent beliefs and desires.

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?
  - Assertions and test cases can fail.

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?
  - Assertions and test cases can fail.
- Why adopt beliefs without first cross-checking all knowledge?

We are *not* building a system that can be inconsistent.
I would cut all this. It's a distraction.

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?
  - Assertions and test cases can fail.
- Why adopt beliefs without first cross-checking all knowledge?
- Why consult multiple models, that can give inconsistent answers?

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?
  - Assertions and test cases can fail.
- Why adopt beliefs without first cross-checking all knowledge?
- Why consult multiple models, that can give inconsistent answers?

> *A man with a watch knows what time it is.*
> *A man with two watches is never sure.*
> – *Segal's Law*

Why Inconsistency?

Why **model** inconsistency?

- Sometimes people have inconsistent beliefs;
- also want to understand the process of resolving it.

Why **build a system** that can be inconsistent?

- Why entertain the possibility of being wrong?
  - Assertions and test cases can fail.
- Why adopt beliefs without first cross-checking all knowledge?
- Why consult multiple models, that can give inconsistent answers?

> *A man with a watch knows what time it is.*
> *A man with two watches is never sure.*
> *– Segal's Law*

I would keep this and the first three lines, and cut the rest.

Freedom from perfect consistency is valuable,
but demands that you also recognize and address internal conflict.

Yet Another Probabilistic Graphical Model

We introduce *probabilistic dependency graphs* (PDGs), a new class of graphical models for representing uncertainty.

- Designed to tolerate inconsistency, so we can model it.

I don't know what it means to "tolerate" inconsistency in this context. Why not just say that we can model inconsistent beliefs

Yet Another Probabilistic Graphical Model

We introduce *probabilistic dependency graphs* (PDGs), a new class of graphical models for representing uncertainty.

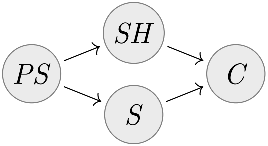
- Designed to tolerate inconsistency, so we can model it.
- In doing so, we get *much* more ...

Two aspects of Bayesian Networks (BNs)

Qualitative BN, $\mathcal{G}$

an independence relation on variables

- $X \perp\!\!\!\perp_{\mathcal{G}} Y \mid \mathbf{Pa}(X)$, for all non-descendents Y of X



Two aspects of Bayesian Networks (BNs)

Qualitative BN, $\mathcal{G}$

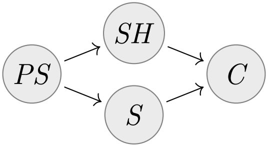
an independence relation on variables

- $X \perp\!\!\!\perp_{\mathcal{G}} Y \mid \mathbf{Pa}(X)$, for all non-descendents Y of X

(Quantitative) BN, $\mathcal{B} = (\mathcal{G}, \mathbf{p})$

a qualitative BN ($\mathcal{G}$) and a cpd $p_X(X \mid \mathbf{Pa}(X))$ for each variable X.

- Defines a joint distribution $\Pr_{\mathcal{B}}$ with the independencies $\perp\!\!\!\perp_{\mathcal{G}}$.



Two aspects of Bayesian Networks (BNs)

Qualitative BN, $\mathcal{G}$

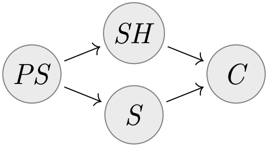
an independence relation on variables

- $X \perp\!\!\!\perp_{\mathcal{G}} Y \mid \mathbf{Pa}(X)$, for all non-descendents Y of X

(Quantitative) BN, $\mathcal{B} = (\mathcal{G}, \mathbf{p})$

a qualitative BN ($\mathcal{G}$) and a cpd $p_X(X \mid \mathbf{Pa}(X))$ for each variable X.

- Defines a joint distribution $\Pr_{\mathcal{B}}$ with the independencies $\perp\!\!\!\perp_{\mathcal{G}}$.



Outline for Section 2

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Simple Example: Floomps and Guns

Grok thinks it likely (.95) that guns are illegal,
but that floomps (local slang) are legal (.90).

Simple Example: Floomps and Guns

Grok thinks it likely (.95) that guns are illegal,
but that floomps (local slang) are legal (.90).

BN

$$\begin{array}{cc} f & \overline{f} \\ \hline \end{array}\quad\begin{array}{cc} g & \overline{g} \end{array}$$

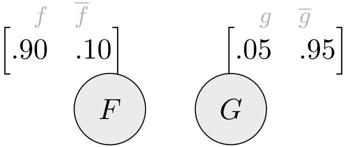
$$\begin{bmatrix} .90 & .10 \end{bmatrix} \qquad \begin{bmatrix} .05 & .95 \end{bmatrix}$$

$$F \qquad G$$

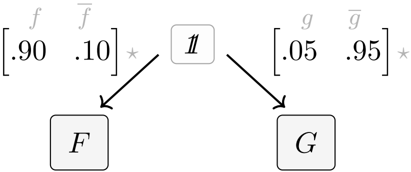
Simple Example: Floomps and Guns

Grok thinks it likely (.95) that guns are illegal,
but that floomps (local slang) are legal (.90).

BN



PDG

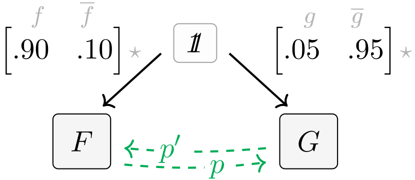


- The cpds of a PDG are attached to edges, not nodes.

Simple Example: Floomps and Guns

BN

PDG

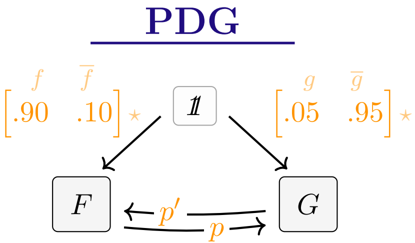


- The cpds of a PDG are attached to edges, not nodes.
- PDGs can incorporate arbitrary new probabilistic information.

Grok learns that Floomps and Guns have the same legal status (92%)

$$p(G\,|\,F) = \begin{array}{c c} \begin{array}{cc} g & \bar{g} \end{array} & \\ \begin{bmatrix} .92 & .08 \\ .08 & .92 \end{bmatrix} & \begin{array}{c} f \\ \bar{f} \end{array} \end{array} = \big(\, p'(F\,|\,G) \,\big)^{\mathsf{T}}$$

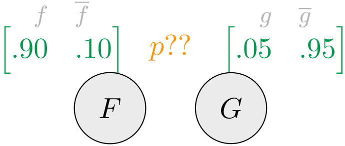
Simple Example: Floomps and Guns



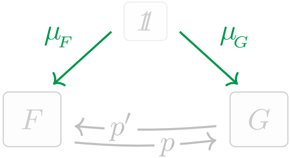
- The cpds of a PDG are attached to edges, not nodes.
- PDGs can incorporate arbitrary new probabilistic information.
- PDGs can be inconsistent

Simple Example: Floomps and Guns

BN



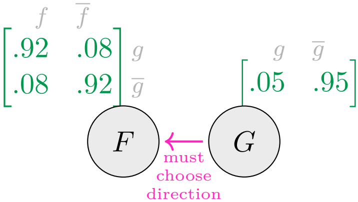
PDG



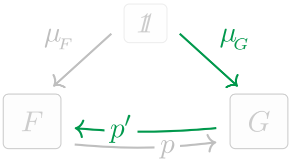
- The cpds of a PDG are attached to edges, not nodes.
- PDGs can incorporate arbitrary new probabilistic information.
- PDGs can be inconsistent,
  - . . . but BNs must resolve inconsistency first, which may break symmetry and irrecoverably lose information.

Simple Example: Floomps and Guns

BN



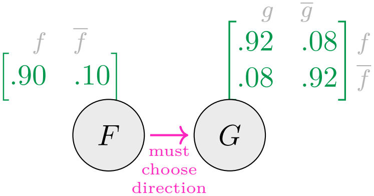
PDG



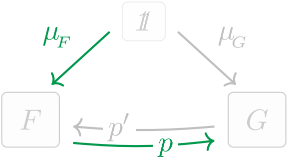
- The cpds of a PDG are attached to edges, not nodes.
- PDGs can incorporate arbitrary new probabilistic information.
- PDGs can be inconsistent,
  - ...but BNs must resolve inconsistency first,
    which may break symmetry and irrecoverably lose information.

Simple Example: Floomps and Guns

BN

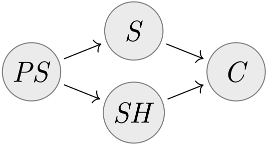


PDG

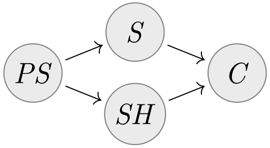


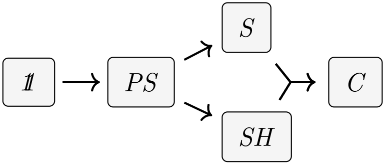
- The cpds of a PDG are attached to edges, not nodes.
- PDGs can incorporate arbitrary new probabilistic information.
- PDGs can be inconsistent,
  - . . . but BNs must resolve inconsistency first,
    which may break symmetry and irrecoverably lose information.

Bayesian Networks as PDGs

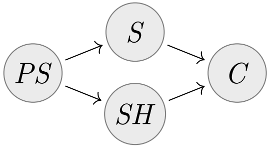


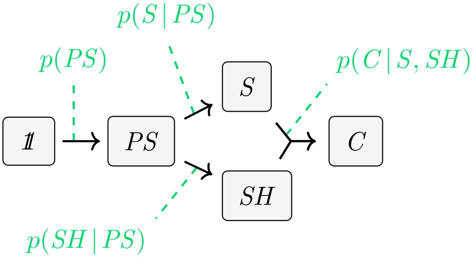
Bayesian Networks as PDGs





Bayesian Networks as PDGs

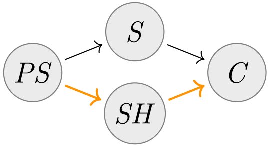


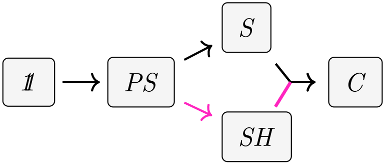


In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;

Bayesian Networks as PDGs

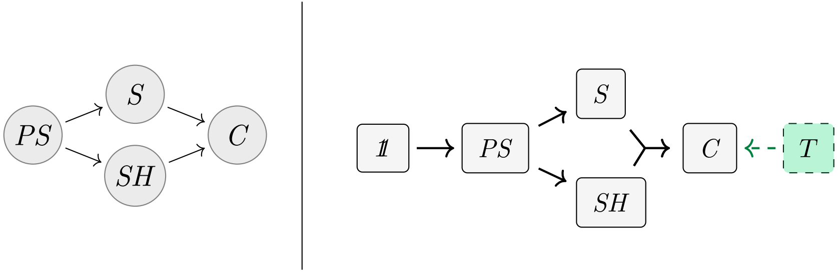




In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;

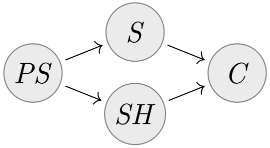
Bayesian Networks as PDGs

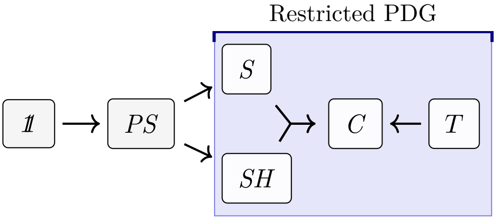


In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;
- a variable can be the target of more than one cpd;

Bayesian Networks as PDGs

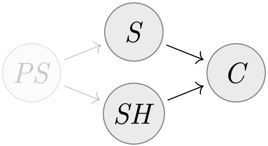




In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;
- a variable can be the target of more than one cpd;
- arbitrary restrictions of PDGs are still PDGs.

Bayesian Networks as PDGs

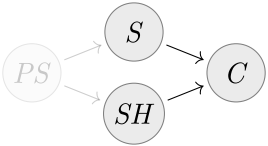


Must now give distributions on SH and S, or distinguish them as "observed" (a *conditional* BN).

In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;
- a variable can be the target of more than one cpd;
- arbitrary restrictions of PDGs are still PDGs.
  - The analogue is false for BNs!

Bayesian Networks as PDGs



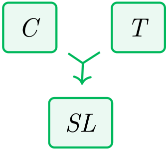
Must now give distributions on SH and S, or distinguish them as "observed" (a *conditional* BN).

In a qualitative BN: *removing* data results in *new* knowledge: $A \perp\!\!\!\perp C$.

In contrast with BNs:

- edge composition has *quantitative* meaning, since edges have cpds;
- a variable can be the target of more than one cpd;
- arbitrary restrictions of PDGs are still PDGs.
  - The analogue is false for BNs!

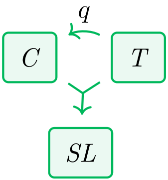
Combining PDGs



Grok wants to be supreme leader (SL).

- She notices that those who use tanning beds have more power, unless they get cancer

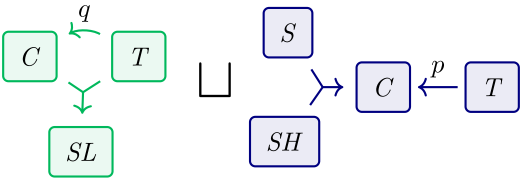
Combining PDGs



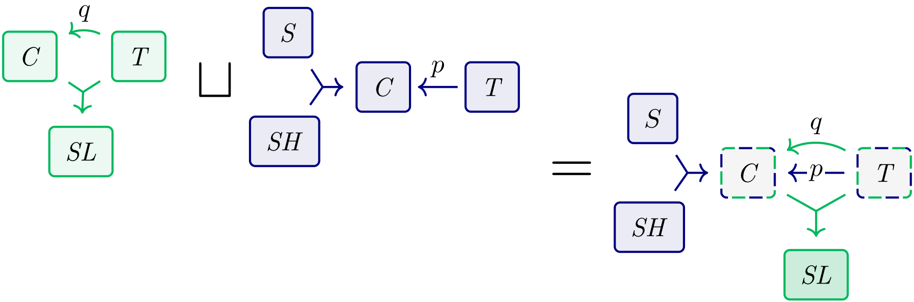
Grok wants to be supreme leader (SL).

- She notices that those who use tanning beds have more power, unless they get cancer
- …but mom says $q(C \mid T) = \begin{array}{cc} & \begin{array}{cc} c & \bar{c} \end{array} \\ & \begin{bmatrix} .15 & .85 \\ .02 & .98 \end{bmatrix} \begin{array}{c} t \\ \bar{t} \end{array} \end{array}.$

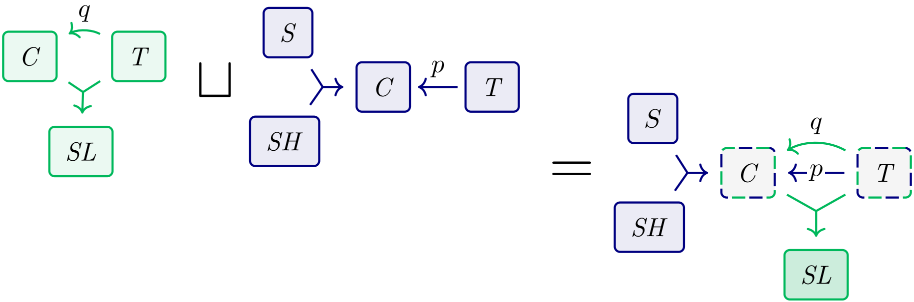
Combining PDGs



Combining PDGs

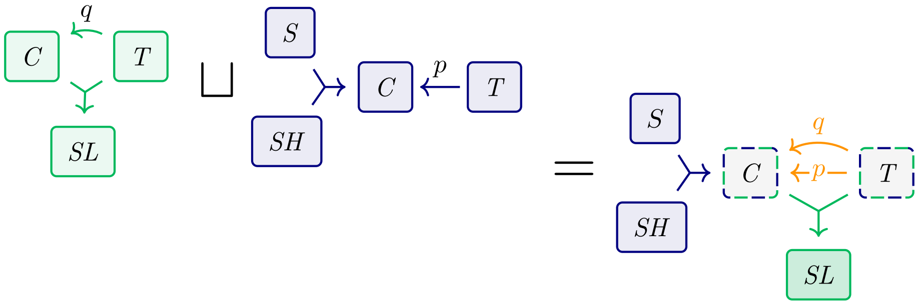


Combining PDGs



- Arbitrary PDGs may be combined without loss of information

Combining PDGs



- Arbitrary PDGs may be combined without loss of information
- They may have parallel edges which directly conflict.

