Dependency Graphs

Oliver Richardson, oli@cs.cornell.edu

January 18, 2020

Abstract

Graphical models have enjoyed substantial success in compressing probability distributions over joint settings of random variables, by encoding correlations locally in links, and making use of independence assumptions elsewhere. Still, due to the fact that they are committed to representing a single consistent distribution, they are not expressive enough to represent common mental states of agents who may not have a perfectly logical coherent . Unrelatedly, the process of modifying graphical models by adding or removing nodes/links can be quite expensive and changes the underlying space, making it more difficult to make use of the intuitive modularity that they seem to offer.

We introduce Probabilistic Dependency Graphs (PDGs) to combat these issues. PDGs are like unshackled Bayesian nets, interpreted more locally and whose links are interpreted as conditional subdistributions. The result is a "graphical model" in a looser sense which may not always be either complete or consistent (from the perspective of a distribution), which can be easily combined with graph operations. We find that by solely altering the set of nodes and links and then running consistency reduction algorithm (one of which is a generalized version of belief propagation), we can recover important capabilities of Bayesian networks, such as the ability to estimate conditional and marginal distributions given observations, belief updating via Jeffrey's rule. Further afield, we discover that this process also naturally models simple learning algorithms, and acting according to a decision rule (see the other paper for more). Bayesian networks, and to some extent, factor graphs, can be seen as special cases of PDGs.

Contents

1 Introduction

Inconsistencies are bad. Unfortunately, they are difficult to avoid: reasonable people are often simultaneously unaware of any inconsistencies in their beliefs, and yet still think it probable that they are not entirely consistent.[1] Standard representations, including existing graphical models, are unable to represent such belief states. While we share the distaste for inconsistency, we nonetheless think it important to represent: in addition to modeling an overwhelmingly common feature of humans, the possibility of inconsistency also allows us to talk about intermediate stages of belief updating (section 6.1), provides a rationale for providing multiple justifications (corroborating evidence means more when there was a possibility of conflict instead; see **??**), and lets us recover standard algorithms such as belief propagation and conditioning on evidence as resolutions of inconsistency (section 7).

This seems very strange. It's not the fact that representing inconsistency lets us recover standard algorithms. Rather, despite modeling inconsistency, we can still recover standard algorithms. That is worth saying, but not here.

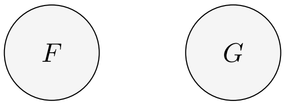
The graphical model we propose, which we call a Probabilistic Dependency Graph (PDG), also offers other advantages over more standard approaches. PDGs are more modular than Bayesian Networks, provide better locality than Factor Graphs (section 3.2), and more; for a complete list, see section 9.1. The rest of this section consists of examples focusing on the benefits of representing inconsistency and improved modularity, and the failure of alternatives to quite capture what we have in mind.

Say it here instead

Example 1. You have arrived in a foreign country well-known for having very clear laws. From prior reading, you have subjective probability 0.95 that owning guns is against the law. Upon landing, you end up talking to some teenagers who use the local slang—after which you believe with 10% probability that the law prohibits floomps.

Let's try to represent this as a Bayesian Network. Make a graph with two nodes: let F be the binary variable (taking values $\{f, \overline{f}\}$) indicating the legality of floomps, and G (taking values $g, \overline{g}$) indicate the legality of guns. Given the semantics of a Bayes Net, this give us two choices: assume that the two are independent, or choose a direction and make up some numbers to put in the table. As there is no reason to believe that either variable depends on the other, we don't put any links between them. Here is the resulting "network":

f	$\overline{f}$
.9	.1



g	$\overline{g}$
.05	.95

[1] In some sense, this is the reason arguments exist: it is possible to get a person to agree to premises and reject a conclusion, revealing inconsistency—inconsistency which can then be used to change someone else's mental state.

You later discover that “floomp” is likely to be another word for gun, and come to believe that if floomps are legal (resp. illegal), then there’s a 92% chance guns are as well; let E be the CPT associated to this realization. A first reaction might be to simply add this conditional information by adding F as a parent of G and incorporating E into the Bayes Net, which amounts to throwing out our old prior distribution on G; another instinct might be to perform a Bayesian update, but this new information is conditional and probabilistic, not an event. Note the following features of this scenario:

please cut this; it really doesn’t help.

Cut this too; less is more.

1. The table E conflicts with the other two, in the sense that there is no joint distribution on $\{F, G\}$ such that all three tables are accurate. Therefore, there is no Bayes Net that contains all three pieces of information exactly: one has to resolve the inconsistency first. There are three different dimensions along which this could be resolved: rejecting either prior belief, or E; in general any mixture would do.

   Cut this; it distracts from your point.

2. It may not be in your best interest to sort this out right away. The choice of resolution may be clearer if you can get confirmation that guns are indeed floomps, or read the laws more carefully—it may be worth sitting on the inconsistency so that you can resolve it properly later, rather than resolving it as best you can immediately.

   This is not a “feature of the scenario” (althoug it’s worth saying).

By contrast, consider the corresponding PDG. In a PDG, the conditional probability tables are attached to edges, rather than nodes of the graph. We will represent the conditional probability tables for links $L = A \to B$ as matrices $\mathbf{L}$, whose element $\mathbf{L}_{a,b}$ at row a, column b is the conditional probability $\Pr(B = b \mid A = a)$. In order to represent unconditional probabilities cleanly, we introduce the *unit variable*, $\mathbf{1}$, or ‘true’, which always takes its unique value (which we call $\star$).

