Dependency Graphs

Oliver Richardson, oli@cs.cornell.edu

December 24, 2019

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 sub-distributions. 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 Jeffreys 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] As we will see, one reason these inconsistencies are difficult to avoid is our conceptual modularity: features of the world can be noticed, forgotten, and fragments of beliefs can be locally recombined long before they crystallize into global distributions or theorems. I would still cut this. I don't think that the average reader will know what you mean, which is bad for the second sentence of an introduction. I think that the net gain for this sentence is negative.
We introduce a new graphical representation of uncertainty that offers significant advantages over more standard approaches, such as Bayesian Networks and Factor Graphs, including both better modularity and the capability of representing inconsistency. While we agree that inconsistency is to be avoided, we also see its possibility as an integral part of reasoning, and we will even be able to recover standard algorithms (conditioning, etc.) as special cases of inconsistency reduction (section 9). The rest of this section consists of examples focusing on these two benefits (modularity and the possibility of inconsistency), why we want them, and the failure of other epistemic representations to quite capture what we have in mind.

1.1 Examples and Comparisons to Bayesian Networks

Example 1.1. 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

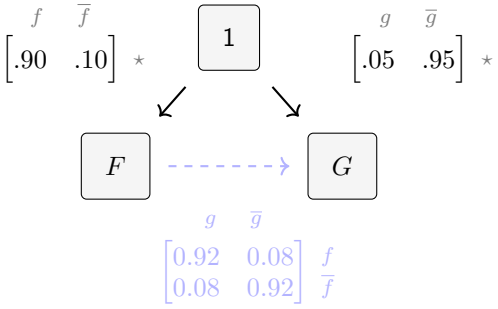
g	$\overline{g}$
.05	.95

[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. This piece of information 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 piece of information, or the new observation; in general any mixture will do. Note some features of this scenario:

1. Even though the additional information takes the form of a conditional probability table, expressing it as an edge in our Bayes Net overwrites other information: by adding a link $F \to G$, the Bayes Net loses all of our prior information about G. On further thought, I don't buy this at all. If you had the "right" prior on F and the "right" CPT on G, the posterior on G should be the prior you had before. I would still cut this.
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.[2][3]

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 (among other things; see section **??**), we introduce the *unit variable*, 1, or 'true', which always takes its unique value (which we call $\star$).



The original state of knowledge, consists of all three nodes and the two edges from 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 when necessary): we merely record the constraints imposed by the given probabilities.

The new knowledge we gain from realizing floomps are guns 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 $1 \to G$, 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 to delete the new edge. △

I would still cut this; I don't think anyone will argue that you need to resolve the inconsistency immediately, except for those who believe that for some reason, you should always have consistent beliefs. But if you believe that, the reasons go beyond needing to make decisions. This is too much of a distraction (so, in my view, a net negative).

[2] [note: *You say that I should remove this, but I feel like it is important to address the Bayesian concern: that you should have a belief state at all times, and when it matters you will have the right one. One could think that*] One might argue that you need to resolve the inconsistency in case you need to make decisions before you can discover the truth—that it is appropriate to soften all three beliefs as appropriate rather than leave the inconsistency. This is a reasonable point of view, and still consistent with using a PDG—but one should note that this is an eager rather than lazy approach, and will in general be more expensive.

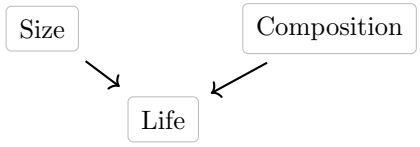
[3] [note: *This used to be the paragraph, but I moved it into a the footnote rather than cutting it. I was a bit unsure about how big a deal the concern was until I thought about it this way.*] One reason to be wary of resolving the inconsistency is that the process is not invertible: from a probabilistic perspective, resolution of the inconsistency is a projection (read: non-injective) into a feasible subspace. Thus this eager resolution loses information, which might be useful in the future.

You should cut this. It's a distraction. You really don't want to get into this at this point in the introduction.