Outline for Section 3

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\boldsymbol{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$,

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\boldsymbol{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$$\boxed{\mathcal{V}(\mathcal{m}) := \prod_{X \in \mathcal{N}} \mathcal{V}(X) \qquad \text{is the set of possible joint variable settings.}}$$

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)
and associated to each $X \xrightarrow{L} Y$, there is:

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)

and associated to each $X \xrightarrow{L} Y$, there is:

$\mathbf{p}_L$ a cpd $\mathbf{p}_L(Y \mid X)$;

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)

and associated to each $X \xrightarrow{L} Y$, there is:

$\mathbf{p}_L$ a cpd $\mathbf{p}_L(Y \mid X)$;

β_L a confidence in the reliability of $\mathbf{p}_L$.

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{m} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)

and associated to each $X \xrightarrow{L} Y$, there is:

$\mathbf{p}_L$ a cpd $\mathbf{p}_L(Y \mid X)$;

β_L a confidence in the reliability of $\mathbf{p}_L$.

Definition (Probabilistic Dependency Graph)

A PDG is a tuple $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

$\mathcal{N}$ is a finite set of nodes (variables)

$\mathcal{V}$ gives a set $\mathcal{V}(X)$ of possible values for each X;

$\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, (or hyper-edges)

and associated to each $X \xrightarrow{L} Y$, there is:

$\mathbf{p}_L$ a cpd $\mathbf{p}_L(Y \mid X)$;

α_L a confidence in the functional dependence $X \to Y$;

β_L a confidence in the reliability of $\mathbf{p}_L$.

Outline for Section 4

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. **Semantics**
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Semantics of PDGs

$\{\!\{m\}\!\} \subseteq \Delta \mathcal{V}(m)$

The set of joint distributions consistent with m;

Semantics of PDGs

$\{\!\{\mathcal{m}\}\!\} \subseteq \Delta\mathcal{V}(\mathcal{m})$

The set of joint distributions consistent with $\mathcal{m}$;

$[\![\mathcal{m}]\!]_\gamma : \Delta\mathcal{V}(\mathcal{m}) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with $\mathcal{m}$;

Semantics of PDGs

$\{\!\{ m \}\!\} \subseteq \Delta\mathcal{V}(m)$

The set of joint distributions consistent with m;

$[\![m]\!]_\gamma : \Delta\mathcal{V}(m) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with m;

$[\![m]\!]^*_\gamma \subseteq \Delta\mathcal{V}(m)$

The distribution(s) most compatible with m (a singleton in many cases of interest);

Semantics of PDGs

$\{\!\!\{ m \}\!\!\} \subseteq \Delta\mathcal{V}(m)$

The set of joint distributions consistent with m;

$[\![m]\!]_\gamma : \Delta\mathcal{V}(m) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with m;

$[\![m]\!]^*_\gamma \subseteq \Delta\mathcal{V}(m)$

The distribution(s) most compatible with m
(a singleton in many cases of interest);

$\langle\!\langle m \rangle\!\rangle_\gamma : \mathbb{R}$

The best posssible compatibility of m with any distribution: the *inconsistency* of m

Semantics of PDGs

$\{\!\{m\}\!\} \subseteq \Delta\mathcal{V}(m)$

The set of joint distributions consistent with m;

$[\![m]\!]_\gamma : \Delta\mathcal{V}(m) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with m;

$[\![m]\!]^*_\gamma \subseteq \Delta\mathcal{V}(m)$

The distribution(s) most compatible with m
(a singleton in many cases of interest);

$\langle\!\langle m \rangle\!\rangle_\gamma : \mathbb{R}$

The best posssible compatibility of m with any distribution: the *inconsistency* of m

...

(other possibilities as well)

Semantics of PDGs

$\{\!\!\{\mathcal{m}\}\!\!\} \subseteq \Delta\mathcal{V}(\mathcal{m})$

The set of joint distributions consistent with $\mathcal{m}$;

$[\![\mathcal{m}]\!]_\gamma : \Delta\mathcal{V}(\mathcal{m}) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with $\mathcal{m}$;

$[\![\mathcal{m}]\!]^*_\gamma \subseteq \Delta\mathcal{V}(\mathcal{m})$

The distribution(s) most compatible with $\mathcal{m}$
(a singleton in many cases of interest);

$\langle\!\langle\mathcal{m}\rangle\!\rangle_\gamma : \mathbb{R}$

The best posssible compatibility of $\mathcal{m}$ with any distribution: the *inconsistency* of $\mathcal{m}$

Semantics of PDGs

$\{\!\{ m \}\!\} \subseteq \Delta\mathcal{V}(m)$

The set of joint distributions consistent with m;

$[\![m]\!]_\gamma : \Delta\mathcal{V}(m) \to \mathbb{R}$

A function (parameterized by $\gamma > 0$) that scores distributions by compatibility with m;

$[\![m]\!]^*_\gamma \subseteq \Delta\mathcal{V}(m)$

The distribution(s) most compatible with m (a singleton in many cases of interest);

$\langle\!\langle m \rangle\!\rangle_\gamma : \mathbb{R}$

The best posssible compatibility of m with any distribution: the *inconsistency* of m

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := Inc_{\mathcal{M}}(\mu) + \gamma \; IDef_{\mathcal{M}}(\mu)$$

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := {\color{green}\mathit{Inc}_{\mathcal{M}}(\mu)} + \gamma \; \mathit{IDef}_{\mathcal{M}}(\mu)$$

Intuition: Measure μ's violation of $\mathcal{M}$'s cpds.

Motivating Examples.

$$\mathcal{M} := \begin{tikzcd} \mathbb{1} \arrow[r, bend left, "q"] \arrow[r, bend right, "p"'] & X \end{tikzcd}$$

Suppose $p = [.4, .6]$.

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{M}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{M}}(\mu)$$

Intuition: Measure μ's violation of $\mathcal{M}$'s cpds.

Motivating Examples.

$$\mathcal{M} := \begin{tikzcd} \mathbb{1} \arrow[r, bend left, "q"] \arrow[r, bend right, "p"'] & X \end{tikzcd}$$

Suppose $p = [.4, .6]$.

- If $p = q$, then $\mathcal{M}$ is clearly consistent, and compatible with the joint distribution $\mu(X) = p = q$, so $\mathit{Inc}_{\mathcal{M}}(p) = 0$.

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Intuition: Measure μ's violation of $\mathcal{m}$'s cpds.

Motivating Examples.

$$\mathcal{m} := \begin{tikzcd} \mathbb{1} \arrow[r, bend left, "q"] \arrow[r, bend right, "p"'] & X \end{tikzcd}$$

Suppose $p = [.4, .6]$.

- If $p = q$, then $\mathcal{m}$ is clearly consistent, and compatible with the joint distribution $\mu(X) = p = q$, so $\mathit{Inc}_{\mathcal{m}}(p) = 0$.
- If q $= [.5, .5]$ then $\mathcal{m}$ is not consistent, but $\mu = [.45, .55]$ matches better than $\mu = [.9, .1]$

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Intuition: Measure μ's violation of $\mathcal{m}$'s cpds.

Motivating Examples.

$$\mathcal{m} := \begin{tikzcd} \boxed{\mathbb{1}} \arrow[r, bend left, "q"] \arrow[r, bend right, "p"'] & \boxed{X} \end{tikzcd}$$

Suppose $p = [.4, .6]$.

- If $p = q$, then $\mathcal{m}$ is clearly consistent, and compatible with the joint distribution $\mu(X) = p = q$, so $\mathit{Inc}_{\mathcal{m}}(p) = 0$.
- If q $= [.5, .5]$ then $\mathcal{m}$ is not consistent, but $\mu = [.45, .55]$ matches better than $\mu = [.9, .1]$
- If $q = [0, 1]$, then $\mathcal{m}$ is much more inconsistent than before, even though $\{\!\{\mathcal{m}\}\!\} = \emptyset$ in both cases.

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Definition (Inc)

The *incompatibility* of a joint distribution μ with $\mathcal{m}$ is given by

$$\mathit{Inc}_{\mathcal{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \boldsymbol{D}(\mu_{Y|X} \parallel \mathbf{p}_L)$$

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Definition (*Inc*)

The *incompatibility* of a joint distribution μ with $\mathcal{m}$ is given by

$$\mathit{Inc}_{\mathcal{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L\ \boldsymbol{D}(\mu_{Y|X} \parallel \mathbf{p}_L)$$

The Scoring Function

$$[\![\boldsymbol{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\boldsymbol{m}}(\mu) + \gamma\ \mathit{IDef}_{\boldsymbol{m}}(\mu)$$

Definition (*Inc*)

The *incompatibility* of a joint distribution μ with $\boldsymbol{m}$ is given by

$$\mathit{Inc}_{\boldsymbol{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L\ \boldsymbol{D}(\mu_{Y|X} \,\|\, \mathbf{p}_L)$$

$$\boldsymbol{D}(\mu \,\|\, \nu) = \sum_{w \in \mathit{Supp}(\mu)} \mu(w) \log \frac{\mu(w)}{\nu(w)}$$ is the relative entropy from ν to μ.

The Scoring Function

$$[\![\boldsymbol{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\boldsymbol{m}}(\mu) + \gamma \; \mathit{IDef}_{\boldsymbol{m}}(\mu)$$

Definition (Inc)

The *incompatibility* of a joint distribution μ with $\boldsymbol{m}$ is given by

$$\mathit{Inc}_{\boldsymbol{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L \mathop{\mathbb{E}}_{x \sim \mu_X} \boldsymbol{D}\Big(\mu(Y \mid X{=}x) \,\Big\|\, \mathbf{p}_{\!L}(x)\Big).$$

$$\boldsymbol{D}(\mu \parallel \nu) = \sum_{w \in \mathit{Supp}(\mu)} \mu(w) \log \frac{\mu(w)}{\nu(w)}$$ is the relative entropy from ν to μ.

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{M}}(\mu) + \gamma \; \mathit{IDef}_{\mathcal{M}}(\mu)$$

Definition (Inc)

The *incompatibility* of a joint distribution μ with $\mathcal{M}$ is given by

$$\mathit{Inc}_{\mathcal{M}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L \mathop{\mathbb{E}}_{x \sim \mu_X} \boldsymbol{D}\Big(\mu(Y \mid X = x) \,\Big\|\, \mathbf{p}_L(x)\Big).$$

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Definition (*IDef*)

The *information deficit* of a distribution μ with respect to $\mathcal{m}$ is

$$\mathit{IDef}_{\mathcal{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \alpha_L \, \mathrm{H}_\mu(Y \mid X) - \mathrm{H}(\mu).$$

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{M}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{M}}(\mu)$$

Definition (*IDef*)

The *information deficit* of a distribution μ with respect to $\mathcal{M}$ is

$$\mathit{IDef}_{\mathcal{M}}(\mu) := \sum_{X \xrightarrow{L} Y} \alpha_L\, \mathrm{H}_\mu(Y \mid X) - \underbrace{\mathrm{H}(\mu)}.$$

(a) # bits needed to determine all variables

The Scoring Function

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Definition (*IDef*)

The *information deficit* of a distribution μ with respect to $\mathcal{m}$ is

$$\mathit{IDef}_{\mathcal{m}}(\mu) := \overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \, \mathrm{H}_\mu(Y \mid X)}^{\text{(b) \# bits required to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{(a) \# bits needed to determine all variables}}.$$

The Scoring Function

$[\![m]\!]_\gamma(\mu) := Inc_m(\mu) + \gamma\, IDef_m(\mu)$

Definition (*IDef*)

The m-*information deficit* of μ:

$$IDef_m(\mu) = \underbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}_{\text{\# bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

Examples

- $m_0 = \boxed{X} \quad \boxed{Y}$

  $IDef_{m_0}(\mu) = -\mathrm{H}_\mu(X,Y)$
  (optimal μ maximizes entropy of X, Y)

People won't know this notation. You can't assume that they do.

The Scoring Function

$[\![m]\!]_\gamma(\mu) := Inc_m(\mu) + \gamma\, IDef_m(\mu)$

Definition (*IDef*)

The m-*information deficit* of μ:

$$IDef_m(\mu) = \underbrace{\overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}^{\text{\# bits to separately determine each target, knowing the source}}}_{} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

Examples

- $m_0 = \boxed{X} \quad \boxed{Y}$

  $IDef_{m_0}(\mu) = -\mathrm{H}_\mu(X, Y)$

  (optimal μ maximizes entropy of X, Y)

- $m_1 = \boxed{X} \longrightarrow \boxed{Y}$

  $IDef_{m_1}(\mu) = -\mathrm{H}_\mu(X)$

  (optimal μ maximizes entropy of X)

The Scoring Function

$$[\![m]\!]_\gamma(\mu) := \mathit{Inc}_m(\mu) + \gamma\, \mathit{IDef}_m(\mu)$$

Definition (*IDef*)

The m-*information deficit* of μ:

$$\mathit{IDef}_m(\mu) = \underbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}_{\text{\# bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

Examples

- $m_0 = \boxed{X} \quad \boxed{Y}$

  $\mathit{IDef}_{m_0}(\mu) = -\mathrm{H}_\mu(X, Y)$

  (optimal μ maximizes entropy of X, Y)

- $m_1 = \boxed{X} \longrightarrow \boxed{Y}$

  $\mathit{IDef}_{m_1}(\mu) = -\mathrm{H}_\mu(X)$

  (optimal μ maximizes entropy of X)

- $m_2 = \boxed{X} \rightrightarrows \boxed{Y}$

  $\mathit{IDef}_{m_2}(\mu) = -\mathrm{H}_\mu(X) + \mathrm{H}_\mu(Y \mid X)$

  (optimal μ maximizes entropy for X, and makes Y a function of X)

The Scoring Function

$$[\![m]\!]_\gamma(\mu) := \mathit{Inc}_{m}(\mu) + \gamma\, \mathit{IDef}_{m}(\mu)$$

Definition (*IDef*)

The m-*information deficit* of μ:

$$\mathit{IDef}_{m}(\mu) = \overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}^{\text{\# bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

Examples

- $m_0 = \boxed{X} \qquad \boxed{Y}$

  $\mathit{IDef}_{m_0}(\mu) = -\,\mathrm{H}_\mu(X, Y)$
  (optimal μ maximizes entropy of X, Y)

- $m_1 = \boxed{X} \longrightarrow \boxed{Y}$

  $\mathit{IDef}_{m_1}(\mu) = -\mathrm{H}_\mu(X)$
  (optimal μ maximizes entropy of X)

- $m_2 = \boxed{X} \rightrightarrows \boxed{Y}$

  $\mathit{IDef}_{m_2}(\mu) = -\,\mathrm{H}_\mu(X) + \mathrm{H}_\mu(Y \mid X)$
  (optimal μ maximizes entropy for X, and makes Y a function of X)

- $m_3 = \boxed{X} \rightleftarrows \boxed{Y}$

  $\mathit{IDef}_{m_3}(\mu) = -\,\mathrm{I}_\mu(X; Y)$
  (opt. μ makes X, Y share information)

The Scoring Function

$$[\![m]\!]_\gamma(\mu) := \mathit{Inc}_m(\mu) + \gamma\, \mathit{IDef}_m(\mu)$$

Definition (*IDef*)

The m-*information deficit* of μ:

$$\mathit{IDef}_m(\mu) = \underbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}_{\text{\# bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

You've lost at least half of your audience at this point, using notation and concepts that they won't understand and you won't have time to explain.
I would cut all this.