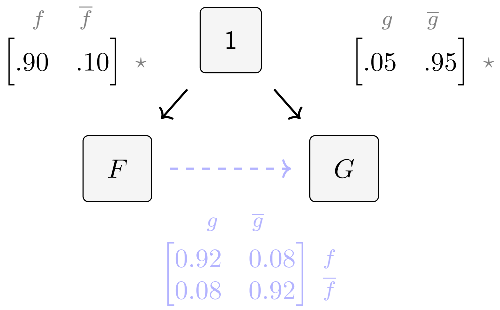

f $\overline{f}$
$\begin{bmatrix} .90 & .10 \end{bmatrix}$ $\star$
$\mathbf{1}$
g $\overline{g}$
$\begin{bmatrix} .05 & .95 \end{bmatrix}$ $\star$
F
G
g $\overline{g}$
$\begin{bmatrix} 0.92 & 0.08 \\ 0.08 & 0.92 \end{bmatrix}$ f $\overline{f}$

Figure 1: PDG representation of the inconsistent state before resolution

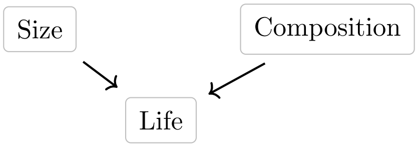
The original state of knowledge, consists of all three nodes and the two edges from $\mathbf{1}$, displayed in black and gray. This is like our earlier Bayes Net graph, except we no longer need to make the assumption up front that F and G are independent, so we don’t (we will do this later if necessary): we merely record the constraints imposed by the given probabilities.

E can be added directly to the diagram in the form of the light blue link from F to G without any issues. This does not change the meaning of the link from $\mathbf{1} \to G$, sidestepping issue 1: the added modularity lets us simply add the information, and resolve inconsistencies when convenient. Regarding point 2, we can stay in this conflicted state as long as necessary (again not even representable in other graphical models), and the operation is even reversible: all we need to do to recover our original belief state is delete the new edge. △

The ability to represent these ‘over-constrained’ states of belief, in which it is possible for there to be inconsistency, can be valuable even when there is none. Conversely, PDGs can represent under-constrained: any subgraph of a PDG is still a PDG, unlike for plain Bayesian Networks. Our next example illustrates both of these effects, while also highlighting an additional way in which BNs fail to be as modular as we might like.

Example 2. Suppose we have a belief about how size and composition affect the habitability of a planet: say we’re astrobiologists, and we have some sense of how likely we are to find life on a given planet, supposing we knew its size (big or small) and its composition (mostly made of rocks vs gas). That is to say, we have a conditional probability table:

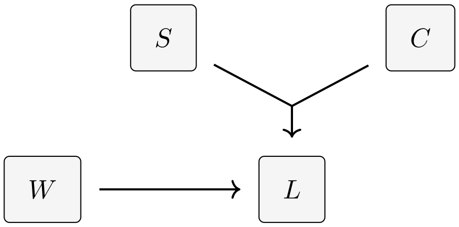
$$\Pr(\text{Life} \mid \text{Size}, \text{Comp}) = \begin{array}{cl} \begin{array}{cc} \text{life} & \text{no life} \end{array} & \\ \left[\begin{array}{cc} .1 & .9 \\ .2 & .8 \\ .05 & 0.95 \\ 0.00001 & 0.99999 \end{array}\right] & \begin{array}{l} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gaseous} \end{array} \end{array}$$



This one conditional probability table is a fragment of the Bayesian Network on the right, but in order to fully interpret the picture as a BN , we would also require distributions over the values of the root nodes 'Size' and 'Composition'—things we may not know anything about. Because many joint distributions are compatible with this table, we consider this data *under-constrained*. A proponent of BNs will have no trouble getting this to work by using *conditional* BNs, which represent conditional distributions rather than joint ones, with two slight inconveniences: (1) they would have to commit which variables are conditioned up-front, and so adding or removing information changes the type of mathematical object represented, and (2) there is no graphical distinction between having a distribution over Size or not, and so we have to color certain nodes and keep track of this additional information. PDGs have neither of these issues, but the bigger payoff is in representing *over-constrained* states.

Cut; less is more. It's a better story if you just cut this.

Now our biologist friend now reminds us that life requires water, and gives us a probability estimate for the existence of life on a planet, with and without water. We trust this friend completely, and totally believe these probabilities. Unfortunately, but there's no way to place it as the parent of the L node, because we don't know what the correlations are between water, size, and composition; neither are we prepared to give a probability of live given a full description of the three, and may not even have the space to keep such a thing.[2] Let S, C, W, L be shortenings of Size, Composition, Water, and Life, respectively. PDGs allow us to interpret each arrow individually, using a picture like this:



This means that we have two conditional probability tables on L: one from $S \times C$ and the other from W. We can now combine our two beliefs, without also requiring us to provide information about the correlations between W and $S \times C$ we do not have. As with the previous example, there is now a possibility of being inconsistent, in the sense that is possible to specify the conditional distributions in the links in such a way that no joint distribution on all variables will marginalize out to them — for instance, if all estimates of L from the W are strictly smaller than any probability estimate of L from $S \times C$. △

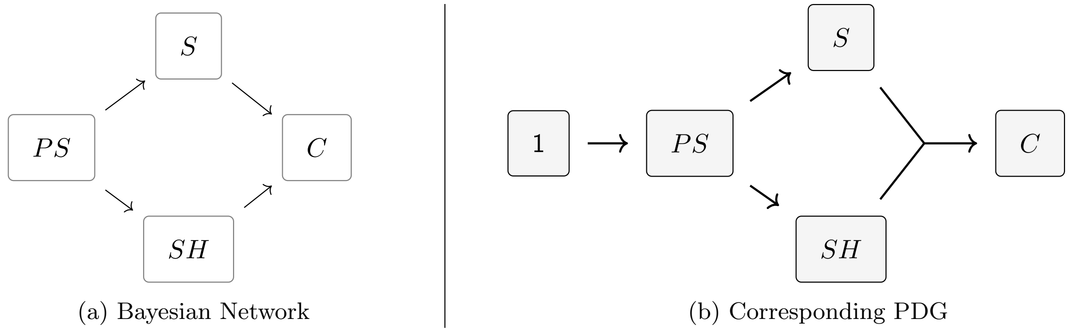
bad grammar

PDGs are strictly more expressive than Bayesian Networks, and there is a straightforward conversion from a BN to a PDG (section 3.1). From a technical perspective, the biggest difference between PDGs and BNs is the interpretation of two colliding arrows, as highlighted in example 3.

Example 3. Consider the classic example used to introduce Bayesian nets, in which the four variables are interest are booleans indicating whether a person (C) develops cancer, (S) smokes, (SH) is exposed to second hand smoke, and (PS) has parents who smoke, presented graphically in figure 2a.