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 1.2. 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:

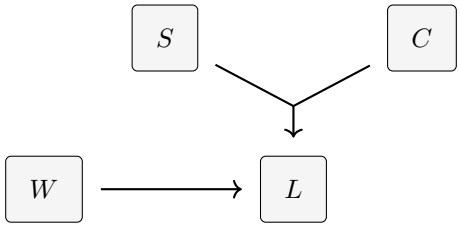
$$\Pr(\text{Life} \mid \text{Size}, \text{Comp}) = \begin{array}{c} \begin{matrix} \text{life} & \text{no life} \end{matrix} \\ \begin{bmatrix} .1 & .9 \\ .2 & .8 \\ .05 & 0.95 \\ 0.00001 & 0.99999 \end{bmatrix} \end{array} \begin{matrix} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gasseous} \end{matrix}$$



This one conditional probability table cannot constitute a full Bayesian network. It is indeed a fragment of the Bayesian Network on the right, but in order to 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;[4] PDGs have no such requirement.

As I said before (and you point out in your footnote), people have long considered qualitative BNs, so I would call this a requirement. I would actually cut this sentence.

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.[5] 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:



which represents having -> This means that we have

which represents having two conditional probability tables on L: one from $S \times C$ and the other from W. This would allow us to combine our two beliefs, without also providing much more information than we're prepared to equivocate on. 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 $S \times C$. △

say instead: requiring us to give information that we do not have

Precisely because people have long considered qualitative BNs, I would cut this footnote and the sentences it's a footnote too.

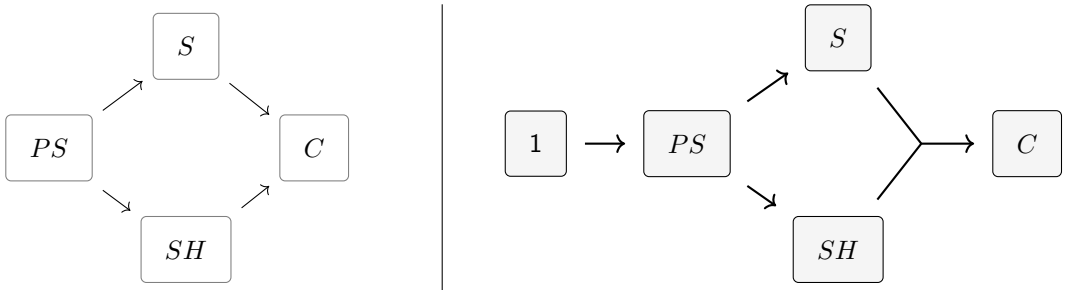
[4] It is also possible to partially interpret Bayesian Networks, and simply not have a table. [note: *I feel uncomfortable cutting the following because discriminative graphical models super commonly used and I suspect many people reading the paper will object that this can already be done; I want to nod to it.*] Models like these are well used to hardcore probabilists as well. In statistical learning theory, an 'incomplete' model like this is a discriminative (or conditional) model, as compared to a generative one.

This is a distraction. Cut it. You need a focused introduction, that doesn't bring in lots of tangential details.

[5] 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.

PDGs are actually strictly more expressive than Bayesian Networks, and there is a straightforward conversion from a BN to a PDG(section **??**). The two graphical models also look very much alike (and indeed are exactly the same for linear chains of variables); in some sense the biggest difference is the interpretation of two colliding arrows; Example 1.3 depicts this difference. Cut

Example 1.3. 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 **??**.



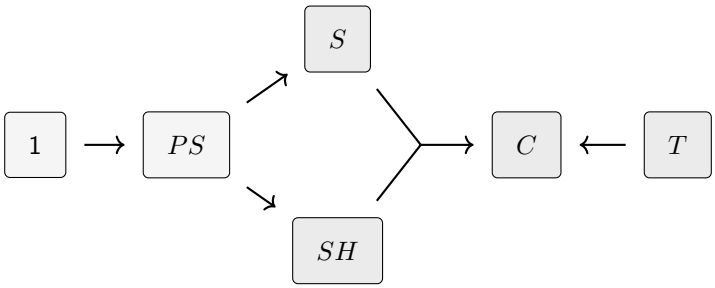
(a) Bayesian Network (b) Corresponding PDG

Figure 1: Both graphical models representing the conditional relationships in example 1.3

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 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 4), 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 fig. 1b as representing the same distribution as the BN. Figure 1(b)

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:



This is again an illustration of the modularity and the possibility for inconsistency; compare the right half of the diagram (shaded slightly darker) with the topological equivalent in example 1.2. △

Cut

1.2 Revisiting Exmaples with Factor Graphs

[note: *After further consideration, I'm thinking that I should use the text below instead of a fresh introduction to factor graphs; I think it is better when tied to the example, especially since I can just refer to a textbook, and the introduction below highlights the important features.*]

cut; nothing we said indicated that the information didn't provide a partial specification of a probability.