Examples

- $m_0 = \boxed{X} \quad \boxed{Y}$

  $\mathit{IDef}_{m_0}(\mu) = -\mathrm{H}_\mu(X, Y)$
  (optimal μ maximizes entropy of X, Y)

- $m_1 = \boxed{X} \longrightarrow \boxed{Y}$

  $\mathit{IDef}_{m_1}(\mu) = -\mathrm{H}_\mu(X)$
  (optimal μ maximizes entropy of X)

- $m_2 = \boxed{X} \rightrightarrows \boxed{Y}$

  $\mathit{IDef}_{m_2}(\mu) = -\mathrm{H}_\mu(X) + \mathrm{H}_\mu(Y \mid X)$
  (optimal μ maximizes entropy for X, and makes Y a function of X)

- $m_3 = \boxed{X} \rightleftarrows \boxed{Y}$

  $\mathit{IDef}_{m_3}(\mu) = -\mathrm{I}_\mu(X; Y)$
  (opt. μ makes X, Y share information)

Information Diagrams

The Scoring Function

$$[\![\mathcal{M}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{M}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{M}}(\mu)$$

tradeoff parameter $\gamma \geq 0$

Definition (*Inc*)

The *incompatibility* of μ with $\mathcal{M}$:

$$\mathit{Inc}_{\mathcal{M}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L\ \boldsymbol{D}(\mu_{Y|X} \parallel \mathbf{p}_L)$$

Definition (*IDef*)

The $\mathcal{M}$-*information deficit* of μ:

bits to separately determine each target, knowing the source

$$\mathit{IDef}_{\mathcal{M}}(\mu) = \overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)} - \underbrace{\mathrm{H}(\mu)}$$

bits to determine all vars

The Scoring Function

- A BN strictly enforces the qualitative picture (large γ)

$$[\![\mathcal{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\mathcal{m}}(\mu) + \gamma\ \mathit{IDef}_{\mathcal{m}}(\mu)$$

Definition (*Inc*)

The *incompatibility* of μ with $\mathcal{m}$:

$$\mathit{Inc}_{\mathcal{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L\ \boldsymbol{D}(\mu_{Y|X} \parallel \mathbf{p}_L)$$

Definition (*IDef*)

The $\mathcal{m}$-*information deficit* of μ:

$$\mathit{IDef}_{\mathcal{m}}(\mu) = \overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}^{\#\text{ bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\#\text{ bits to determine all vars}}$$

The Scoring Function

- A BN strictly enforces the qualitative picture (large γ)
- we are interested in the quantitative limit (small γ)

$$[\![\boldsymbol{m}]\!]_\gamma(\mu) := \mathit{Inc}_{\boldsymbol{m}}(\mu) + \gamma\ \mathit{IDef}_{\boldsymbol{m}}(\mu)$$

Definition (*Inc*)

The *incompatibility* of μ with $\boldsymbol{m}$:

$$\mathit{Inc}_{\boldsymbol{m}}(\mu) := \sum_{X \xrightarrow{L} Y} \beta_L\ \boldsymbol{D}(\mu_{Y|X} \parallel \mathbf{p}_L)$$

Definition (*IDef*)

The $\boldsymbol{m}$-*information deficit* of μ:

$$\mathit{IDef}_{\boldsymbol{m}}(\mu) = \overbrace{\sum_{X \xrightarrow{L} Y} \alpha_L \mathrm{H}_\mu(Y \mid X)}^{\text{\# bits to separately determine each target, knowing the source}} - \underbrace{\mathrm{H}(\mu)}_{\text{\# bits to determine all vars}}$$

Properties of the Optimal Distribution

Proposition (uniqueness for small γ)

1. *If* $0 < \gamma \leq \min_L \beta_L^{\boldsymbol{m}}$*, then* $[\![\boldsymbol{m}]\!]^*_\gamma$ *is a singleton.*
2. $\lim\limits_{\gamma \to 0} [\![\boldsymbol{m}]\!]^*_\gamma$ *exists and is a singleton.*

Properties of the Optimal Distribution

Proposition (uniqueness for small γ)

1. *If* $0 < \gamma \leq \min_L \beta_L^{\mathcal{m}}$, *then* $[\![\mathcal{m}]\!]_\gamma^*$ *is a singleton.*
2. $\lim_{\gamma \to 0} [\![\mathcal{m}]\!]_\gamma^*$ *exists and is a singleton.*

This lets us define $[\![\mathcal{m}]\!]^* :=$ unique element $\left(\lim_{\gamma \to 0} [\![\mathcal{m}]\!]_\gamma^* \right)$.

Proposition (*the set of consistent distributions is the zero set of the scoring function*)

$\{\!\{\mathcal{m}\}\!\} = \{\mu : [\![\mathcal{m}]\!]_0(\mu) = 0\}$.

Proposition (*If there there are distributions consistent with $\mathcal{m}$, the best distribution is one of them.*)

$[\![\mathcal{m}]\!]^* \in [\![\mathcal{m}]\!]_0^*$, *so if* $\mathcal{m}$ *is consistent, then* $[\![\mathcal{m}]\!]^* \in \{\!\{\mathcal{m}\}\!\}$.

Properties of Inconsistency

$$\langle\!\langle \mathcal{M} \rangle\!\rangle_\gamma := \inf_\mu [\![\mathcal{M}]\!]_\gamma$$

Nice properties for minimization:

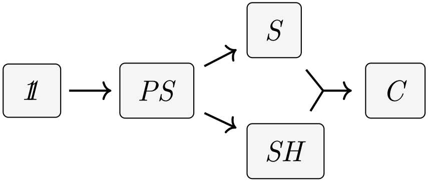
- The function $\gamma \mapsto \langle\!\langle \mathcal{M} \rangle\!\rangle_\gamma$ is continuous for all γ
- The function $p \mapsto \langle\!\langle \mathcal{M} \sqcup p \rangle\!\rangle_\gamma$ is smooth and strictly convex on its interior.

Outline for Section 5

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Capturing Bayesian Networks

For a BN $\mathcal{B}$ with N nodes and a vector $\beta \in \mathbb{R}^N$, let $\boldsymbol{m}_{\mathcal{B},\beta}$ be the PDG corresponding to $\mathcal{B}$, with $\alpha = \mathbf{1}$, and the given vector β of confidences.



Capturing Bayesian Networks

For a BN $\mathcal{B}$ with N nodes and a vector $\beta \in \mathbb{R}^N$, let $\boldsymbol{m}_{\mathcal{B},\beta}$ be the PDG corresponding to $\mathcal{B}$, with $\alpha = \mathbf{1}$, and the given vector β of confidences.

Theorem (*BNs are PDGs*)

If $\mathcal{B}$ is a BN and $\Pr_{\mathcal{B}}$ is the distribution it specifies, then for all $\gamma > 0$ and all vectors β,

$$[\![\boldsymbol{m}_{\mathcal{B},\beta}]\!]^*_\gamma = \{\Pr_{\mathcal{B}}\}, \quad \textit{and thus} \quad [\![\boldsymbol{m}_{\mathcal{B},\beta}]\!]^* = \Pr_{\mathcal{B}}\,.$$

Capturing Bayesian Networks

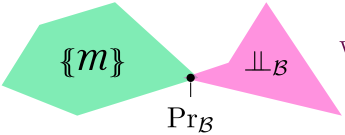
For a BN $\mathcal{B}$ with N nodes and a vector $\beta \in \mathbb{R}^N$, let $\boldsymbol{m}_{\mathcal{B},\beta}$ be the PDG corresponding to $\mathcal{B}$, with $\alpha = \mathbf{1}$, and the given vector β of confidences.

Theorem (*BNs are PDGs*)

If $\mathcal{B}$ is a BN and $\Pr_{\mathcal{B}}$ is the distribution it specifies, then for all $\gamma > 0$ and all vectors β,

$$[\![\boldsymbol{m}_{\mathcal{B},\beta}]\!]^*_\gamma = \{\Pr_{\mathcal{B}}\}, \quad \textit{and thus} \quad [\![\boldsymbol{m}_{\mathcal{B},\beta}]\!]^* = \Pr_{\mathcal{B}}\,.$$

space of distributions consistent with $\boldsymbol{m}_{\mathcal{B}}$ (which minimize *Inc*)



space of distributions with independencies of $\mathcal{B}$ (which can be shown to minimize *IDef*)

Bayesian Networks: Maximum Entropy?

Common distributions tend to maximize entropy subject to natural constraints.

<distribution>	constraints
Gaussian $\mathcal{N}(\mu, \sigma^2)$	mean μ, variance σ^2
Exponential $\mathrm{Exp}(\lambda)$	positive support, mean λ
Factor graphs	moment matching.
...	...

Bayesian Networks: Maximum Entropy?

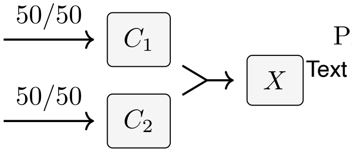
Common distributions tend to maximize entropy subject to natural constraints.

<distribution>	constraints
Gaussian $\mathcal{N}(\mu, \sigma^2)$	mean μ, variance σ^2
Exponential $\mathrm{Exp}(\lambda)$	positive support, mean λ
Factor graphs	moment matching.
...	...
Bayesian Networks	cpds + ???

Bayesian Networks: Maximum Entropy?

Common distributions tend to maximize entropy subject to natural constraints.

<distribution>	constraints
Gaussian $\mathcal{N}(\mu, \sigma^2)$	mean μ, variance σ^2
Exponential $\mathrm{Exp}(\lambda)$	positive support, mean λ
Factor graphs	moment matching.
$\dots$	$\dots$
Bayesian Networks	cpds + ???



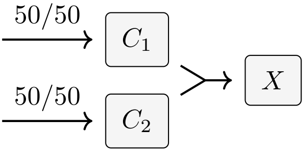
$$\Pr(X \mid C_1, C_2) = \begin{cases} \mathbb{1}[X = x_0] & \text{if } C_1 = C_2 \\ \mathrm{Unif}(X) & \text{if } C_1 \neq C_2 \end{cases}$$

You're going into way too much technical detail here. Your audience is not just Ziv! I would cut all this.

Bayesian Networks: Maximum Entropy?

Common distributions tend to maximize entropy subject to natural constraints.

<distribution>	constraints
Gaussian $\mathcal{N}(\mu, \sigma^2)$	mean μ, variance σ^2
Exponential $\text{Exp}(\lambda)$	positive support, mean λ
Factor graphs	moment matching.
...	...
Bayesian Networks	cpds + ???



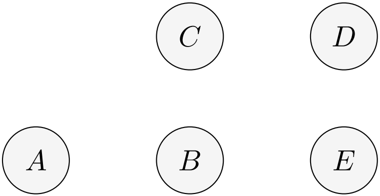
$$\Pr(X \mid C_1, C_2) = \begin{cases} \mathbb{1}[X = x_0] & \text{if } C_1 = C_2 \\ \text{Unif}(X) & \text{if } C_1 \neq C_2 \end{cases}$$

Corollary

Among the distributions in $\{\!\{\mathcal{B}\}\!\}$, $\Pr_{\mathcal{B}}$ *has the maximum entropy, beyond the entropy of the given cpds.*

IDef says maximize: $$\mathrm{H}(\mu) - \sum_{X \in \mathcal{N}} \mathrm{H}_\mu(X \mid \mathbf{Pa}\, X)$$

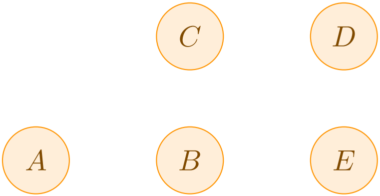
Factor Graphs



Definition

A *factor graph* Φ is

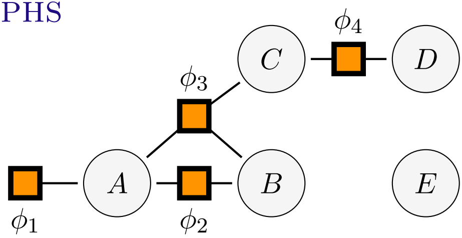
Factor Graphs



Definition

A *factor graph* Φ is a set of variables $\mathcal{X} = \{X_i\}$,

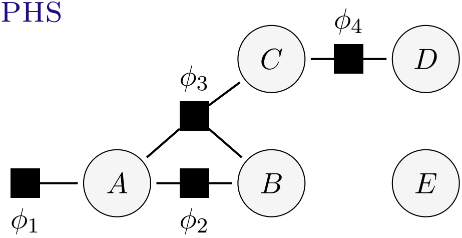
Factor Graphs



Definition

A *factor graph* Φ is a set of variables $\mathcal{X} = \{X_i\}$, and *factors* $\{\phi_J : \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J \in \mathcal{J}}$, with $X_J \subseteq \mathcal{X}$;

Factor Graphs



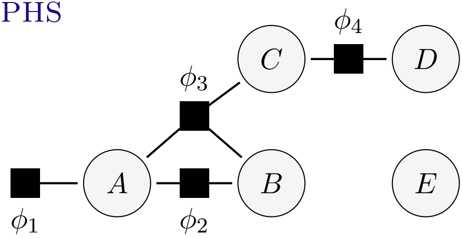
Definition

A *factor graph* Φ is a set of variables $\mathcal{X} = \{X_i\}$, and *factors* $\{\phi_J : \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J \in \mathcal{J}}$, with $X_J \subseteq \mathcal{X}$; Φ defines a distribution

$$\Pr_\Phi(\vec{x}) := \frac{1}{Z_\Phi} \prod_{J \in \mathcal{J}} \phi_J(\vec{x}_J),$$ where Z_Φ is the normalization constant.

Give some intuition for the \phi_J? What do they represent?

Factor Graphs



Definition

A *factor graph* Φ is a set of variables $\mathcal{X} = \{X_i\}$, and *factors* $\{\phi_J : \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J \in \mathcal{J}}$, with $X_J \subseteq \mathcal{X}$; Φ defines a distribution

$$\Pr_\Phi(\vec{x}) := \frac{1}{Z_\Phi} \prod_{J \in \mathcal{J}} \phi_J(\vec{x}_J),$$

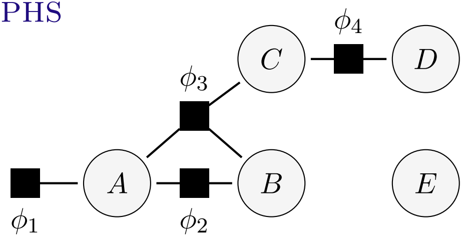
where Z_Φ is the normalization constant.

Φ defines a "variational free energy"

$$VFE_\Phi(\mu) := \mathop{\mathbb{E}}_\mu \Big[- \sum_{J \in \mathcal{J}} \log \phi_J(X_J) \Big] - \mathrm{H}(\mu)$$

Why do we need this? You're overwhelming the poor listener.

Factor Graphs



Definition

A *factor graph* Φ is a set of variables $\mathcal{X} = \{X_i\}$, and *factors* $\{\phi_J : \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J \in \mathcal{J}}$, with $X_J \subseteq \mathcal{X}$; Φ defines a distribution

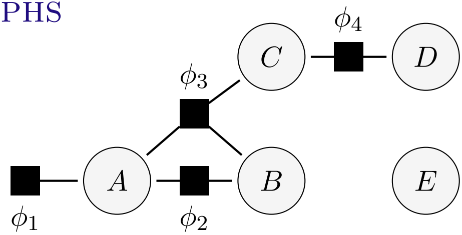
$$\Pr_\Phi(\vec{x}) := \frac{1}{Z_\Phi} \prod_{J \in \mathcal{J}} \phi_J(\vec{x}_J),$$

where Z_Φ is the normalization constant.

Φ defines a "variational free energy"

$$\mathit{VFE}_\Phi(\mu) := \mathop{\mathbb{E}}_{\mu}\Big[- \sum_{J \in \mathcal{J}} \log \phi_J(X_J) \Big] - \mathrm{H}(\mu)$$

Factor Graphs



Definition

A weighted *factor graph* $\Psi = (\Phi, \theta)$ is a set of variables $\mathcal{X} = \{X_i\}$, *factors* $\{\phi_J : \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J \in \mathcal{J}}$, and weights $(\theta_J)_{J \in \mathcal{J}}$ with $X_J \subseteq \mathcal{X}$; Ψ defines a distribution

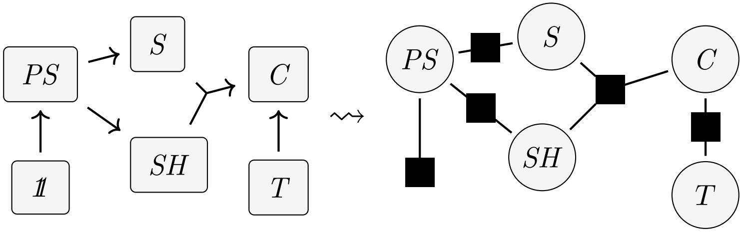
$$\Pr_\Psi(\vec{x}) := \frac{1}{Z_\Phi} \prod_{J \in \mathcal{J}} \phi_J(\vec{x}_J)^{\theta_J},$$

where Z_Ψ is the normalization constant.