The BN is a compact representation of a joint distribution over all four variables, which achieves compactness by taking advantage of independence between variables. It encodes an assumption that every node is

[2] If *Size* and *Composition* each had $\approx \sqrt{N}$ elements, and *Water* had $\approx N$ elements, it would be $O(N^2)$ to store a full joint table, compared to $O(N)$ for the two individual ones.



(a) Bayesian Network

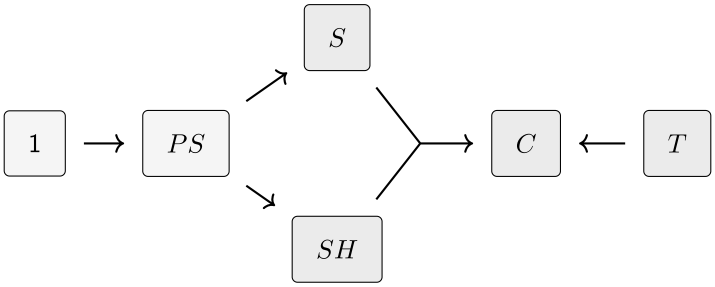
(b) Corresponding PDG

Figure 2: Both graphical models representing the conditional relationships in example 3

independent of its non-descendants given its parents. Most of the time, we do not make the independence assumption because we know for certain that the variables are independent; rather, we just suspect that the identified links are by much more important than the others. Determining for sure that smoking and second hand smoke are independent, controlling for parents' smoking habits, would extremely difficult, and to do properly would require much more empiricism to validate.

The PDG, on the other hand, represents merely the set of constraints on marginals given in the tables. Depending on our chosen semantics (we offer several in section 2), we can further interpret the constraints. For instance, by looking at the maximum entropy distribution consistent with the constraints, we get back the independence assumptions from the BN, and can thereby view Figure 2b as representing the same distribution as the BN.

Now, suppose you read a very thorough empirical study which demonstrates that people who use tanning beds have a 10% incidence of cancer, compared with 1% in the control. Just as in the previous example, this cannot be encoded directly into the Bayesian Network. The PDG on the other hand, has no trouble, and is simply the union of the two pieces of information:



Just as in the previous example, simply adding T to the parents of C would make it impossible to use our old table, and require us to guess at the interactions between tanning beds and smoking, which we have no information about. △

While other graphical representations (such as factor graphs) are also more modular, they do not represent inconsistency either, and raise their own issues (see section 3.2).

Thus far, we have given a taste of why PDGs could be valuable: they can represent under and over-constrained epistemic states, allow us to avoid making unnecessary assumptions, and are more modular, in the sense that changes to the underlying graph are much simpler and require fewer assumptions than for Bayesian Networks. However, we have only described parts of our simplest model; before we get to the more powerful variants, we need to formalize what we have.

2 Formal Definitions and Semantics

We have seen some examples of PDGs, but elided all of the details; we now begin a more careful treatment of these mathematical objects. Rather than representing a probability distribution, PDGs can be thought of as representing *constraints* on distributions.

You have to relate this to qualitative BNs (no more than 1-2 sentences)

Compared to a Bayseian Network, a PDG still consists of a directed graph, and the interpretations of the edges are still conditional probability tables, but now each edge is interpreted individually, rather than in groups with a shared target node. For instance, the graph $A \to C \leftarrow B$ would be interpreted as a single conditional probability table $\Pr(C \mid A, B)$ in a BN, but as two separate tables $\Pr(C \mid A)$ and $\Pr(B \mid A)$ in a PDG.

rewrite; much too fuzzy.

Definition 2.1. A *strict Probabilistic Dependency Graph* is a tuple $(\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ where

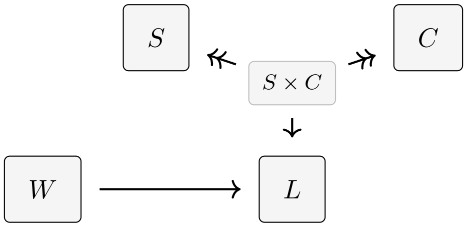
- $\mathcal{N}$ is a finite collection of nodes
- $\mathcal{L} \subseteq N \times N$ is a set of directed links between nodes
- $\mathcal{V} : \mathcal{N} \to \textbf{FinSet}$ is an N-indexed family of measurable sets, representing the values that a node can take
- $\boldsymbol{\mu} : ((A, B) : \mathcal{L}) \to \mathcal{V}(A) \to \Delta(\mathcal{V}(B))$ is a family of conditional probability distributions on $\mathcal{V}(B)$ indexed by the values of A for every link $(A, B) \in \mathcal{L}$

Just say this; let's be gentle.

The definition of $\boldsymbol{\mu}$ is probably more familiar than it looks. If every $\mathcal{V}(N)$ is finite with all subsets measurable, then $\boldsymbol{\mu}_{A,B}$ is just a conditional probability table, or a stochastic matrix. You may have noted that this only includes data for the normal arrows, rather than the multi-tailed ones described in the examples. This can be achieved either by generalizing $\mathcal{L}$ to higher orders, or introducing product nodes; we opt for the latter as it simplifies the formalism and paves the way for additional special nodes in **??**.

Cut this. Either make mutli-tailed arrows part of the syntax, or explain how they be captured.