Example 1.2 (continuing from p. 4)**.** 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 $p(L = l \mid S = s, C = c)$ and $p(L = l \mid W = w)$ as *factors*, which multiplied together give the relative probability density of any setting of variables $S \times C \times W \times L$

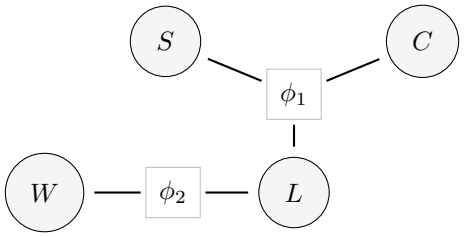
We may be able to get away with this approach to introducing factor graphs, but this needs to be rewritten; see the bottom of the page

Is Pr different from p?

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

where $\phi_1(s, c, l) = p(L = l \mid S = s, C = c)$, $\phi_2(w, l) = p(L = l \mid W = w)$. This can also be represented graphically, with a *factor graph*—a commonly used graphical model hailed as a strict generalization of Bayesian Networks and Markov networks.

hailed -> which is; no need for flowery language



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. [note: *You tell me to provide an example, but this will take time to do well, and you told me not to add anything. Do you want me to add an additional example focused on this?*] 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. [note: *This is actually a separate point and I don't think redundant; I've moved it here.*] 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.

After some though, I think the best thing to do for the intro is just to say after Example 1.3 something like "While other graphical representations (such as factor graphs [REF]) can deal with some of these concerns, they raise other problems (see Section ?). If we keep it here, then something like the following would be better: "We might try to represent the information in Example 1.2 using another type of graphical model that avoids the problems of BNs. One natural candidate is a factor graph. The idea here is that we try preserve the relative ratios of the specified probabilities, while also more clearly exposing our independence assumptions. This can be done by treating the conditional distributions p(L = l | S = s,C = c) and p(L = l | W = w) as factors, which multiplied together give the relative probability density of any setting of variables S × C × W × L." But I think this will be too confusing for someone who hasn't seen factor graphs before.

△

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.

In retrospect, I would replace this by a summary paragraph, which has much less than what you have here.

1.3 A More Comprehensive 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. Replace "overconstrained" by "inconsistent", cut "underconstrained"; that's standard
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. Cut
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. Replace this by "easier to add"
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. Cut points 5 and 6.
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. This should be folded into point 1 (and perhaps point 4)

2 Related Work

3 Formal Definitions and Variants

Definition 3.1. A *strict Probabilistic Dependency Graph* is a tuple $(N, L, \boldsymbol{\mu}, \mathcal{V})$ where

- $\mathcal{N}$: **Set** is a finite collection of nodes cut ": Set"
- $\mathcal{L} \subseteq N \times N$ is a set of directed links between nodes
- $\mathcal{V} : \mathcal{N} \to$ **MeasSet** is an N-indexed family of measurable sets, representing the values that a node can take Simplify this to "V associates with each node in N a domain". No need to talk about measurable sets.
- $\boldsymbol{\mu} : ((A, B) : \mathcal{L}) \to \mathcal{V}(A) \to \Delta(\mathcal{V}(B))$ is a family of conditional probability distributions on $\mathcal{V}(B)$ indexed by the values of A for every link $(A, B) \in \mathcal{L}$ [6]

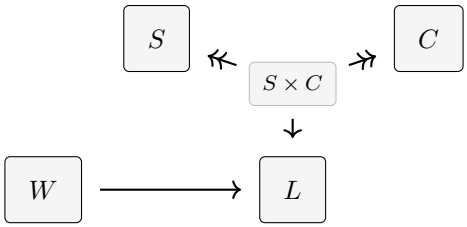
The definition of $\boldsymbol{\mu}$ is probably more familiar than it looks. If every $\mathcal{V}(N)$ is finite with all subsets measurable, then $\boldsymbol{\mu}_{A,B}$ is just a conditional probability table, or a stochastic matrix.

Example 1.2 (continuing from p. 4)**.** Earlier in example 1.2, we displayed the arrow from S and C to L as a directed hyper-edge. While we would like to maintain this intuition, it turns out that we can simplify our formalism by de-sugaring this picture into the following:

This is UAI/IJCAI, not POPL. Do not talk about de-sugaring.