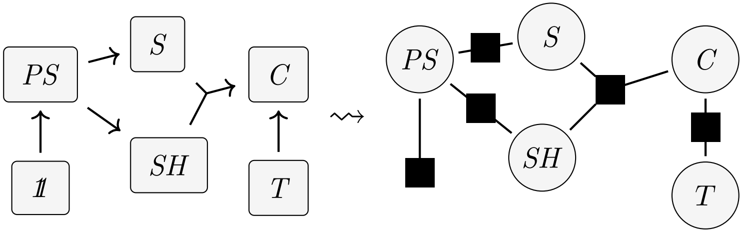
Ψ defines a "variational free energy"

$$\mathit{VFE}_\Phi(\mu) := \mathop{\mathbb{E}}_{\mu}\Big[- \sum_{J \in \mathcal{J}} \theta_J \log \phi_J(X_J) \Big] - \mathrm{H}(\mu)$$

PDGs as Factor Graphs



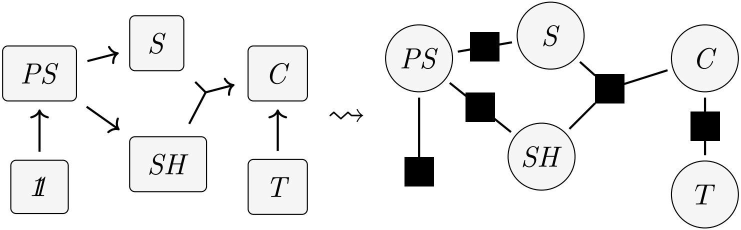
PDGs as Factor Graphs



The cpds of a PDG are essentially factors. Are the semantics different?

What, intuitively, is a factor?

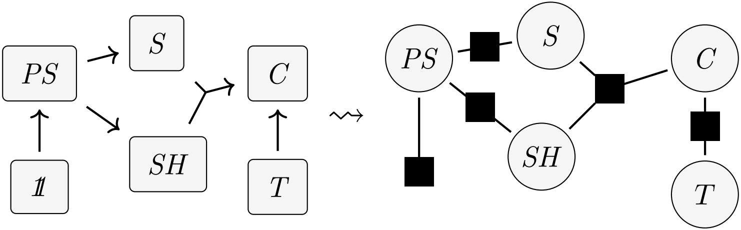
PDGs as Factor Graphs



The cpds of a PDG are essentially factors. Are the semantics different?

Not for $\gamma = 1$.

PDGs as Factor Graphs



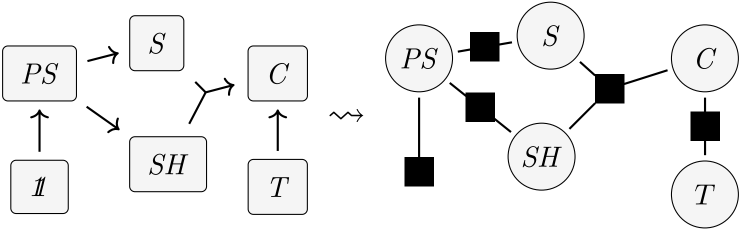
The cpds of a PDG are essentially factors. Are the semantics different?

Not for $\gamma = 1$.

Theorem

$[\![\mathcal{M}]\!]^*_1 = \Pr_{\Phi_{\mathcal{M}}}$ *for all unweighted PDGs* $\mathcal{M}$.

PDGs as Factor Graphs



The cpds of a PDG are essentially factors. Are the semantics different?

Not for $\gamma = 1$.

Theorem

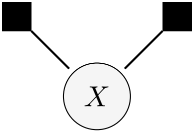
$[\![\mathcal{M}]\!]_1^* = \Pr_{\Phi_{\mathcal{M}}}$ *for all unweighted PDGs* $\mathcal{M}$.

Theorem

For all unweighted PDGs $\mathcal{M}$ *and non-negative vectors* $\mathbf{v}$ *over the edges of* $\mathcal{M}$*, and all* $\gamma > 0$*, we have that* $[\![(\mathcal{M}, \mathbf{v}, \gamma\mathbf{v})]\!]_\gamma = \gamma \; \mathit{VFE}_{(\Phi_{\mathcal{M}}, \mathbf{v})}$*; consequently,* $[\![(\mathcal{M}, \mathbf{v}, \gamma\mathbf{v})]\!]_\gamma^* = \{\Pr_{(\Phi_{\mathcal{M}}, \mathbf{v})}\}$.

An Important Difference between PDGs and Factor Graphs

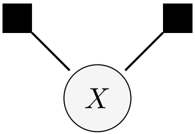
$$m := \begin{tikzcd} \boxed{\mathbb{1}} \arrow[d, bend right, "q"'] \arrow[d, bend left, "p"] \\ \boxed{X} \end{tikzcd}$$



$=: \Phi$

An Important Difference between PDGs and Factor Graphs

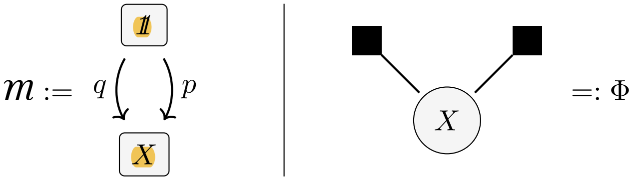
$$m := \begin{tikzcd} \mathbb{1} \arrow[d, bend right, "q"'] \arrow[d, bend left, "p"] \\ X \end{tikzcd}$$



$=: \Phi$

- If $p = q$, then $[\![m]\!]^* = p = q \ldots$

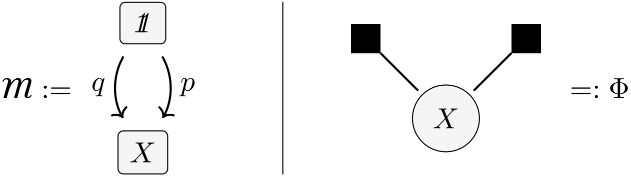
An Important Difference between PDGs and Factor Graphs



- If $p = q$, then $[\![m]\!]^* = p = q \ldots$
- $\ldots$ but $\Pr_\Phi \propto p^2$

Cut this. It will be of interest only to experts in factor graphs (which I believe is none of your audience).

An Important Difference between PDGs and Factor Graphs

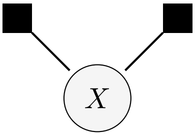


- If $p = q$, then $[\![m]\!]^* = p = q \ldots$
- $\ldots$ but $\Pr_\Phi \propto p^2$
- Individual factors have *no probabilistic meaning*,

So what do they mean? You have to say something about this.

An Important Difference between PDGs and Factor Graphs

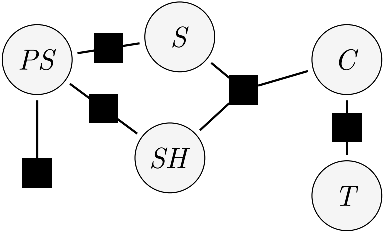
$$\boldsymbol{m} := \begin{tikzcd} \boxed{\mathbb{1}} \arrow[d, bend right, "q"'] \arrow[d, bend left, "p"] \\ \boxed{X} \end{tikzcd}$$



$=: \Phi$

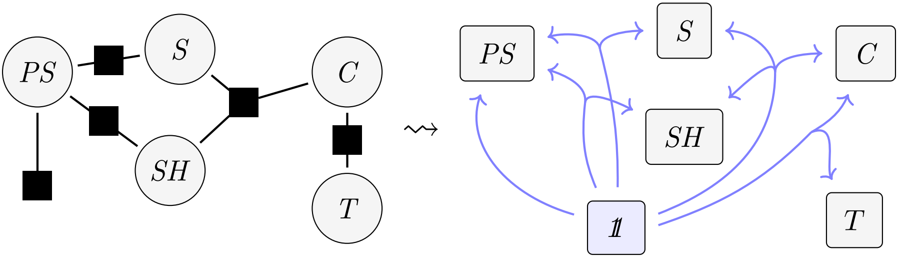
- If $p = q$, then $[\![\boldsymbol{m}]\!]^* = p = q \ldots$
- $\ldots$ but $\Pr_\Phi \propto p^2$
- Individual factors have *no probabilistic meaning*,
- a factor graph can fail to normalize, in which case it has no global semantics either.

Factor Graphs as PDGs

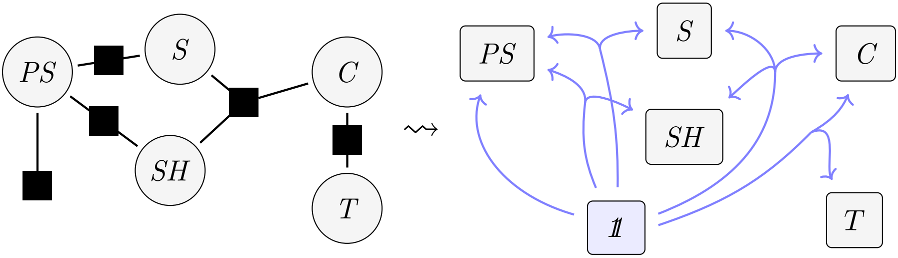


$\rightsquigarrow$

Factor Graphs as PDGs



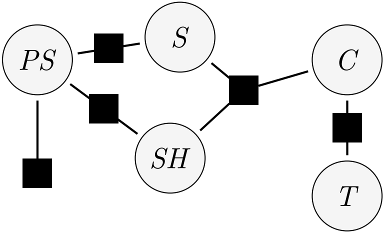
Factor Graphs as PDGs



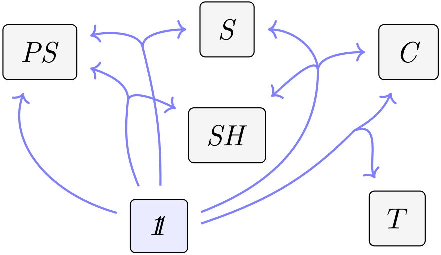
Theorem

$\Pr_\Phi = [\![\mathcal{n}_\Phi]\!]_1^*$ *for all factor graphs* Φ.

Factor Graphs as PDGs



$\rightsquigarrow$



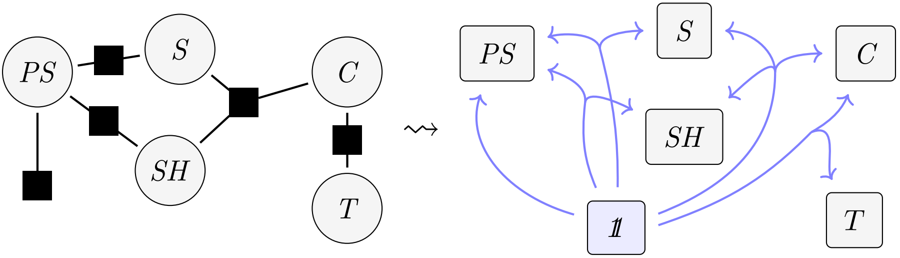
Theorem

$\Pr_\Phi = [\![\mathcal{n}_\Phi]\!]_1^*$ *for all factor graphs* Φ*.*

Theorem

For all weighted factor graphs $\Psi = (\Phi, \theta)$ *and all* $\gamma > 0$*, we have that* $\mathit{VFE}_\Psi = {}^1\!/\!_\gamma [\![\mathcal{m}_{\Psi,\gamma}]\!]_\gamma + C$ *for some constant* C*, so* $\Pr_\Psi$ *is the unique element of* $[\![\mathcal{m}_{\Psi,\gamma}]\!]_\gamma^*$*.*

Factor Graphs as PDGs



Theorem

$\Pr_\Phi = [\![\mathcal{N}_\Phi]\!]_1^*$ *for all factor graphs* Φ.

Theorem

For all weighted factor graphs $\Psi = (\Phi, \theta)$ *and all* $\gamma > 0$, *we have that* $\mathit{VFE}_\Psi = {}^1\!/\!{}_\gamma [\![\mathcal{M}_{\Psi,\gamma}]\!]_\gamma + C$ *for some constant* C, *so* $\Pr_\Psi$ *is the unique element of* $[\![\mathcal{M}_{\Psi,\gamma}]\!]_\gamma^*$.

Also: $\log Z_\Phi = \langle\!\langle \mathcal{N}_\Phi \rangle\!\rangle_1$.

Letting $x^{\mathbf{w}}$ and $y^{\mathbf{w}}$ denote the values of X and Y, respectively, in $\mathbf{w} \in \mathcal{V}(\boldsymbol{m})$, we have

$$[\![\boldsymbol{m}]\!](\mu) = \mathop{\mathbb{E}}_{\mathbf{w}\sim\mu}\Bigg\{ \sum_{X \xrightarrow{L} Y} \Bigg[\overbrace{\beta_L \log \frac{1}{\mathbf{p}_L(y^{\mathbf{w}}|x^{\mathbf{w}})}}^{\text{log likelihood / cross entropy}} + \underbrace{(\alpha_L\gamma - \beta_L) \log \frac{1}{\mu(y^{\mathbf{w}}|x^{\mathbf{w}})}}_{\text{local regularization } (\beta_L > \alpha_L\gamma)} \Bigg] - \underbrace{\gamma \log \frac{1}{\mu(\mathbf{w})}}_{\text{global regularization}} \Bigg\}.$$

Outline for Section 6

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.

To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\left[\!\!\left[\mathcal{m} \sqcup (Y{=}y) \right]\!\!\right]^* = [\![\mathcal{m}]\!]^* \mid (Y{=}y).$$

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.

To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\left[\!\left[\mathcal{M} \sqcup (Y{=}y) \right]\!\right]^* = [\![\mathcal{M}]\!]^* \mid (Y{=}y).$$

Querying $\Pr(Y \mid X)$ **in a PDG** $\mathcal{M}$**.**

- We can add $X \xrightarrow{p} Y$ to $\mathcal{M}$, to get $\mathcal{M} \sqcup p$.

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.

To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\Big[\!\!\Big[\mathcal{m} \sqcup (Y{=}y)\Big]\!\!\Big]^* = [\![\mathcal{m}]\!]^* \mid (Y{=}y).$$

Querying $\Pr(Y \mid X)$ in a PDG $\mathcal{m}$.

- We can add $X \xrightarrow{p} Y$ to $\mathcal{m}$, to get $\mathcal{m} \sqcup p$.
- The choice of cpd p that minimizes the inconsistency of $\mathcal{m} \sqcup p$ (which is strictly convex and smooth in p) is $[\![\mathcal{m}]\!]^*(Y \,|\, X)$,

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.

To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\Big[\!\!\Big[\mathcal{m} \sqcup (Y{=}y) \Big]\!\!\Big]^* = [\![\mathcal{m}]\!]^* \mid (Y{=}y).$$

Querying $\Pr(Y \mid X)$ **in a PDG** $\mathcal{m}$**.**

- We can add $X \xrightarrow{p} Y$ to $\mathcal{m}$, to get $\mathcal{m} \sqcup p$.
- The choice of cpd p that minimizes the inconsistency of $\mathcal{m} \sqcup p$ (which is strictly convex and smooth in p) is $[\![\mathcal{m}]\!]^*(Y \,|\, X)$,
  - so an inconsistency oracle yields fast inference by gradient descent.

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.
To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\Big[\!\!\Big[\mathcal{M} \sqcup (Y{=}y) \Big]\!\!\Big]^* = [\![\mathcal{M}]\!]^* \mid (Y{=}y).$$

Querying $\Pr(Y \mid X)$ **in a PDG** $\mathcal{M}$**.**

- We can add $X \xrightarrow{p} Y$ to $\mathcal{M}$, to get $\mathcal{M} \sqcup p$.
- The choice of cpd p that minimizes the inconsistency of $\mathcal{M} \sqcup p$ (which is strictly convex and smooth in p) is $[\![\mathcal{M}]\!]^*(Y \,|\, X)$,
  - so an inconsistency oracle yields fast inference by gradient descent.

(Theorem): Unfortunately,

1. Deciding if $\mathcal{M}$ is consistent is NP-hard.
2. Computing $\langle\!\langle \mathcal{M} \rangle\!\rangle_\gamma$ is #P-hard, for $\gamma > 0$.

Inference via Inconsistency Reduction

Identify the event $Y{=}y$ with the cpd $\mathbb{1} \xrightarrow{\delta_y} Y$.