Example 2 (continuing from p. 3)**.** Earlier in example 2, we displayed the arrow from S and C to L as a directed hyper-edge. While we would like to maintain this intuition, the way to do this is using only graphs with edges, as in definition 2.1, is to use intermediate nodes:



We will sometimes use double headed arrows like this to emphasize degenerate conditional distributions, which are deterministic. We can now present this PDG formally with the elements specified in definition 2.1.

$$\mathcal{N} = \{S,\ C,\ L,\ W,\ S \times C\}$$
$$\mathcal{L} = \{(S \times C, L), (W, L), (S \times C, S), (S \times C, C)\}$$
$$\mathcal{V} = \begin{cases} \mathcal{V}(S) & = \{\mathit{big}, \mathit{small}\} \\ \mathcal{V}(C) & = \{\mathit{rocky}, \mathit{gasseous}\} \\ \mathcal{V}(L) & = \{l, \neg l\} \\ \mathcal{V}(W) & = \{\mathit{none}, \mathit{some}, \mathit{mostly}\} \\ \mathcal{V}(S \times C) & = \mathcal{V}(S) \times \mathcal{V}(C) \end{cases}$$

$$\boldsymbol{\mu} = \begin{cases} \boldsymbol{\mu}[S \times C, L] = \begin{array}{ccl} l & \neg l & \\ .1 & .9 & \text{big, rocky} \\ .2 & .8 & \text{small,rocky} \\ .05 & 0.95 & \text{big, gasseous} \\ 0.00001 & 0.99999 & \text{small,gasseous} \end{array} & \boldsymbol{\mu}[W, L] = \begin{array}{ccl} l & \neg l & \\ 0 & 1 & \text{none} \\ .005 & .995 & \text{some} \\ .05 & 0.95 & \text{mostly} \end{array} \\ \boldsymbol{\mu}[S \times C, C] = \begin{array}{ccl} \text{rocky} & \text{gasseous} & \\ 1 & 0 & \text{big, rocky} \\ 1 & 0 & \text{small,rocky} \\ 0 & 1 & \text{big, gasseous} \\ 0 & 1 & \text{small,gasseous} \end{array} & \boldsymbol{\mu}[S \times C, S] = \begin{array}{ccl} \text{small} & \text{big} & \\ 0 & 1 & \text{big, rocky} \\ 1 & 0 & \text{small,rocky} \\ 0 & 1 & \text{big, gasseous} \\ 1 & 0 & \text{small,gasseous} \end{array} \end{cases}$$

△

Before we continue, we first need some notation:

Definition 2.2. If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG, let $W_\mathcal{V}$ denote the set of a-priori possible worlds, i.e., selections out of each $\mathcal{V}(N)$.

$$W_\mathcal{V} := \prod_{N \in \mathcal{N}} \mathcal{V}(N)$$

These graphs admit multiple semantics. We think of Probabilistic Dependency Graphs as being a representation of beliefs in and of themselves, rather than a compression of something more fundamental such as a probability distribution. Still, we will find it useful to interpret them in various ways: doing so will make it possible to compare them more directly with existing graphical models, which one thinks of as really just being compressed distributions. In this section, we would like to highlight three important semantics.

2.1 As Sets of Distributions

If the focus is on under-constrained models, then just as a BN represents a distribution on joint space, a PDG might be thought of as representing the set of all distributions that marginalize out to it exactly.

Definition 2.3. If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG, let $[\![M]\!]_{\textbf{Set}}$ be the set of distributions over the variables in M consistent with $\boldsymbol{\mu}$ on every marginal. Formally,

$$\left[\!\!\left[M\right]\!\!\right]_{\textbf{Set}} := \left\{ \mu \in \Delta[W_\mathcal{V}] \;\middle|\; \begin{array}{c} \mu(B = b \mid A = a) = \boldsymbol{\mu}[A, B](b \mid a) \\ \quad\quad \text{for all } A, B \in \mathcal{L},\ a \in \mathcal{V}(A), \text{ and } b \in \mathcal{V}(B) \end{array} \right\}$$

We can now finally give a formal definition of consistency.

Definition 2.4. M is consistent iff $[\![M]\!]_{\textbf{Set}} \neq \varnothing$.

Finally, we state the most important mathematical characterization of these objects

You have to motivate this. Why should anyone care? (I realize that you use it later, but people will be disappointed if this is the most important result.)

Proposition 2.1. *$[\![M]\!]_{\textit{Set}}$ is convex, for any PDG M.*

Proof. Choose any two distributions $p, q \in [\![M]\!]_{\textbf{Set}}$ consistent with M, any mixture coefficient $\alpha \in [0, 1]$, and any link $(A, B) \in \mathcal{L}$.

By the definition of $[\![M]\!]_{\textbf{Set}}$, we have $p(B = b \mid A = a) = q(B = b \mid A = a) = \boldsymbol{\mu}_{A,B}(a, b)$. For brevity, we will use little letters (a) in place of events $(A = a)$. Therefore, $p(a \wedge b) = \boldsymbol{\mu}_{A,B}(a, b)p(a)$ and $q(ab) = \boldsymbol{\mu}_{A,B}(a, b)q(a)$.

Some algebra reveals:

$$\begin{aligned}
\Big(\alpha p + (1-\alpha)q\Big)(B = b \mid A = a) &= \frac{\Big(\alpha p + (1-\alpha)q\Big)(b \wedge a)}{\Big(\alpha p + (1-\alpha)q\Big)(a)} \\
&= \frac{\alpha p(b \wedge a) + (1-\alpha)q(b \wedge a)}{\Big(\alpha p(a) + (1-\alpha)q(a)} \\
&= \frac{\alpha \boldsymbol{\mu}_{A,B}(a,b)p(a) + (1-\alpha)\boldsymbol{\mu}_{A,B}(a,b)q(a)}{\Big(\alpha p(a) + (1-\alpha)q(a)} \\
&= \boldsymbol{\mu}_{A,B}(a,b)\left(\frac{\alpha p(a) + (1-\alpha)q(a)}{\Big(\alpha p(a) + (1-\alpha)q(a)}\right) \\
&= \boldsymbol{\mu}_{A,B}(a,b)
\end{aligned}$$

and so the mixture $\Big(\alpha p + (1-\alpha)q\Big)$ is also contained in $[\![M]\!]_{\mathbf{Set}}$. □

2.2 As Weighted Distributions

We can also generalize the semantics in the previous section. For a candidate distribution μ, rather than being either consistent or not, we give it a continuous score.

Definition 2.5. The inconsistency, ζ, of a PDG $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ with respect to a distribution $p \in \Delta[W_{\mathcal{V}}]$ is

$$\zeta((\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu}); p) := \sum_{L=(A,B)\in\mathcal{L}} \mathop{\mathbb{E}}_{a\sim p(A)} \Big[D_{\mathrm{KL}}\Big(\boldsymbol{\mu}_L(a) \;\Big\|\; p(B \mid A = a)\Big)\Big]$$

where D_{KL} is the relative entropy, taken between the two distributions over B: one given by the link from $\boldsymbol{\mu}_{A,B}(a)$ and the other given by the marginal distribution of p conditioned on $A = a$, over the variable B.