[6] The definition $\boldsymbol{\mu}$ is slightly over-simplified if not everything is measurable. More generally if $\mathcal{V}(A) = (X, \mathcal{A})$ and $\mathcal{V}(B) = (Y, \mathcal{B})$, then for any link $L \in \mathcal{L}$, we're really referring to a function $\boldsymbol{\mu}_L : X \times \mathcal{B} \to [0, 1]$ such that $\boldsymbol{\mu}_L(x, -) : \mathcal{B} \to [0, 1]$ is a probability distribution and $\boldsymbol{\mu}_L(-, S)^{-1}(R) \in \mathcal{A}$ for any $S \in \mathcal{B}$, and measurable subset $R \subseteq [0, 1]$, technically making $\boldsymbol{\mu}_{A,B}$ a *Markov Kernel* from A to B.

Cut! For what it's worth, I don't understand what it means that V(A) = (X,\A). What the X? Don't introduce notation out of the blue! If you want to talk about Markov kernels (which is almost surely a bad idea at this point, you need to be much more gentle). I'm happy to assume that everything is measurable at this point.



If it's not relevant, don't mention it!

The double headed arrows are for degenerate conditional distributions, which are fully deterministic, but for now this is not terribly relevant. We can now present this PDG formally with the elements specified in definition 3.1; below we assume everything is measurable and omit this part of the formalism.

You should omit it from the beginning.

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

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

△

This works pretty well for the two edges that we described before, but the structural overhead of the additional de-sugaring: the $\boldsymbol{\mu}[S \times C \to S]$ and $\boldsymbol{\mu}[S \times C \to C]$ tables, as well as the set $\mathcal{V}(S \times C)$ seem like they didn't need to be specified, and one might even feel that it would be a mistake to allow any other table. Some reasons for this design decision include:

- It is easier to prove things about graphs than directed hyper-graphs. Similarly, defining composition and paths becomes a lot simpler.
- We can eliminate the clunkiness by fusing the model with an algebra, as in **??** — which will give us a lot more than modeling the hyper-edges directly.
- We will eventually also want to allow for the possibility of keeping only a relaxed, approximate representation of $\mathcal{V}$ and $\boldsymbol{\mu}$, and in particular, of the ones constructed logically in this way. By specifying them explicitly for now, we will have to do less work to regain manual control in **??**

Cut! Discussing the design choices may be reasonable, but the discussion should not use terms like "hyper-graphs" and should not talk about fusing models with algebras (whatever that means)

3.1 Alternate Presentations

3.1.1 Random Variables

Cut. This is a distraction.

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,[7] 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 3.1. *There is a natural correspondence between strict PDGs as defined in definition 3.1, 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 $O(XW)$

3.2 Sub-stochastic Transitions

If we include this in the UAI/IJCAI submission (and that's a big if) it needs to be introduced by an example.

In this section we will see why we called the object in definition 3.1 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. Relaxing the requirement $\int_W \mu(w)\ \mathrm{d}w = 1$ to $\int_W \mu(w)\ \mathrm{d}w \leq 1$
3. Requiring that matrices be sub-stochastic, rather than stochastic
4. Replacing probabilities, with the more general class of lower probability measures.

Make everything finite, so we can sum, rather than integrating.

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,

[7] 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 $X_i^{-1}(B) \in \mathcal{F}$

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 $L[\text{RightAns} \to ☺]$. 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,

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

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.

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

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.

3.2.1 Relation to Partial Functions of W

3.2.2 Reduction to Lower Probability Measures

4 Semantics

This should go immediately after the syntax. You need to slow down here and give lots of intuiition.

These graphs admit multiple semantics. As discussed in section **??**, we think of Probabilistic Dependency Graphs as being a representation of knowledge 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.

4.1 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 might thought of as representing the set of all distributions that marginalize out to it exactly.

Definition 4.1. If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG, let $[\![M]\!]_{\mathbf{Set}}$ be the set of distributions over the variables in M consistent with $\boldsymbol{\mu}$ on every marginal. Formally,

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

4.2 As Weighted Distributions

Definition 4.2. If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG with joint worlds W, and $\ell : \Delta W \to \mathbb{D}$ is a scoring function, let $[\![M]\!]^{\ell}$ be the set of distributions that dominate. Formally,

$$\Big[\!\Big[M\Big]\!\Big]^{\ell}(\mu) := \left\{ \mu \in \Delta \left[\prod_{N \in \mathcal{N}} \mathcal{V}(N) \right] \;\middle|\; \begin{array}{r} \ell(\mu) \succeq \ell(\mu') \\ \text{for all } \mu' \in \Delta \prod \mathcal{V} \end{array} \right\}$$