Conditioning as inconsistency resolution.
To condition on an event $(Y{=}y)$, simply add it to the PDG. Then the new best distribution is the old one, conditioned on $(Y{=}y)$. That is,

$$\left[\!\left[\mathcal{m} \sqcup (Y{=}y) \right]\!\right]^* = [\![\mathcal{m}]\!]^* \mid (Y{=}y).$$

Querying $\Pr(Y \mid X)$ **in a PDG** $\mathcal{m}$**.**

- We can add $X \xrightarrow{p} Y$ to $\mathcal{m}$, to get $\mathcal{m} \sqcup p$.
- The choice of cpd p that minimizes the inconsistency of $\mathcal{m} \sqcup p$ (which is strictly convex and smooth in p) is $[\![\mathcal{m}]\!]^*(Y \mid X)$,
  - so an inconsistency oracle yields fast inference by gradient descent.

(Theorem): Unfortunately,

1. Deciding if $\mathcal{m}$ is consistent is NP-hard.
2. Computing $\langle\!\langle \mathcal{m} \rangle\!\rangle_\gamma$ is #P-hard, for $\gamma > 0$.

...just like for BNs and Factor Graphs.

Outline for Section 7

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...
  - choice is made by instinct, tradition, or pragmatics, which makes results difficult to motivate, and vulnerable to overfitting.

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...
  - choice is made by instinct, tradition, or pragmatics, which makes results difficult to motivate, and vulnerable to overfitting.
- Choice of *model* admits more principled discussion.

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...
  - choice is made by instinct, tradition, or pragmatics, which makes results difficult to motivate, and vulnerable to overfitting.
- Choice of *model* admits more principled discussion.

This is important, and it needs *much* more discussion.
How do you use the model? Where might it come from?

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...
  - choice is made by instinct, tradition, or pragmatics, which makes results difficult to motivate, and vulnerable to overfitting.
- Choice of *model* admits more principled discussion.

Surprising Result

Most standard objectives arise naturally as the inconsistency of the obvious PDG describing the situation.

Inconsistency: The Universal Loss

- Fruitful to cast AI as optimization. But what to optimize?
  - `Cross Entropy`, `Square Loss`, `Accuracy`, ...
  - choice is made by instinct, tradition, or pragmatics, which makes results difficult to motivate, and vulnerable to overfitting.
- Choice of *model* admits more principled discussion.

Surprising Result

Most standard objectives arise naturally as the inconsistency of the obvious PDG describing the situation.

Bonus

An intuitive visual language for reasoning about relationships between objective functions.

Surprise as Inconsistency

This can't come out of the blue. You need to introduce surprise, and explain why people think it's a reasonable objective the minimize.

Proposition

Consider a distribution over X with mass function $p(X)$. The surprise (or information content) $\mathrm{I}_p(x) := -\log p(X{=}x)$ at seeing a sample x is the inconsistency of the pdg containing p and the event $X = x$, i.e.,

$$\mathrm{I}_p(x) = \log \frac{1}{p(X{=}x)} = \left\langle\!\!\left\langle \xrightarrow{p} \boxed{X} \xleftarrow{X=x} \right\rangle\!\!\right\rangle.$$

Surprise as Inconsistency

Proposition

Consider a distribution over X with mass function $p(X)$. The surprise (or information content) $\mathrm{I}_p(x) := -\log p(X{=}x)$ at seeing a sample x is the inconsistency of the pdg containing p and the event $X = x$, i.e.,

$$\mathrm{I}_p(x) = \log \frac{1}{p(X{=}x)} = \left\langle\!\!\left\langle \xrightarrow{p} \boxed{X} \xleftarrow{X=x} \right\rangle\!\!\right\rangle.$$

- PDG semantics just so happen to give the standard meaure of compatibility between a sample and distribution.

Surprise as Inconsistency

Proposition

Consider a distribution over X with mass function $p(X)$. The surprise (or information content) $\mathrm{I}_p(x) := -\log p(X{=}x)$ at seeing a sample x is the inconsistency of the pdg containing p and the event $X = x$, i.e.,

$$\mathrm{I}_p(x) = \log \frac{1}{p(X{=}x)} = \left\langle\!\!\left\langle \xrightarrow{p} \boxed{X} \xleftarrow{X=x} \right\rangle\!\!\right\rangle.$$

- PDG semantics just so happen to give the standard meaure of compatibility between a sample and distribution.
- Known as "surprise", a particular kind of internal conflict.

It's too late to say this here. "Surprise" has already been used in the proposition. You need to introduce surprise at the beginning, not the end!

Variations: Surprise as Inconsistency

Proposition (marginal information as inconsistency)

If $p(X, Z)$ is a joint distribution, the (marginal) information of the (partial) observation $X = x$ is given by

$$\mathrm{I}_p(x) = \log \frac{1}{p(x)} = \left\langle\!\!\left\langle \boxed{Z} \overset{p}{\swarrow\!\searrow} \boxed{X} \overset{x}{\twoheadleftarrow} \right\rangle\!\!\right\rangle.$$

Variations: Surprise as Inconsistency

Proposition (marginal information as inconsistency)

If $p(X, Z)$ is a joint distribution, the (marginal) information of the (partial) observation $X = x$ is given by

$$\mathrm{I}_p(x) = \log \frac{1}{p(x)} = \left\langle\!\!\left\langle \begin{tikzcd} \boxed{Z} & \bullet \arrow[l] \arrow[r, "p"'] & \boxed{X} & \arrow[l, twoheadrightarrow, "x"'] \end{tikzcd} \right\rangle\!\!\right\rangle.$$

Proposition (cross entropy as inconsistency)

What's f?

The inconsistency of the PDG containing $f(Y \mid X)$ and a high-confidence empirical distribution $\Pr_{\underline{\mathbf{xy}}}$ of samples $\underline{\mathbf{xy}} = \{(x_i, y_i)\}$ is equal to the cross entropy (plus $\mathrm{H}(Y \mid X)$, a constant that depends only on the data $\Pr_{\underline{\mathbf{xy}}}$). That is,

Where did "high confidence" come from? This needs much more discussion and intuition. You're far too focused on listing results, rather than explaining them

$$\left\langle\!\!\left\langle \begin{tikzcd} & \Pr_{\underline{\mathbf{xy}}} \; {\scriptstyle (\beta:\infty)} \arrow[dl] \arrow[dr] & \\ \boxed{X} \arrow[rr, "f"'] & & \boxed{Y} \end{tikzcd} \right\rangle\!\!\right\rangle = \frac{1}{|\underline{\mathbf{xy}}|} \sum_{(x,y)\in\underline{\mathbf{xy}}} \left[\log \frac{1}{f(y \mid x)}\right] \quad - \mathrm{H}_{\Pr_{\underline{\mathbf{xy}}}}(Y \mid X).$$

Proposition (Accuracy as Inconsistency)

Consider a predictor $h : X \to Y$ for true labels $f : X \to Y$, and a distribution $D(X)$. The inconsistency of believing all three is

$$\left\langle\!\!\left\langle \xrightarrow{D^{(\beta)}} \boxed{X} \overset{h}{\underset{f}{\rightrightarrows}} \boxed{Y} \right\rangle\!\!\right\rangle = -\beta \log\Big(\mathrm{accuracy}_{f,D}(h)\Big) = \beta\ \mathrm{I}_D[f = h].$$

I have no clue what the notation D^{(\beta)} means. Snowing me with notation like this is a *bad* idea.

Proposition (Mean Square Error as Inconsistency)

$$\left\langle\!\!\left\langle \begin{tikzcd} {} \arrow[r, "D!"] & X \arrow[r, bend left, "\mathcal{N}(f(x),1)"] \arrow[r, bend right, "\mathcal{N}(g(x),1)"'] & Y \end{tikzcd} \right\rangle\!\!\right\rangle = \left\langle\!\!\left\langle \begin{tikzcd} & & \mu_f \arrow[dr, "\mathcal{N}_1"] & \\ {} \arrow[r, "D!"] & X \arrow[ur, two heads, "f"] \arrow[dr, two heads, "h"'] & & Y \\ & & \mu_h \arrow[ur, "\mathcal{N}_1"'] & \end{tikzcd} \right\rangle\!\!\right\rangle$$

Same comment as above
for D^{\beta}

$$= \mathbb{E}_D \Big(f(X) - h(X) \Big)^2 =: \mathrm{MSE}(f,h)$$

where $\mathcal{N}_1 = \mathcal{N}(-,1)$ *is the normal distribution with unit variance, and mean equal to its argument.*

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$,

That is,

$$\left\langle\!\!\left\langle \quad \boxed{\Theta} \xrightarrow{f} \boxed{Y} \quad \right\rangle\!\!\right\rangle$$

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$*, have a prior* $p(\theta)$*,*

That is,

$$\left\langle\!\!\left\langle \xrightarrow{p^{(\beta)}} \boxed{\Theta} \xrightarrow{f} \boxed{Y} \right\rangle\!\!\right\rangle$$

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$, have a prior $p(\theta)$, and have an empirical distribution $D(Y)$ which you trust.

That is,

$$\left\langle\!\!\left\langle \xrightarrow{p^{(\beta)}} \boxed{\Theta} \xrightarrow{f} \boxed{Y} \xleftarrow{D!} \right\rangle\!\!\right\rangle$$

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$, have a prior $p(\theta)$, and have an empirical distribution $D(Y)$ which you trust. Then the inconsistency of also believing $\Theta = \theta_0$ is

That is,

$$\left\langle\!\!\left\langle \begin{array}{c} \xrightarrow{p^{(\beta)}} \\ \xrightarrow[\theta_0]{} \end{array} \boxed{\Theta} \xrightarrow{f} \boxed{Y} \xleftarrow{D!} \right\rangle\!\!\right\rangle =$$

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$, have a prior $p(\theta)$, and have an empirical distribution $D(Y)$ which you trust. Then the inconsistency of also believing $\Theta = \theta_0$ is the regularized-cross entropy loss, and controlled by the strength β_p of the prior. That is,

I'm feeling overwhelmed by notation.

$$\left\langle\!\!\left\langle \begin{array}{c} \xrightarrow{p^{(\beta)}} \\ \xrightarrow[\theta_0]{} \end{array} \boxed{\Theta} \xrightarrow{f} \boxed{Y} \xleftarrow{D!} \right\rangle\!\!\right\rangle = \mathop{\mathbb{E}}_{y\sim D}\left[\log\frac{1}{f(y \mid \theta_0)}\right] + \beta\log\frac{1}{p(\theta_0)} - \mathrm{H}(D)$$

Proposition (Regularizers as priors)

Suppose you believe $Y \sim f_\theta(Y)$, have a prior $p(\theta)$, and have an empirical distribution $D(Y)$ which you trust.Then the inconsistency of also believing $\Theta = \theta_0$ isthe regularized-cross entropy loss, and controlled by the strength β_p of the prior. That is,

$$\left\langle\!\!\left\langle \xrightarrow{p^{(\beta)}} \overset{\theta_0}{\twoheadrightarrow} \boxed{\Theta} \xrightarrow{f} \boxed{Y} \xleftarrow{D!} \right\rangle\!\!\right\rangle = \mathop{\mathbb{E}}_{y\sim D}\left[\log\frac{1}{f(y \mid \theta_0)}\right] + \beta \log \frac{1}{p(\theta_0)} - \mathrm{H}(D)$$

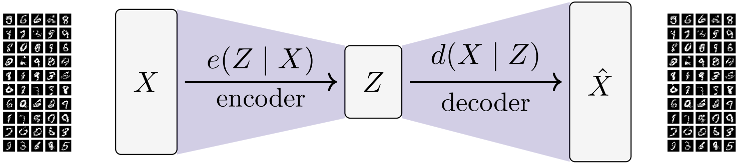
Using a (discretized) unit gaussian as a prior, $p(\theta) = \frac{1}{k}\exp(-\frac{1}{2}\theta^2)$ for a normaization constant k, the RHS becomes

$$\underbrace{\mathop{\mathbb{E}}_{D}\left[\log\frac{1}{f(Y \mid \theta_0)}\right]}_{\substack{\text{Cross entropy loss of } f_\theta \text{ w.r.t. } D \\ \text{(data-fit cost of } \theta_0)}} \quad + \quad \underbrace{\frac{\beta}{2}\theta_0^2}_{\substack{\ell_2 \text{ regularizer} \\ \text{(complexity cost of } \theta_0)}} \quad \underbrace{+\beta\log k - \mathrm{H}(D)}_{\text{constant in } f \text{ and } \theta_0}.$$

At this point, I guarantee you've lost your audience, unless you're prepared to spend 10 minutes explaining this.

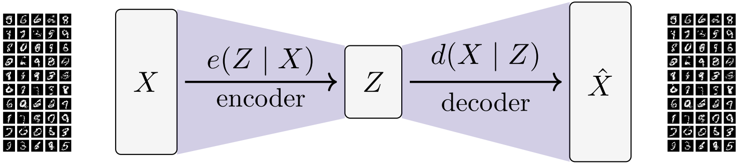
Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):



Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):

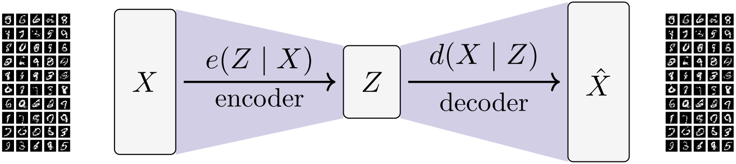


- Objective:

What's Z? Where did it come from?

Variational Auto-Encoders, Take 1

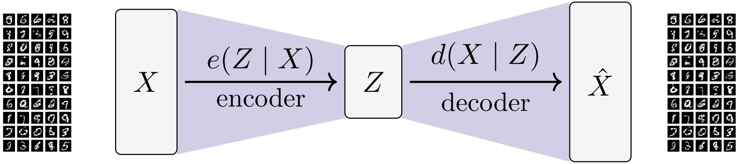
- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}^{\text{“reconstruction error”}}$

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):

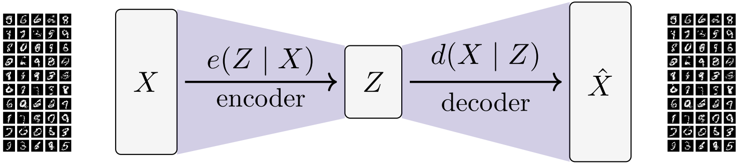


- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{-\mathop{\mathbb{E}}_{z\sim e|x} \log d(x \mid z)}^{\text{“reconstruction error”}}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.

Why? Where did this come from? (I still don’t know what Z is.)