Definition 2.6. If we think of the inconsistency as a potential energy, we can use something like a Boltzmann law to get likelihood over distributions. We therefore define

$$[\![M]\!]_\zeta := \mu \mapsto e^{-\zeta(M;\mu)}$$

This is not part of the definition, but part of the story (which needs to be fleshed out more). Why is this a reasonable score? You need much more motivation.

Proposition 2.2. *ζ is a convex function in p*

Proof. It is well-known that D_{KL} is convex, in the sense that

$$D_{KL}(\lambda q_1 + (1-\lambda)q_2 \mid\mid \lambda p_1 + (1-\lambda)p_2) \leq \lambda D_{KL}(q_1 \mid\mid p_1) + (1-\lambda)D_{KL}(q_2 \mid\mid p_2)$$

Choose any link $L \in \mathcal{L}$ from A to B, and also any $a \in \mathcal{V}(A)$. Setting $q_1 = q_2 = \boldsymbol{\mu}_L(a)$, we get

$$D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid \lambda p_1 + (1-\lambda)p_2) \leq \lambda D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_1) + (1-\lambda)D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_2)$$

Since this is true for every a and link, we can take a weighted sum of these inequalities for each a weighted by $p(A = a)$, and therefore

$$\begin{aligned}
\mathop{\mathbb{E}}_{a\sim p(A=a)} D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid \lambda p_1 + (1-\lambda)p_2) &\leq \mathop{\mathbb{E}}_{a\sim p(A=a)} \lambda D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_1) + (1-\lambda)D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_2) \\
\sum_{(A,B)\in\mathcal{L}} \mathop{\mathbb{E}}_{a\sim p(a)} D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid \lambda p_1 + (1-\lambda)p_2) &\leq \sum_{(A,B)\in\mathcal{L}} \mathop{\mathbb{E}}_{a\sim p(A)} \lambda D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_1) + (1-\lambda)D_{KL}(\boldsymbol{\mu}_L(a) \mid\mid p_2) \\
\zeta(M; \lambda p_1 + (1-\lambda)p_2) &\leq \lambda\zeta(M; p_1) + (1-\lambda)\zeta(M; p_2)
\end{aligned}$$

□

Conjecture 2.3. *$[\![M]\!]_\zeta$ is a quasiconvex function.*

Definition 2.7. The inconsistency of a PDG $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is the minimum value that could be achieved with any test distribution p; that is,

$$\zeta(M) = \inf_{\mu \in \Delta[W_\mathcal{V}]} \zeta(M;\mu) = -\log \sup_{p \in \Delta[W_\mathcal{V}]} [\![M]\!]_\zeta(p)$$

We may also consider the links as having different strengths: if we associate a positive "temperature" coefficient, to each edge, with higher numbers indicating higher uncertainty about the actual value.

Definition 2.8. A weighted PDG is a PDG $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ together with a temperature $T_L \in \mathbb{RP}^1$ for each link $L \in \mathcal{L}$.

Example 1 (continuing from p. 2)**.** Recall the PDG from our first example, in figure 1. Suppose that both the of the initial links $1 \to F$ and $1 \to G$ have temperature 1, and we give our new observation 10^{-3}.

Then $[\![M]\!]_\zeta$ △

2.3 As Distributions

To satisfy any lingering desire to compress all of the information to a single distribution, we also offer a way of interpreting a PDG as a single distribution.

Definition 2.9. If M is a PDG, $[\![M]\!]_{\mathbf{S}} \subseteq \Delta W_M$ is a set of distributions, $(D, \preceq)$ is an ordered set, and $\ell : \Delta W_M \to (D, \leq)$ is a scoring function for probabilities, let *the upper $\leq$-frontier under ℓ*, denoted $[\![M]\!]^\ell_{\mathbf{S}}$, be the set of distributions that are not strictly dominated by any others. More formally,

$$[\![M]\!]^\ell_{\mathbf{S}} = \left\{ \mu \in [\![M]\!]_{\mathbf{S}} \;\middle|\; \forall \mu' \in [\![M]\!]_{\mathbf{S}}.\ \ell(\mu') \preceq \ell(\mu) \right\}$$

One particularly useful scoring function is the following one, maximizing entropy:

$$\Big[\!\!\Big[M \Big]\!\!\Big]_{\overset{\uparrow}{\mathbf{H}}} := \Big[\!\!\Big[M \Big]\!\!\Big]^{-H(\cdot)}_{\mathbf{Set}}$$

Theorem 2.4. *If M is consistent, then $[\![M]\!]_{\overset{\uparrow}{\mathbf{H}}}$ contains a unique distribution; otherwise $[\![M]\!]_{\overset{\uparrow}{\mathbf{H}}}$ is empty.*

Proof. This is a direct consequence of proposition 2.1 □

This corresponds to a lexicographical ordering on *all* distributions (as opposed to simply the ones in $[\![M]\!]_{\mathbf{Set}}$), where we order first by satisfaction of constraints, and then by entropy. In general, we might be willing to relax the constraints a little, and sacrifice some fit for aa more general distribution. This allows us to define a free energy; see **??** for more.

3 Relations to Other Graphical Models

3.1 Bayesian Networks

A PDG can be seen as a generalization of a Bayesian Network in two important directions.

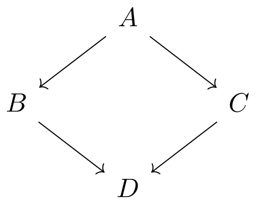
1. (1) In contrast with a Bayesian Network, in which each node has a set of parents, each node of a PDG has possibly many sets of parents, where each set of parents corresponds to a different constraint, associated to a different table, and (2) We no longer require conditional independence of non-descendants given children

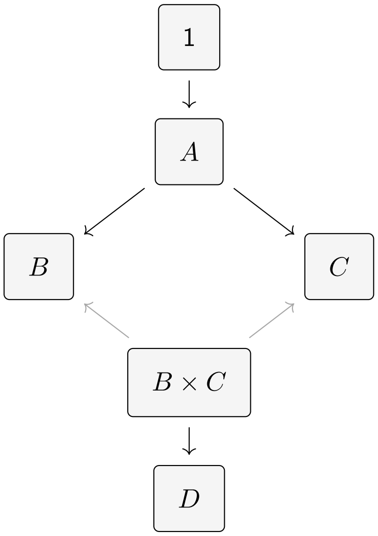
2. A BN is just a PDG where every cycle commutes

The semantics of a Baysian network ensure that there is no inconsistency: the arrows into a node taken together collectively determine a single well-defined probability distribution. Formally they consist of a set of nodes $\mathcal{N}$, and for each $N \in \mathcal{N}$, a set of parents $\mathrm{Par}(N)$, and a conditional probability distribution $\Pr(N \mid \mathrm{Par}(N))$, which is a distribution over the values of N for each setting of every variable in $\mathrm{Par}(N)$. While each of our arrows can be interpreted by itself as a marginal, a collection of arrows into a single node must be taken together to have any meaning in a BN.

The procedure for converting to a BN is simple: we simply take every node's incoming arrows, and insert the product of its parents as a node before it. With this procedure, if a node N has just one parent P, we replace the subgraph $[P \to N]$ with $[P \to N \overset{\sim}{\rightleftarrows} N]$, which is redundant so we don't draw this. If a node had zero parents (i.e., the BN just gives it a probability distribution not dependent on anything), we insert the product of zero things, i.e., the singleton node $\mathsf{1}$ (with $\mathcal{V}(\mathsf{1}) = \{\star\}$), and set $\Pr(N \mid *) = \Pr(N)$.

This sound much more complicated than it is. Consider the example below, where the left is a BN, and the right is the corresponding Probabilistic Dependency Graph.





We have effectively changed two things: first, visually encoded the probability distribution of A as the arrow $1 \to A$ (which we are now allowed to omit; sometimes you don't want priors on things, such as your own actions). Second, we have combined the two arrows $B \to D$ and $C \to D$ into a single one, $B \times C \to D$. Though certainly more verbose, this is arguably visually clearer if want to follow arrows: you cannot compute D from B; you need both B and C.

In order to fully get the joint representation given by the BN we would also need to make the final assumption that $B \perp\!\!\!\perp C \mid A$. This is possible to do with an extra arrow, but this solution doe not scale well and clutters the diagram. Instead, we will leave the picture as it is, and tackle the independence by using our maximum entropy semantics: the distribution encoded by the BN is the maximum entropy one encoded by the PDG.

3.1.1 Formalism

To show exactly how they correspond, we need a more formal definition of a Bayesian Network:

Definition 3.1. A Baysian network (BN) is a tuple

$$\mathcal{B} = \left(\mathcal{N} : \mathbf{FinSet}, \quad \mathrm{Par} : \mathcal{N} \to 2^{\mathcal{N}}, \quad \mathcal{S} : \mathcal{N} \to \mathbf{FinSet}, \quad \Pr : \prod_{N : \mathcal{N}} \left[\mathcal{S}_N \times \left(\prod_{P : \mathrm{Par}(N)} \mathcal{S}_P \right) \to [0,1] \right] \right)$$

such that

- the graph $\bigcup_{N,P \in \mathrm{Par}(N)} (N, P)$ is acyclic, i.e., there exists no cycle of nodes $N_0, N_1, \cdots, N_k = N_0$ in $\mathcal{N}^k$ such that $N_{i+1} \in \mathrm{Par}(N_i)$ for each $i \in \{0, 1, \cdots, k\}$.
- For all $N \in \mathcal{N}$, $\mathrm{Pr}(N)$ is a probability distribution on $\mathcal{S}_N$, i.e.,

$$\forall N \in \mathcal{N}.\ \forall \vec{p} \in \prod_{P:\mathrm{Par}(N)} \mathcal{S}_P. \quad \sum_{n \in \mathcal{S}_N} \underset{N}{\mathrm{Pr}}(\vec{p}, n) = 1$$

Definition 3.2. If $B = (\mathcal{N}, \mathrm{Par}, \mathcal{S}, \mathrm{Pr})$ is a Bayesian Network, then let $\Gamma(B)$ denote the corresponding PDG given by the procedure in section 3.1. Explicitly,

$$\Gamma\mathcal{B} := (\mathcal{N}', \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$$

where

$$\begin{aligned}
\mathcal{N}' &= \Big\{ \{N\} \mid N \in \mathcal{N} \Big\} \cup \{ \mathrm{Par}(N) \mid N \in \mathcal{N} \} \\
\mathcal{L} &= \Big\{ (\mathrm{Par}(N), \{N\}) \mid N \in \mathcal{N} \Big\} \cup \Big\{ (P, \{X\}) \mid X \in P, P = \mathrm{Par}(N) \text{ for some } N \in \mathcal{N} \Big\} \\
\mathcal{V}_N &= \prod_{X \in N} \mathcal{S}_X \\
\mathbf{p} &= \begin{cases} (\mathrm{Par}(N), \{N\}) & \mapsto \lambda(p, B).\ \sum_{b \in B} \mathrm{Pr}(b \mid p) \\ (P, X) & \mapsto, \lambda(p, B).\ \mathbb{1}_{\pi_X(p) \in B} \end{cases}
\end{aligned}$$

All we've done is explicitly add parent nodes and projection edges to our graph, and also subtly (by adding curly braces in the right places and taking unions rather than disjoint unions) eliminated the duplicate nodes arising from edges in the original BN which only have a single parent.

We can now state the following theorem, showing that Bayesian Networks are a special case of PDGs:

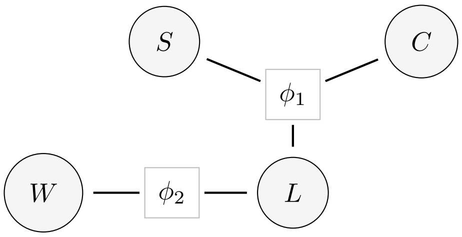
Theorem 3.1. *If B is a Bayesian Network, then* $[\![\Gamma(B)]\!]_{\overset{\uparrow}{\mathbf{H}}} = \{\mathrm{Pr}_B\}$.