Variational Auto-Encoders, Take 1

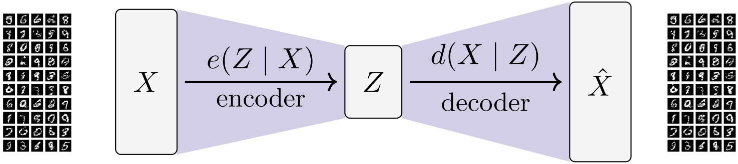
- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overset{\text{“reconstruction error”}}{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get
    $\mathrm{ELBO}_{p,e,d}(x) :=$

Variational Auto-Encoders, Take 1

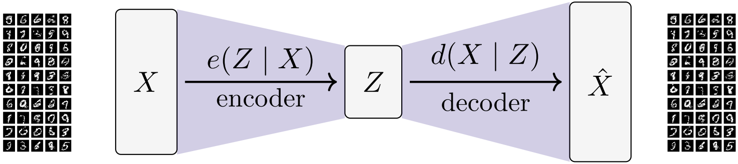
- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}^{\text{"reconstruction error"}}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get $\mathrm{ELBO}_{p,e,d}(x) :=$ $-\underbrace{\boldsymbol{D}\big(e(Z|x) \,\|\, p(Z)\big)}_{\text{divergence from prior}}$

Variational Auto-Encoders, Take 1

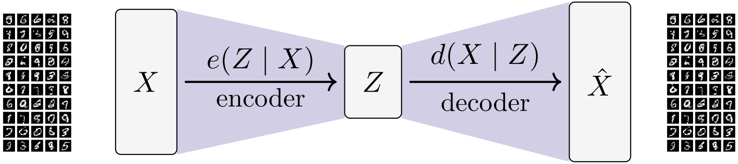
- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}^{\text{"reconstruction error"}}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get
    $\mathrm{ELBO}_{p,e,d}(x) :=$
    $- \underbrace{\boldsymbol{D}\big(e(Z|x) \,\|\, p(Z)\big)}_{\text{divergence from prior}} - \mathrm{Rec}(x)$

Variational Auto-Encoders, Take 1

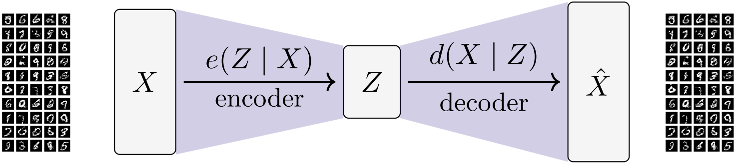
- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}^{\text{“reconstruction error”}}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get
    $\mathrm{ELBO}_{p,e,d}(x) :=$
    $$\underbrace{-\boldsymbol{D}\big(e(Z|x) \,\|\, p(Z)\big)}_{\text{divergence from prior}} - \mathrm{Rec}(x) = \mathop{\mathbb{E}}_{z \sim e|x} \left[\log \frac{p(z) d(x \mid z)}{e(z \mid x)} \right]$$

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):



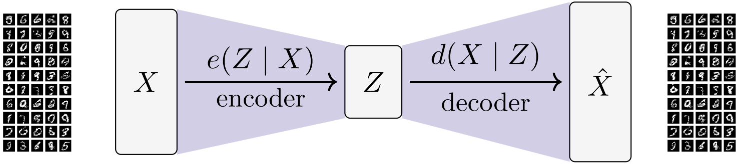
- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := \overbrace{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}^{\text{"reconstruction error"}}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get

    $\mathrm{ELBO}_{p,e,d}(x) :=$

    $$\underbrace{-\boldsymbol{D}\big(e(Z|x) \,\|\, p(Z)\big)}_{\text{divergence from prior}} - \mathrm{Rec}(x) = \mathop{\mathbb{E}}_{z \sim e|x}\left[\log \frac{p(z) d(x \mid z)}{e(z \mid x)}\right] \leq \overbrace{\log \Pr_{pd}(x)}^{\text{"evidence"}}$$

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):



- Objective:
  - For each x, want to minimize $\mathrm{Rec}(x) := -\overset{\text{“reconstruction error”}}{\underset{z\sim e|x}{\mathbb{E}} \log d(x \mid z)}$
  - Also have a prior $p(Z)$ that we want encodings of x to follow.
  - Add terms & negate to get

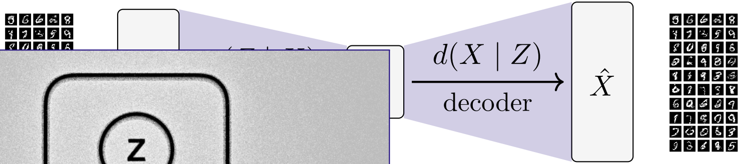
    $\mathrm{ELBO}_{p,e,d}(x) :=$

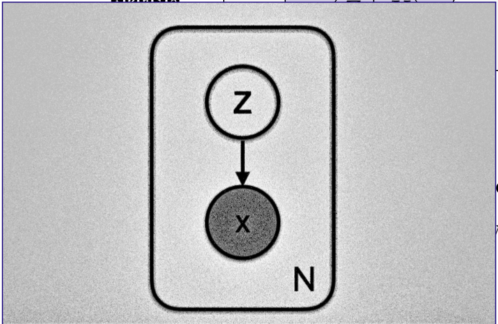
    $$-\underbrace{\boldsymbol{D}\big(e(Z|x) \,\|\, p(Z)\big)}_{\text{divergence from prior}} - \mathrm{Rec}(x) = \underset{z\sim e|x}{\mathbb{E}}\left[\log \frac{p(z)d(x \mid z)}{e(z \mid x)}\right] \leq \overset{\text{“evidence”}}{\log \Pr_{pd}(x)}$$

Urge to use graphical models (even if can't quite capture *entire* VaE)

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):





THE GRAPHICAL MODEL REPRESENTATION OF THE MODEL IN THE VARIATIONAL AUTOENCODER. THE LATENT VARIABLE Z IS A STANDARD NORMAL, AND THE DATA ARE DRAWN FROM P(X|Z). THE SHADED NODE FOR X DENOTES OBSERVED DATA. FOR BLACK AND WHITE IMAGES OF HANDWRITTEN DIGITS, THIS DATA LIKELIHOOD IS BERNOULLI DISTRIBUTED.

$$\text{c}(x) := \overset{\text{“reconstruction error”}}{- \mathop{\mathbb{E}}_{z \sim e|x} \log d(x \mid z)}$$

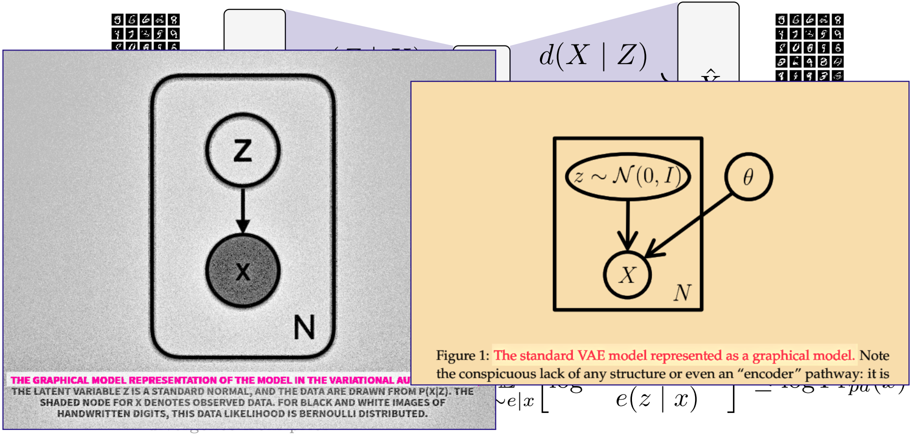
ant encodings of x to follow.

$$\mathop{\mathbb{E}}_{\sim e|x} \left[\log \frac{p(z) d(x \mid z)}{e(z \mid x)} \right] \leq \overset{\text{“evidence”}}{\log \Pr_{pd}(x)}$$

Urge to use graphical models (even if can't quite capture *entire* VaE)

Variational Auto-Encoders, Take 1

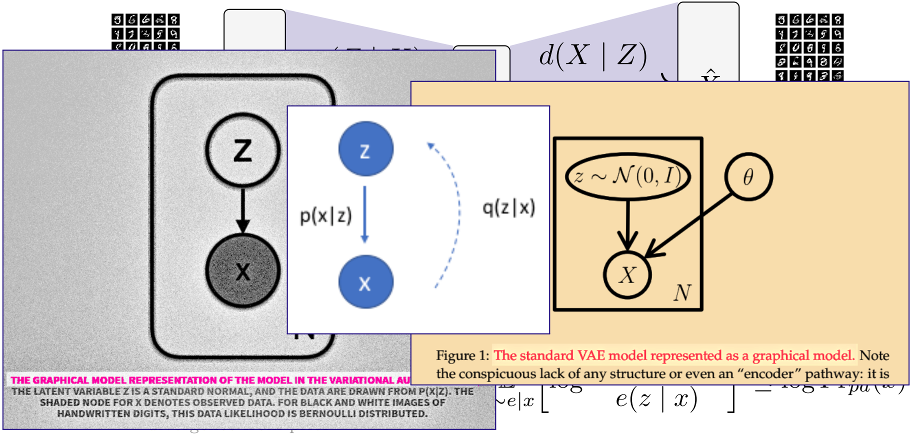
- Structure consists of two neural networks (cpds):



Urge to use graphical models (even if can't quite capture *entire* VaE)

Variational Auto-Encoders, Take 1

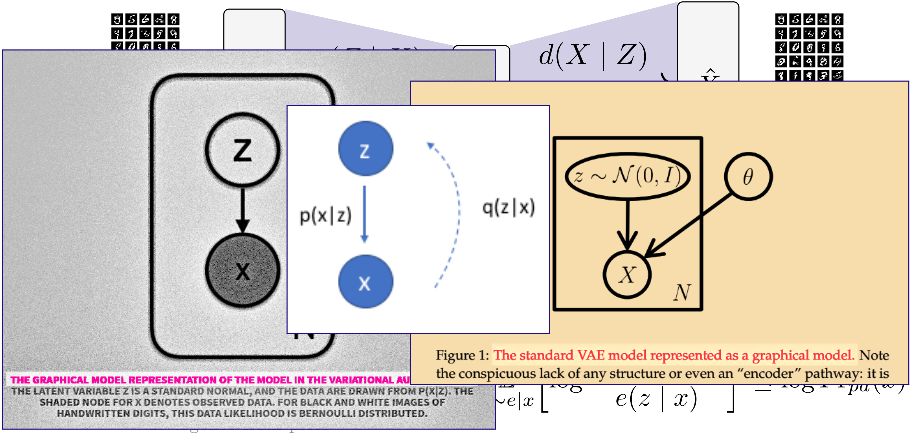
- Structure consists of two neural networks (cpds):



Urge to use graphical models (even if can't quite capture *entire* VaE)

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):

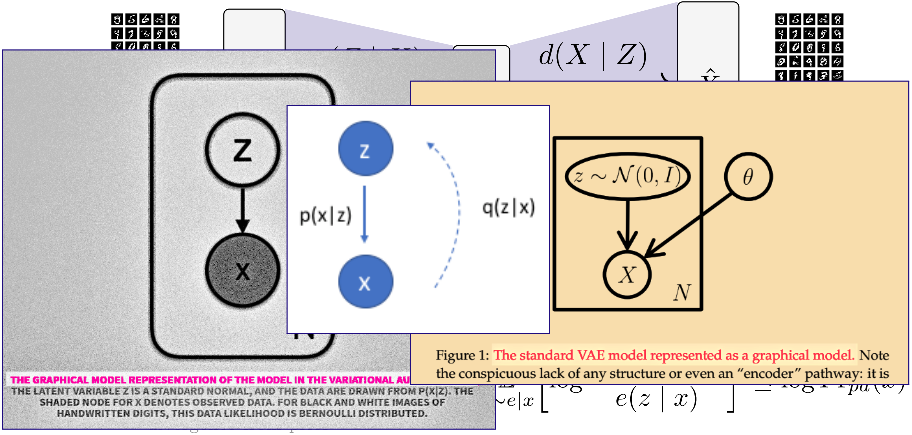


Urge to use graphical models (even if can't quite capture *entire* VaE)

- $e(Z \mid X)$ has same target as $p(Z)$, so can't put in BN;

Variational Auto-Encoders, Take 1

- Structure consists of two neural networks (cpds):



Urge to use graphical models (even if can't quite capture *entire* VaE)

- $e(Z \mid X)$ has same target as $p(Z)$, so can't put in BN;
- The heart of the VaE is not its structure, but its objective.

Variational Auto-Encoders, Take 2

- Structure:

Variational Auto-Encoders, Take 2

- Structure:
  $e(Z \mid X)$: encoder

$$\begin{tikzcd} Z & X \arrow[l, bend left, "e"] \end{tikzcd}$$

Variational Auto-Encoders, Take 2

- Structure:

  $e(Z \mid X)$: encoder

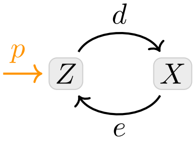
  $d(X \mid Z)$: decoder

$$\begin{tikzcd} Z \arrow[r, bend left, "d"] & X \arrow[l, bend left, "e"] \end{tikzcd}$$

Variational Auto-Encoders, Take 2

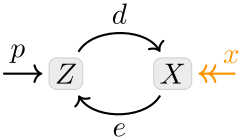
- Structure:

  $e(Z \mid X)$: encoder
  $d(X \mid Z)$: decoder
  $p(Z)$: prior



Variational Auto-Encoders, Take 2

- Structure:
  $e(Z \mid X)$: encoder
  $d(X \mid Z)$: decoder
  $p(Z)$: prior
- observe a sample x

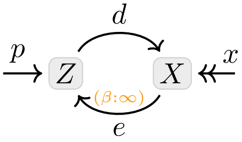


Variational Auto-Encoders, Take 2

- Structure:

  $e(Z \mid X)$: encoder
  $d(X \mid Z)$: decoder
  $p(Z)$: prior

- observe a sample x
  - and trust encoding



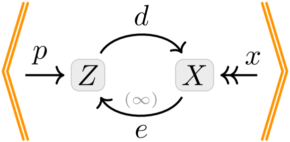
Variational Auto-Encoders, Take 2

- Structure:

  $e(Z \mid X)$: encoder
  $d(X \mid Z)$: decoder
  $p(Z)$: prior

- observe a sample x
  - and trust encoding

Objective function is free:



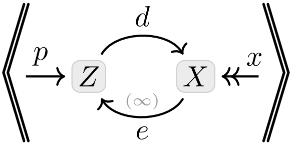
$= \mathrm{ELBO}_{p,e,d}(x)$

Variational Auto-Encoders, Take 2

- Structure:

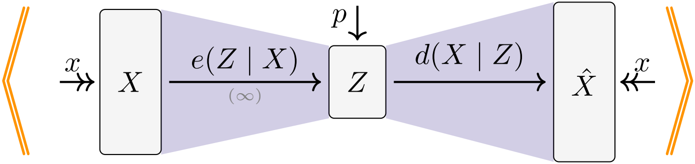
  $e(Z \mid X)$: encoder
  $d(X \mid Z)$: decoder
  $p(Z)$: prior
- observe a sample x
  - and trust encoding

Objective function is free:



$= \mathrm{ELBO}_{p,e,d}(x)$

$=$



A Very Useful Fact

Believing more things can't make you any less inconsistent.

Lemma (monotonicity of inconsistency)

For all pdgs $\mathcal{M}$, $\mathcal{M}'$, *and all* $\gamma > 0$,

1. $\langle\!\langle \mathcal{M} \sqcup \mathcal{M}' \rangle\!\rangle_\gamma \geq \langle\!\langle \mathcal{M} \rangle\!\rangle_\gamma$.
2. *If* $\mathcal{M}$ *and* $\mathcal{M}'$ *have respective confidence vectors* β *and* β', *and* $\beta \succeq \beta'$ *(that is,* $\beta_L \geq \beta'_L$ *for all* $L \in \mathcal{E}$*), then* $\langle\!\langle \mathcal{M} \rangle\!\rangle_\gamma \geq \langle\!\langle \mathcal{M}' \rangle\!\rangle_\gamma$.