3.2 Factor Graphs

Example 2 (continuing from p. 3)**.** Recall our discussion of life on an unknown planet. The probabilists in us might not be willing to so easily give up the notion that this data ought to define a probability distribution, at least implicitly. There are other, more general graphical models after all. Maybe there's something simple we can do to turn it into one? It turns out that we can (almost always) simultaneously get a distribution and commit to preserving the relative ratios of the specified probabilities within the links, while also more clearly exposing our independence assumptions. This can be done by treating the conditional distributions $\mathrm{Pr}(L = l \mid S = s, C = c)$ and $\mathrm{Pr}(L = l \mid W = w)$ as *factors*, which multiplied together give the relative probability density of any setting of variables $S \times C \times W \times L$

$$\mathrm{Pr}(s, c, w, l) \propto \phi_1(s, c, l)\phi_2(w, l)$$

where $\phi_1(s, c, l) = \mathbf{p}(L = l \mid S = s, C = c)$, $\phi_2(w, l) = \mathrm{Pr}(L = l \mid W = w)$. This can also be represented graphically, with a *factor graph*—a commonly used graphical model which is roughly a generalization of Bayesian and Markov Networks.



In this diagram, circles represent variables, and the boxes represent factors that depend on variables they connect to. This is a lot more modular (we can add and remove factors as we like). We now have a distribution that represents both beliefs, but this is not really what we were thinking of earlier. Beyond simply the inevitable effects of representing our knowledge as a distribution, such as forcing us to implicitly adopt marginal distributions over the variables S, C, and W, a product of factors has additional undesirable properties that are not shared by PDGs:

1. We can't weight the pieces of information differently. Although the scale of each factor ϕ_i gives us a degree of freedom in which to encode this information, it cannot be used, as $(a\phi_1)(b\phi_2) = (ab)(\phi_1\phi_2)$, and the coefficient ab is entirely negated by the normalization constant.

2. The resulting picture does not encode conditional probabilities in quite the way that we had wanted: now updating on S does not preserve $L \mid C, S$, bringing L along as required, but rather does something unclear and very global: we've lost the dependency structure we had in the first few pictures. Relatedly, we have lost the directedness of the edges, and with it, hope that the edges represent anything causal. Furthermore, the addition of new factors can dramatically change the meanings of existing ones. For all of these reasons, it is incredibly difficult to interpret part of the graph by itself.For instance, knowing the joint distribution does not determine the values of the factors.

3. If at least one factor is zero for every setting of S, C, W, L, no distribution is defined — in the face of inconsistency, the entire formalism ceases to work at all.

4. More locally, had the two sources of conditional distributions on L been incompatible, (e.g., the support of each $\boldsymbol{\mu}(L \mid w)$ strictly larger than any $\boldsymbol{\mu}(L \mid s, c)$) one would have reason to further examine both beliefs — a situation that is indistinguishable from an alternate factor graph where they agreed somewhere in between.

△

While factor graphs offer a solution of great generality, they sacrifice interpretability and important internal features of our original belief representation, so that they can represent distributions.

3.3 Conditional Random Fields

A conditional random field (CRF) is an undirected graphical model which represents a conditional distribution, making it *under-constrained*.

4 The Full Model: Sub-stochastic Transitions

In this section we will see why we called the object in definition 2.1 a *strict* PDG. Sometimes an otherwise very useful variable might not apply in a small percentage of cases; in this case, we want a way of putting all of the extra probability mass in a "something else happened" bucket, giving us effectively a sub-stochastic matrix, or a a lower probability on singletons. For instance, the variable describing whether or not your answer is correct doesn't make sense if you weren't solving problems; the amount of money in your wallet doesn't make sense if you don't own one, and so forth. So now, when you're trying to predict the probability of certain amounts of money in your wallet, some of the probability mass needs to go into the "not applicable / something else" bucket.

There are several closely concepts that we will be able to employ with our framework after integrating them

1. Allowing random variables to be partial, rather than total functions of W.
2. Allow matrices to be sub-stochastic, rather than stochastic
3. Replacing probabilities, with the more general class of lower probability measures.

This generalization is useful, but our primary motivation for this generalization is so that we can represent implication, and thus a weakening of knowledge as it travels through our graph, in a way that is not just entropy (which might not be distinguishable from certain knowledge of a high entropy distribution otherwise).

At first glance, though, it might not be clear why this particular weakening buys us anything at all, because we can always just add the "something else" bucket $\bullet$, to $\mathcal{V}(X)$ for each X, and come up with a new strict PDG. A variable which might not make sense can always take a `null` value, and so now the set of possible is once again exhaustive. From the perspective of providing conditional distributions, however, this resolution poses a problem: our marginals now require us to estimate distributions from a null value— this is problematic, as a big part of the reason we've been using links to avoid assigning probabilities to everything. Suppose you are trying to represent the belief that you're happier when you get the right answer as a marginal link $L[\text{RightAns} \to \text{☺}]$. We now need a distribution on happiness when you get the right answer, when you get the wrong answer, and also for when $\bullet$. Why might it not be applicable? Are you not solving problems because you're skiing? Because you've been injured? Maybe you are solving problems but there are multiple right answers? You can't just answer with a prior over happiness if you want to have consistent beliefs, because solving problems and happiness might be correlated. One *could* have such a thing but it seems unreasonable not to be able to express a belief about "does the right answer make you happy?" without also answering the much more difficult question, "how happy are you when 'the right answer' is not applicable to your current situation?"

To see how this increases our expressive power, suppose A, B are binary variables (taking values $a, \bar{a}$ and $b, \bar{b}$ respectively). While we can easily easily represent $A = B$, $A = \neg B$ as stochastic matrices,

$$p(B \mid A) = \begin{array}{cc} \begin{array}{cc} b & \bar{b} \end{array} & \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & \begin{array}{c} a \\ \bar{a} \end{array} \end{array} \qquad \text{and} \qquad p(B \mid A) = \begin{array}{cc} \begin{array}{cc} b & \bar{b} \end{array} & \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} & \begin{array}{c} a \\ \bar{a} \end{array} \end{array}$$

we cannot (via stochastic matrices) represent an assertion that $A \Rightarrow B$ without also giving a distribution over B given $\bar{a}$. One strategy is a uniform prior (used in [**logicalinduction**]), but this can easily lead to avoidable inconsistencies — perhaps for totally different reasons you have very good reason to believe that the true distribution of B is true in 90% of cases; you don't want an arbitrary assumption of a prior competing with actual knowledge.

For this reason, we drop the requirement that our null element, $\bullet$, indexes a distribution in marginals. Below is an example of transition matrix $A \to B$ including the extra element. As mentioned, the last row is not something we are keeping track of.

$$\begin{array}{cc} \begin{array}{ccc} b_0 & b_1 & \bullet \end{array} & \\ \left[\begin{array}{ccc} .2 & .1 & 0.7 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ \hline .2 & .6 & 0 \end{array}\right] & \begin{array}{c} a_0 \\ a_1 \\ a_2 \\ \bullet \end{array} \end{array}$$

Furthermore, because the final column is just whatever is necessary to make the rows sum to 1, we don't need to keep that either; as a result, it is sufficient to keep a smaller matrix without any $\bullet$-indices; the only price that we pay is that this matrix is *sub*-stochastic rather than stochastic: its row entries sum to at most 1, rather than exactly 1. Composition works just as before; the product of sub-stochastic matrices is sub-stochastic. A probability distribution alone, and by extension a standard Bayesian network cannot do this — because we require the look-up tables to exactly match all possible values, we can't drop any without totally giving up on any world which looks like that.

4.1 Relation to Partial Functions of W

4.2 Reduction to Lower Probability Measures

5 Relations to Other Representations of Uncertainty

Probabilistic Dependency Graphs are far from the first formalism to provide a weaker notion of uncertainty than probability. Belief functions, inner measures, sets of probabilities, lower probabilities, weighted sets of probabilities, and plausibility measures have all been studied extensively in the past. One feature that each of these has in common is that they are under-specified, from the perspective of wanting probabilities for everything.

$$\begin{tikzcd} \boxed{1} \arrow[r] & \boxed{\Omega} \end{tikzcd}$$

The natural question now becomes: to what do these under-constrained representations of belief correspond to under-constrained bits of a Probabilistic Dependency Graph?

5.1 Conditional Probability Spaces

$$\begin{tikzcd} & & \boxed{U} \\ \boxed{1} \arrow[r] & \boxed{\Omega} \arrow[ur, two heads] \arrow[dr, two heads] & \\ & & \boxed{V} \end{tikzcd}$$

5.2 Sets of Probability Measures

5.3 Lower Probabilities

6 Using Inconsistency

6.1 Belief Updates

7 Algorithms

7.1 Belief Propagation

7.2 Sampling

One of the nice thignWe still need a sampling

8 Discussion

8.1 Ways to View This

- An unshackled bayesian network with explicit higher order edges
- A vectorized, higher dimensional version of conditional probability spaces that includes torsion
- An attention-shaped diagram into the Markov Category

9 Conclusions

9.1 A List of PDG Benefits

1. We can represent both over-constrained and under-constrained mental states, both of which we argue are important components of an agent's state.
2. Over-constrained models may be inconsistent; such inconsistencies provide a natural way of prescribing changes in mental state. Moreover, many standard algorithms, such as belief updating via Jeffrey's rule, as well as marginalization algorithms such as belief propagation, can be regarded as special cases of consistency reduction.
3. PDGs can emulate the functionality of not only other graphical models (such as Bayesian Networks, and to a large extent, factor graphs), but also other non-probabilistic notions of uncertainty.
4. The local interpretation of arrows makes it much less invasive to add, remove, and partially interpret parts of the model, compared to other graphical models.
5. In conjunction with the ability to merge, split, and compress variables, agents can use the inconsistency and modularity that PDGs offer to subjectively expand and contract the set of possible worlds, without necessarily interacting with one true set of them which happens to be common knowledge.
6. The modularity enables type-forming rules which can be used to implement deductive inference.
7. In contrast with a simple collection of constraints, inconsistencies can be more local, and individual pieces of information have limited impact on the semantics.

A Proofs

B

B.1 Random Variables

If $\mathcal{W} = (W, \mathcal{F}, \mu)$ is a measure space, and $\mathcal{X} = \{X_i : W \to \mathcal{V}(X_i)\}_{i \in I}$ is a collection of measurable random variables on W,[3] and $\mathcal{L} \subseteq I \times I$ is a collection of pairs of variables such that the agent [todo: *what is a way of phrasing this that doesn't sound like it's shoehorned in? $\mathcal{L}$ really can represent anything an agent knows. Any subjective conditional probability distribution μ' such that the only measurable subsets are "axis aligned", in that they involve queries on only one variable, can be represented by $\mathcal{L}$, and for other queries we can simply change variables.*], we call $(\mathcal{W}, \mathcal{X})$ an *ensemble*.

Proposition B.1. *There is a natural correspondence between strict PDGs as defined in definition 2.1, and ensembles such that* [todo: *spell this out explicitly to avoid vague categorical intuition*] *. . . μ's are defined on same set and produce same values.*

Proof. [outline: On the one hand, $(\prod_{N \in \mathcal{N}} \mathcal{V}(N).\text{set}, \bigotimes_{N \in \mathcal{N}} \mathcal{V}(N).\text{algebra}, \boldsymbol{\mu})$ is a measure space, with $\{X_N = \pi_N : (\prod \mathcal{V}(N')) \to \mathcal{V}(N)\}_{N \in \mathcal{N}}$ a set of random variables

and on the other, $(I, \mathcal{L}, \mathcal{X}', \mu|_{\mathcal{L}})$ is a strict PDG. □

This is the technical underpinning of our flippant, noncommittal treatment of possible worlds: any time we are thinking in terms of random variables or probability distributions on a fixed set W, we can instead reduce

The complexity of the representation is $O(XV + LV^2)$, compared to $O(XW)$

[3] that is: $\mathcal{V}(X_i)$ is a measurable space, taking the form $(D, \mathcal{D})$, and $X_i : W \to D$ is a set function such that for every $B \in \mathcal{D}$, the set $X_i^{-1}(B) \in \mathcal{F}$