Visual Proof: The Varitaional Bound

Visual Proof: The Varitaional Bound

$$\left\langle\!\!\left\langle \begin{tikzcd} {} \arrow[r, "p"] & Z \arrow[r, bend left, "d"] & X \arrow[l, bend left, "e!"] & {} \arrow[l, two heads, "x"'] \end{tikzcd} \right\rangle\!\!\right\rangle = -\,\mathrm{ELBO}_{p,e,d}(x).$$

Visual Proof: The Varitaional Bound

$$-\log \Pr_{p,d}(X{=}x) \;=$$

$$\left\langle\!\!\left\langle \xrightarrow{p} \boxed{Z} \xrightarrow{d} \boxed{X} \overset{x}{\twoheadleftarrow} \right\rangle\!\!\right\rangle \qquad \left\langle\!\!\left\langle \xrightarrow{p} \boxed{Z} \underset{e!}{\overset{d}{\rightleftarrows}} \boxed{X} \overset{x}{\twoheadleftarrow} \right\rangle\!\!\right\rangle$$

$$= \; -\mathrm{ELBO}_{p,e,d}(x).$$

Visual Proof: The Varitaional Bound

$$-\log \Pr_{p,d}(X\!=\!x) \;=\; \left\langle\!\!\left\langle \xrightarrow{p} Z \xrightarrow{d} X \overset{x}{\twoheadleftarrow} \right\rangle\!\!\right\rangle \;\le\; \left\langle\!\!\left\langle \begin{tikzcd} {} \arrow[r, "p"] & Z \arrow[r, bend left, "d"] & X \arrow[l, bend left, "e!"] & {} \arrow[l, two heads, "x"'] \end{tikzcd} \right\rangle\!\!\right\rangle \;=\; -\,\mathrm{ELBO}_{p,e,d}(x).$$

Divergences and Inconsistency

You believe both $p(X)$ and $q(X)$.

$$\xrightarrow{p} \boxed{X} \xleftarrow{q}$$

Divergences and Inconsistency

You believe both $p(X)$ and $q(X)$.
Your inconsistency: a divergence between p and q?

$$\left\langle\!\!\left\langle \xrightarrow{p} \boxed{X} \xleftarrow{q} \right\rangle\!\!\right\rangle$$

Divergences and Inconsistency

You believe both $p(X)$ and $q(X)$.
Your inconsistency: a divergence between p and q?

$$\left\langle\!\!\left\langle \xrightarrow[(\beta:r)]{p} \boxed{X} \xleftarrow[(\beta:s)]{q} \right\rangle\!\!\right\rangle$$

Divergences and Inconsistency

You believe both $p(X)$ and $q(X)$.
Your inconsistency: a divergence between p and q?

$$\text{Let } \boldsymbol{D}^{\text{PDG}}_{(r,s)}(p,q) := \left\langle\!\!\left\langle \xrightarrow[(\beta:r)]{p} \boxed{X} \xleftarrow[(\beta:s)]{q} \right\rangle\!\!\right\rangle$$

Divergences and Inconsistency

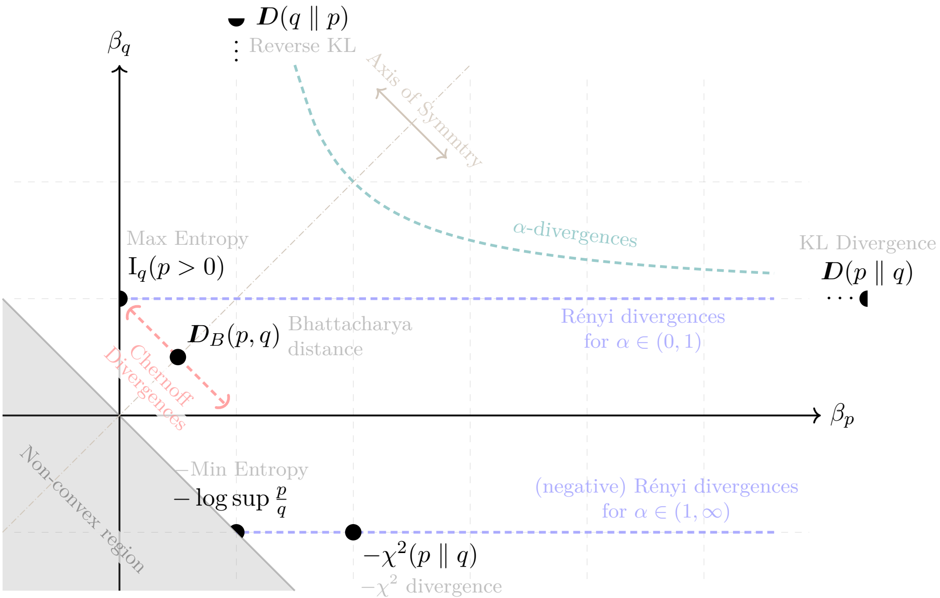
You believe both $p(X)$ and $q(X)$.
Your inconsistency: a divergence between p and q?

Let $\boldsymbol{D}^{\text{PDG}}_{(r,s)}(p,q) := \left\langle\!\!\left\langle \xrightarrow[(\beta:r)]{p} \boxed{X} \xleftarrow[(\beta:s)]{q} \right\rangle\!\!\right\rangle$

Lemma

$$\boldsymbol{D}^{\text{PDG}}_{(r,s)}(p,q) = -(r+s)\log \sum_x \Big(p(x)^r q(x)^s \Big)^{\frac{1}{r+s}} .$$

Divergences as Inconsistencies



Visual Proof: Data Processing Inequality

$$\boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(p \,\Big\|\, q\Big) \geq \boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(f \circ p \,\Big\|\, f \circ q\Big)$$

$$\left\langle\!\!\left\langle \xrightarrow[(\beta)]{p} \boxed{X} \xleftarrow[(\zeta)]{q} \right\rangle\!\!\right\rangle$$

$$\left\langle\!\!\left\langle \xrightarrow[(\beta)]{f \circ p} \boxed{X} \xleftarrow[(\zeta)]{f \circ q} \right\rangle\!\!\right\rangle$$

Visual Proof: Data Processing Inequality

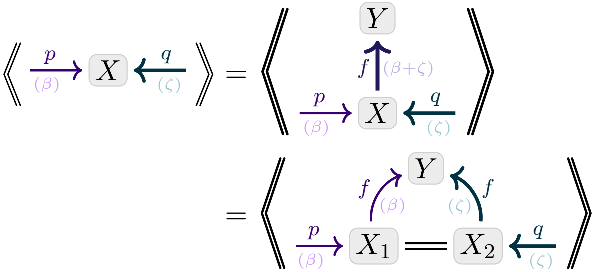
$$\boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(p \,\Big\|\, q\Big) \geq \boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(f \circ p \,\Big\|\, f \circ q\Big)$$

$$\left\langle\!\!\left\langle \xrightarrow[(\beta)]{p} \boxed{X} \xleftarrow[(\zeta)]{q} \right\rangle\!\!\right\rangle = \left\langle\!\!\left\langle \begin{array}{c} \boxed{Y} \\ f \Big\uparrow (\beta+\zeta) \\ \xrightarrow[(\beta)]{p} \boxed{X} \xleftarrow[(\zeta)]{q} \end{array} \right\rangle\!\!\right\rangle$$

$$\left\langle\!\!\left\langle \xrightarrow[(\beta)]{f\circ p} \boxed{X} \xleftarrow[(\zeta)]{f\circ q} \right\rangle\!\!\right\rangle$$

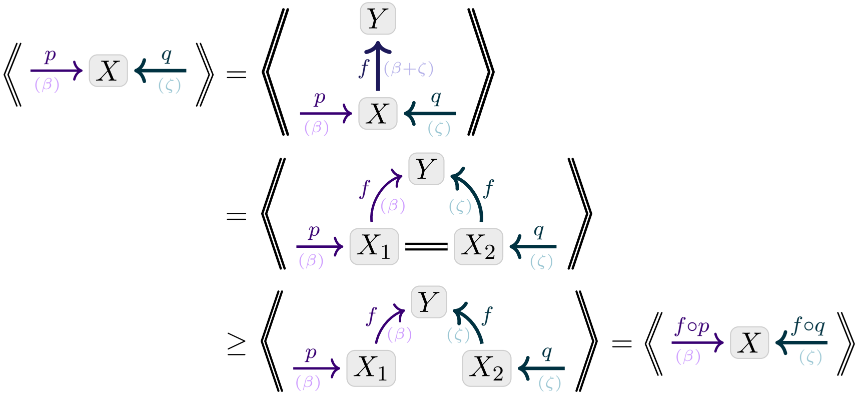
Visual Proof: Data Processing Inequality

$$\boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(p \,\Big\|\, q\Big) \geq \boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(f \circ p \,\Big\|\, f \circ q\Big)$$



Visual Proof: Data Processing Inequality

$$\boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(p \,\Big\|\, q\Big) \geq \boldsymbol{D}^{\mathrm{PDG}}_{(\beta,\zeta)}\Big(f \circ p \,\Big\|\, f \circ q\Big)$$



Outline for Section 8

1. Introduction
2. Modeling Examples
   - A Simple Example: What are Floomps?
   - Differences from BNs
   - PDG Union and Restriction
3. Syntax
   - Formal Definitions of PDGs
4. Semantics
5. Capturing other Graphical Models
   - Bayesian Networks
   - Factor Graphs
6. Inference
7. Inconsistency as Loss
   - Motivation
   - Standard Metrics as Inconsistency
   - Variational AutoEncoders
   - Inconsistency and Statistical Divergences
8. Other Aspects of PDGs
   - Databases
   - Open Problems + Future Work

Database

$\mathbf{R}_1$

A	B	C
a_1	b_1	c_1
a_2	b_2	c_2

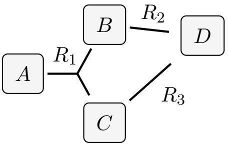
$\mathbf{R}_2$

B	D
b_2	d_1
b_3	d_2
b_4	d_3

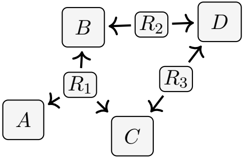
$\mathbf{R}_3$

C	D
c_2	d_1
c_1	d_3

Relational Schema



Row-PDG



Proposition

If $\mathcal{D}$ is a database and μ is a joint distribution over $\mathcal{M}_{\mathcal{D}}$, then $\mu \in \{\!\!\{\mathcal{M}_{\mathcal{D}}\}\!\!\}$ iff $\mathit{Supp}(\mu)$ is a universal relation for $\mathcal{D}$.

Corollary

$\mathcal{M}_{\mathcal{D}}$ is consistent iff $\mathcal{D}$ is join consistent.

Open Problems and Future Work

- Technical tool: PDGs with incomplete cpds
- Encoding preferences, and understanding preference changes
- Trace Semantics: and the probabilistic automaton generated by a PDG
- $**$ Do PDGs capture Dependency Networks? $**$
- Multi-agent systems

Technical Recap

⟨ `update with second half` ⟩

PDGs...

- capture inconsistency, including conflicting information from multiple sources with varying reliability.

Technical Recap

⟨ `update with second half` ⟩

PDGs…

- capture inconsistency, including conflicting information from multiple sources with varying reliability.
- are especially modular; to combine info from two sources, simply take a PDG union. This incorporates new data (edge cpds) and concepts (nodes) without affecting previous information.

Technical Recap

⟨ `update with second half` ⟩

PDGs...

- capture inconsistency, including conflicting information from multiple sources with varying reliability.
- are especially modular; to combine info from two sources, simply take a PDG union. This incorporates new data (edge cpds) and concepts (nodes) without affecting previous information.
- cleanly separate quantitative info (the cpds) from qualitative info (the edges), with variable confidence in both (the weights β and α). This is captured by terms *Inc* and *IDef* in our scoring function.

Technical Recap

⟨ `update with second half` ⟩

PDGs...

- capture inconsistency, including conflicting information from multiple sources with varying reliability.
- are especially modular; to combine info from two sources, simply take a PDG union. This incorporates new data (edge cpds) and concepts (nodes) without affecting previous information.
- cleanly separate quantitative info (the cpds) from qualitative info (the edges), with variable confidence in both (the weights β and α). This is captured by terms *Inc* and *IDef* in our scoring function.
- have (several) natural semantics; one of them allows us to pick out a unique distribution. Using this distrbution, PDGs can capture BNs and factor graphs.

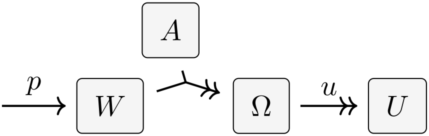
Technical Recap

$\langle$ `update with second half` $\rangle$

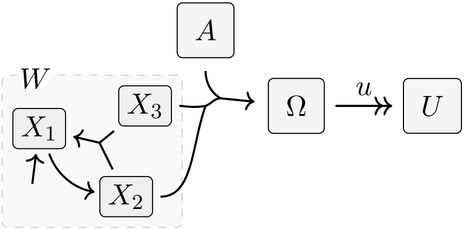
PDGs...

- capture inconsistency, including conflicting information from multiple sources with varying reliability.
- are especially modular; to combine info from two sources, simply take a PDG union. This incorporates new data (edge cpds) and concepts (nodes) without affecting previous information.
- cleanly separate quantitative info (the cpds) from qualitative info (the edges), with variable confidence in both (the weights β and α). This is captured by terms *Inc* and *IDef* in our scoring function.
- have (several) natural semantics; one of them allows us to pick out a unique distribution. Using this distrbution, PDGs can capture BNs and factor graphs.

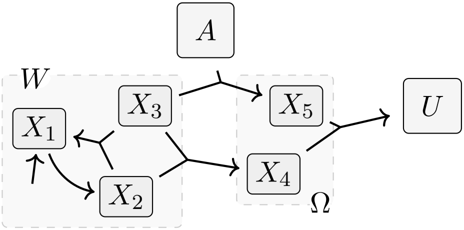
But there is much more to be done!



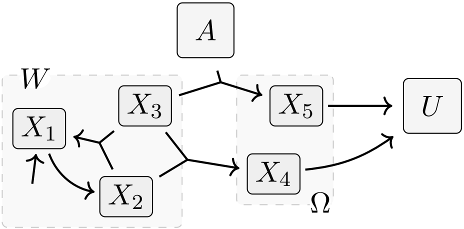
- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not "favorite number go up"



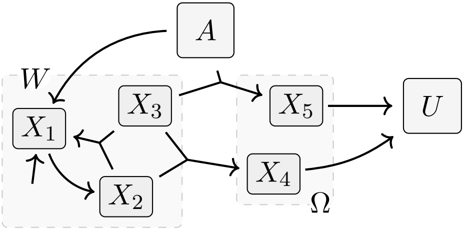
- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not “favorite number go up”



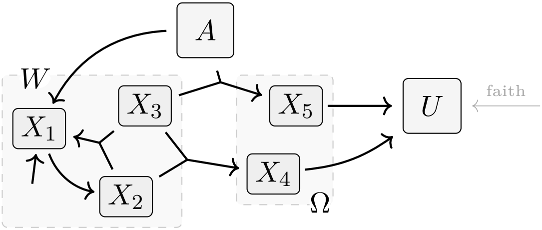
- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not “favorite number go up”



- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not "favorite number go up"



- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not "favorite number go up"



- decompose states and beliefs (like PGMs)
- driven by pursuit of coherent identity, not “favorite number go up”

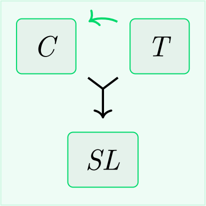
Outline for Section 9

9 Hyper-graphs

10 The Information Deficiency

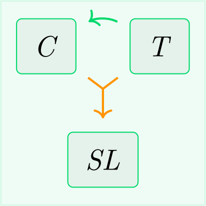
11 Category Theory
- PDGs as diagrams of the Markov Category

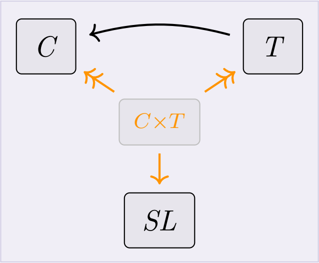
Hyper-graphs? Or merely graphs?



main definition

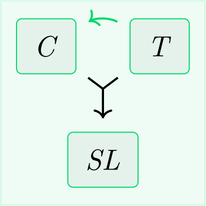
Hyper-graphs? Or merely graphs?

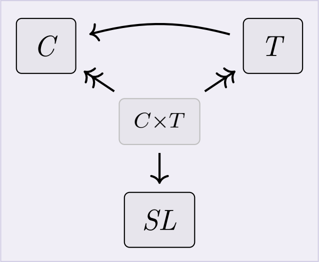




main definition

Hyper-graphs? Or merely graphs?

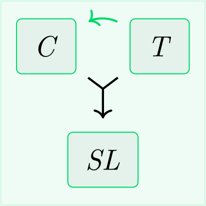


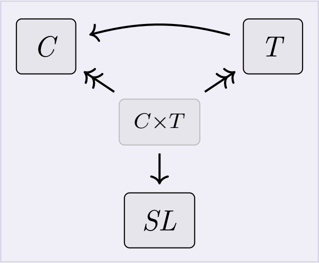


- This widget expands state space, but graphs are simpler.

main definition

Hyper-graphs? Or merely graphs?



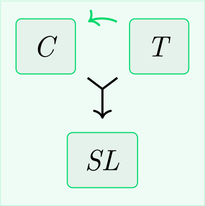


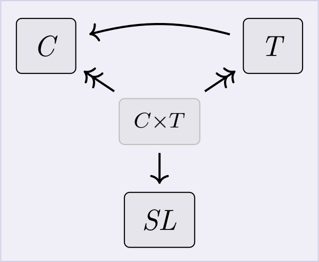
- This widget expands state space, but graphs are simpler.
- There is a natural correspondence

joint distributions $\leftrightarrows$ expanded joint distributions satisfying coherence constraints

main definition

Hyper-graphs? Or merely graphs?





- This widget expands state space, but graphs are simpler.
- There is a natural correspondence

joint distributions $\leftrightarrows$ expanded joint distributions satisfying coherence constraints

(working directly with hypergraphs is also possible)

main definition

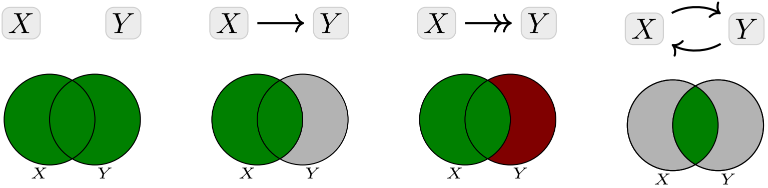
Outline for Section 10

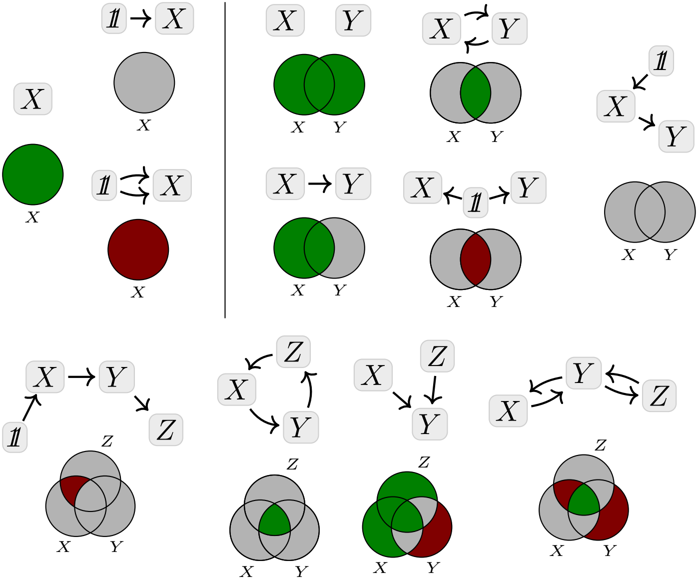
9 Hyper-graphs

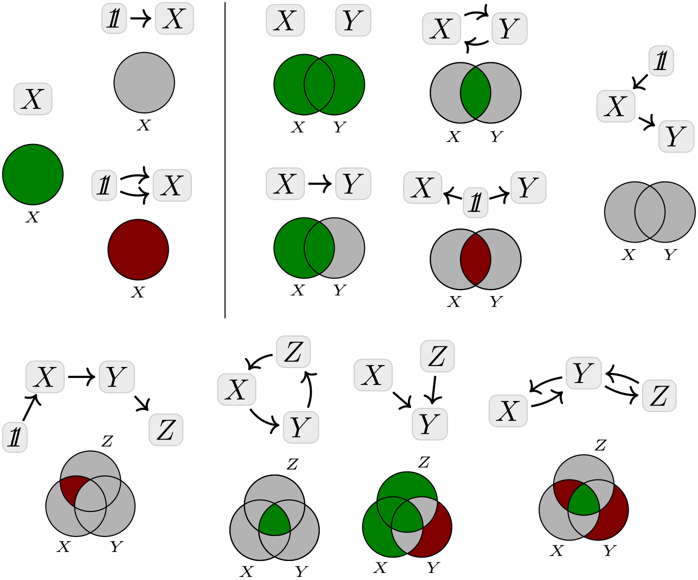
10 The Information Deficiency

11 Category Theory
- PDGs as diagrams of the Markov Category

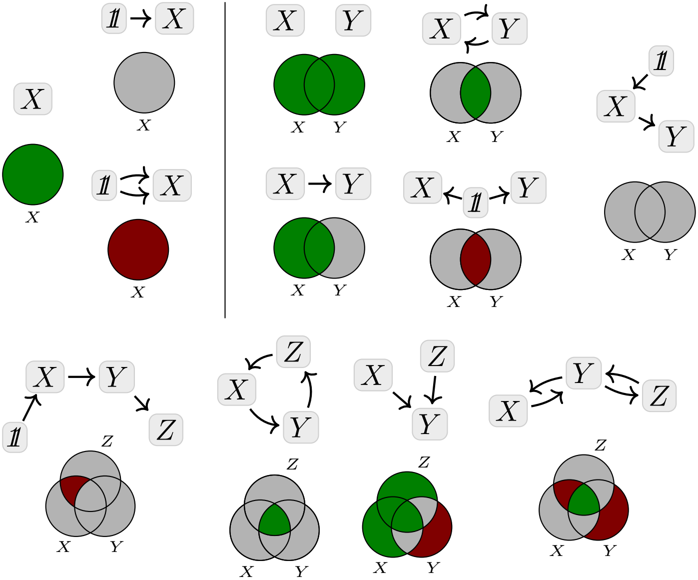
ILLUSTRATIONS OF *IDef*

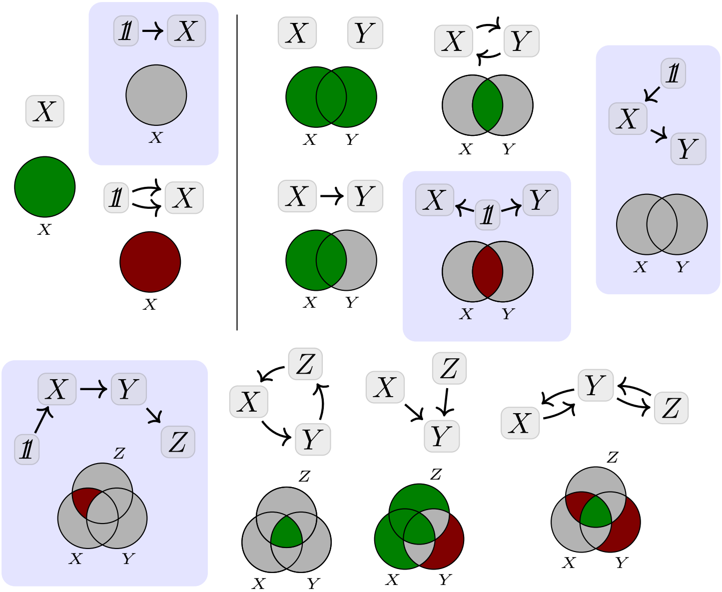



X
𝟙 → X
X
𝟙 ⇉ X
X
X Y
X Y
X → Y
X Y
X ⇄ Y
X Y
X ← 𝟙 → Y
X Y
𝟙 → X → Y
X Y
𝟙 → X → Y → Z
Z
X Y
Z → X → Y → Z
Z
X Y
X → Y ← Z
Z
X Y
X ⇄ Y ⇄ Z
Z
X Y



X
X
𝟙 → X
X
𝟙 ⇉ X
X
X Y
X Y
X ⇄ Y
X Y
X → Y
X Y
X ← 𝟙 → Y
X Y
𝟙 → X → Y
X Y
𝟙 → X → Y → Z
Z X Y
Z → X → Y → Z
Z X Y
X → Y ← Z
Z X Y
X ⇄ Y ⇄ Z
Z X Y


X
Y
Z
𝟙



X
X
𝟙 → X
X
𝟙 X
X
X Y
X Y
X Y
X Y
X → Y
X Y
X 𝟙 Y
X Y
𝟙
X
Y
X Y
X → Y
𝟙
Z
Z
X Y
Z
X
Y
Z
X Y
X
Z
Y
Z
X Y
Y
X
Z
Z
X Y

Outline for Section 11

9 Hyper-graphs

10 The Information Deficiency

11 Category Theory
- PDGs as diagrams of the Markov Category

Definition (PDG)

$$
\begin{array}{lr}
\mathcal{N} : \textbf{Set} & \text{(node set)} \\
\mathcal{V} : \mathcal{N} \to \textbf{Set} & \text{(node values)} \\
\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label} & \text{(edge set)} \\
\text{For } X \xrightarrow{L} Y \in \mathcal{E}, & \\
\quad \mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y) & \text{(edge cpd)} \\
\quad \alpha_L : \mathbb{R} & \text{(functional determination)} \\
\quad \beta_L : \mathbb{R} & \text{(cpd confidence)}
\end{array}
$$

- $(\mathcal{N}, \mathcal{V})$ is a set of variables

Definition (PDG)

$\mathcal{N}$: **Set** (node set)

$\mathcal{V} : \mathcal{N} \to \mathbf{Set}$ (node values)

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label}$ (edge set)

For $X \xrightarrow{L} Y \in \mathcal{E}$,

$\mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y)$ (edge cpd)

$\alpha_L : \mathbb{R}$ (functional determination)

$\beta_L : \mathbb{R}$ (cpd confidence)

- $(\mathcal{N}, \mathcal{V})$ is a set of variables
- $(\mathcal{N}, \mathcal{E})$ is a multigraph

Definition (PDG)

$\mathcal{N}$: **Set** (node set)

$\mathcal{V} : \mathcal{N} \to \mathbf{Set}$ (node values)

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label}$ (edge set)

For $X \xrightarrow{L} Y \in \mathcal{E}$,

$\mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y)$ (edge cpd)

$\alpha_L : \mathbb{R}$ (functional determination)

$\beta_L : \mathbb{R}$ (cpd confidence)

- $(\mathcal{N}, \mathcal{V})$ is a set of variables
- $(\mathcal{N}, \mathcal{E})$ is a multigraph
- $(\mathcal{N}, \mathcal{E}, \alpha)$, the qualitative data, forms a weighted multigraph.

Definition (PDG)

$\mathcal{N}$: **Set** (node set)

$\mathcal{V} : \mathcal{N} \to \textbf{Set}$ (node values)

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label}$ (edge set)

For $X \xrightarrow{L} Y \in \mathcal{E}$,

$\mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y)$ (edge cpd)

$\alpha_L : \mathbb{R}$ (functional determination)

$\beta_L : \mathbb{R}$ (cpd confidence)

- $(\mathcal{N}, \mathcal{V})$ is a set of variables
- $(\mathcal{N}, \mathcal{E})$ is a multigraph
- $(\mathcal{N}, \mathcal{E}, \alpha)$, the qualitative data, forms a weighted multigraph.
- We call $(\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p})$ an *unweighted* PDG
  - and give it semantics as though $\alpha_L = \beta_L = 1$.

Definition (PDG)

$\mathcal{N}$: **Set** (node set)

$\mathcal{V} : \mathcal{N} \to$ **Set** (node values)

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label}$ (edge set)

For $X \xrightarrow{L} Y \in \mathcal{E}$,

$\mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y)$ (edge cpd)

$\alpha_L : \mathbb{R}$ (functional determination)

$\beta_L : \mathbb{R}$ (cpd confidence)

- $(\mathcal{N}, \mathcal{V})$ is a set of variables
- $(\mathcal{N}, \mathcal{E})$ is a multigraph
- $(\mathcal{N}, \mathcal{E}, \alpha)$, the qualitative data, forms a weighted multigraph.
- We call $(\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p})$ an *unweighted* PDG
  - and give it semantics as though $\alpha_L = \beta_L = 1$.

Let **Mark** be the category of measurable spaces and Markov kernels.

Definition (PDG)

$\mathcal{N}$: **Set** (node set)

$\mathcal{V} : \mathcal{N} \to$ **Set** (node values)

$\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N} \times \mathit{Label}$ (edge set)

For $X \xrightarrow{L} Y \in \mathcal{E}$,

$\mathbf{p}_L : \mathcal{V}(X) \to \Delta\mathcal{V}(Y)$ (edge cpd)

$\alpha_L : \mathbb{R}$ (functional determination)

$\beta_L : \mathbb{R}$ (cpd confidence)

- $(\mathcal{N}, \mathcal{V})$ is a set of variables
- $(\mathcal{N}, \mathcal{E})$ is a multigraph
- $(\mathcal{N}, \mathcal{E}, \alpha)$, the qualitative data, forms a weighted multigraph.
- We call $(\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p})$ an *unweighted* PDG
  - and give it semantics as though $\alpha_L = \beta_L = 1$.

Let **Mark** be the category of measurable spaces and Markov kernels.

Equivalent Categorical Definition

An unweighted PDG is a functor $\langle \mathbf{p}, \mathcal{V} \rangle : \mathit{Paths}(\mathcal{N}, \mathcal{E}) \to$ **Mark**.
So a PDG is a *diagram* in **Mark**, in the usual mathematical sense.

What do you do with diagrams? Take limits / colimits.

$$\cdots \qquad X_1 \twoheadrightarrow X_2 \leftarrow X_3 \qquad \cdots$$

What do you do with diagrams? Take limits / colimits.

$$\begin{tikzcd}
 & & L \arrow[dl] \arrow[d] \arrow[dr] & & \\
\cdots & X_1 \arrow[r, two heads] & X_2 & X_3 \arrow[l] & \cdots
\end{tikzcd}$$

What do you do with diagrams? Take limits / colimits.

$$\begin{tikzcd}
& & V \arrow[dl, "\forall"'] \arrow[d] \arrow[dr] & L \arrow[dll] \arrow[dl] \arrow[d] & \\
\cdots & X_1 \arrow[r, two heads] & X_2 & X_3 \arrow[l] & \cdots
\end{tikzcd}$$

What do you do with diagrams? Take limits / colimits.

$$
\begin{tikzcd}
 & & V \arrow[dl, "\forall"'] \arrow[d] \arrow[dr] \arrow[r, dashed, "\exists!"] & L \arrow[dll] \arrow[dl] \arrow[d] & \\
\cdots & X_1 \arrow[r, two heads] & X_2 & X_3 \arrow[l] & \cdots
\end{tikzcd}
$$

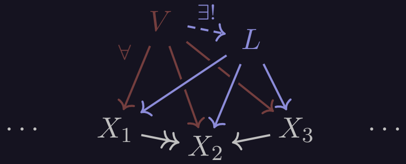
What do you do with diagrams? Take limits / colimits.

$$\begin{tikzcd}
& & V \arrow[r, dashed, "\exists!"] \arrow[dl, "\forall"'] \arrow[d] \arrow[dr] & \Omega \cong X_1 X_3 \arrow[dll] \arrow[dl] \arrow[d] & \\
\cdots & X_1 \arrow[r, two heads] & X_2 & X_3 \arrow[l] & \cdots
\end{tikzcd}$$

For the deterministic sub-PDG $\mathcal{m}_{\text{det}} \subseteq \mathcal{m}$:

$$\lim \mathcal{m}_{\text{det}} = \left(\begin{matrix}\text{natural}\\ \text{sample space}\end{matrix}\ \Omega,\quad \begin{matrix}\text{random}\\ \text{variables}\end{matrix}\ \left\{ \tilde{X} : \Omega \to \mathcal{V}(X) \right\}_{X \in \mathcal{N}} \right)$$

What do you do with diagrams? Take limits / colimits.

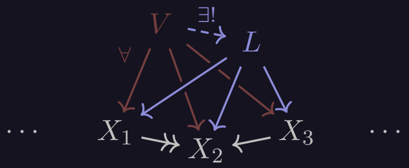


For the deterministic sub-PDG $\mathcal{m}_{\text{det}} \subseteq \mathcal{m}$:

$$\lim \mathcal{m}_{\text{det}} = \left(\begin{matrix}\text{natural} \\ \text{sample space}\end{matrix}\ \Omega, \quad \begin{matrix}\text{random} \\ \text{variables}\end{matrix}\ \left\{ \tilde{X} : \Omega \to \mathcal{V}(X) \right\}_{X \in \mathcal{N}} \right)$$

In general: $$\lim \mathcal{m} = \Big(\text{Verts}(\underbrace{\mathbb{L}\mathcal{m}}_{\substack{\text{Locally Consistent Polytope} \\ \text{(possible states of the Sum-Product algorithm)}}}), \quad \{ \text{variable marginals} \} \Big)$$

What do you do with diagrams? Take limits / colimits.



For the deterministic sub-PDG $\mathcal{m}_{\text{det}} \subseteq \mathcal{m}$:

$$\lim \mathcal{m}_{\text{det}} = \left(\begin{matrix}\text{natural}\\ \text{sample space}\end{matrix}\ \Omega,\quad \begin{matrix}\text{random}\\ \text{variables}\end{matrix}\ \left\{ \tilde{X} : \Omega \to \mathcal{V}(X) \right\}_{X \in \mathcal{N}} \right)$$

In general:

$$\lim \mathcal{m} = \Big(\text{Verts}(\underbrace{\mathbb{L}\mathcal{m}}_{\substack{\text{Locally Consistent Polytope}\\ \text{(possible states of the Sum-Product algorithm)}}}),\quad \{ \text{variable marginals} \} \Big)$$

For a BN $\mathcal{B}$:

$$\lim \mathcal{m}_{\mathcal{B}} = \left(\mathbb{1},\ \left\{ \Pr_{\mathcal{B}}(X) \right\}_{X \in \mathcal{N}} \right)$$