4.3 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 4.3. If M is a PDG, $[\![-]\!]_{\mathbf{S}}$ is a semantics, $(D, \preceq)$ 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,

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

Fact 4.1.

$$[\![M]\!]^{\ell}_{\boldsymbol{Set}} = \Big\{ \mu \in \Delta \prod \mathcal{V} : [\![M]\!]^{\ell}(\mu) \Big\}$$

One particularly useful one for emulating Bayesians is the following one, maximizing entropy:

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

This corresponds to a lexicographical ordering on distributions

Similarly, we can define

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

5 Relations to Other Graphical Models

5.1 Bayesian Networks

5.2 Factor Graphs

6 For Bayesians

The standard approach to probabilistic modeling is to start by selecting a measurable space of possible outcomes Ω, and then put a normalized measure on it and compute desired quantities with it. Before you can begin to think about random variables, which are defined as set $X_i : \Omega \to V_i$ from outcomes to the set of values V_i that X_i can take on, you have to specify Ω. This construction works well, so long as Ω is large enough to express everything you ever cared to conceptualize. Because agents are expected to have probability distributions over Ω, the set of worlds that they consider possible must effectively stay constant over time, to use conditioning as a sole way of changing a mental state.

This strategy, works so long as you are an omniscient modeler. If you are modeling a system in which you know the set of all possible outcomes, either implicitly or explicitly, you can just collect them and use this to be Ω, marginalize out appropriately, and let agents figure out their own distributions on subspaces of Ω. Still, this is not entirely satisfying, for several reasons.

[todo: *most of these are unfinished thoughts, some need to be deleted*]

1. The set of possible worlds may be arbitrarily finer than the set of possible worlds, depending on the language of the agent, which may involve multiple descriptions of the same phenomenon
2. Even if it is true that subspaces of Ω which are isomorphic to the sets of worlds W_i that the agents are modeling in their heads (if Ω is a maximally fine over-approximation to possible), the embedding is not at all clear [todo:]
3. There may not be an omniscient modeler, and even if there were one, it seems very strange for an agent to have any access to it. Suppose you are using probabilities to describe real uncertainties in your life. To do this the standard way, you need to chose the subspace of the one true Ω [todo: *why is this problematic? for reasons other than inability to change?*]
4. Agents can never gain access via any standard mechanism to new worlds. There's no principled way to add worlds from Ω to W_i. Effectively, they can never learn new concepts.
5. Any updates must be done on the entire space of things you consider possible [todo: *response: of course, this is all handled implicitly, so it's taken care of*].
6. While agents are free to merge multiple states of Ω into a single state in W_i, they cannot do the reverse: an agent cannot have a finer granularity than Ω for discerning events. This would . This also implies that agents are logically omniscient.

For all of these reasons, we take the view that probability should be thought of subjectively,

At the same time, starting with a set of random variables $\{X_i\}$, and setting $\Omega = \prod_i V_i$ to be every possible assignment of variables is also an abuse of the word "possible".

7 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.

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

7.1 Sets of Probability Measures

As we discuss in section 4.1

8 Using Inconsistency

Given a distribution μ, and a

Definition 8.1 (consistency)**.** A Probabilistic Dependency Graph $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is *consistent* if there exists some joint probability Pr on all of the variables, or equivalently, if $[\![M]\!]_{\mathbf{Set}} \neq \emptyset$

$$\zeta(\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})) := \inf_{p \in \Delta(W^{\mathcal{V}})} \sum_{L[A,B] \in \mathcal{L}} \mathop{\mathbb{E}}_{a \sim p(A)} \left[D_{\mathrm{KL}}\Big(L(a) \,\Big\|\, p(B \mid a) \Big) \right] \tag{1}$$

9 Algorithms

9.1 Belief Propagation

9.2 Sampling

10 Discussion

10.1 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

10.2 Inconsistency

Inconsistency is bad. Believing a logically inconsistent formula can lead you to arbitrarily bad conclusions, having an infeasible set of constraints makes all answers you could give wrong, and having inconsistent preferences can lose you infinite money. We don't want to build inconsistent systems or agents with incoherent views of the world, and so, where possible, we design them so they cannot possibly be broken in this way. Suppose, for example, that we are trying to represent some quantity that must be a point on the unit circle. We could do it with an x and y coordinate, but this could be problematic because $x^2 + y^2$ might not be 1 — it would be safer and harder to go awry if we parameterize it by an angle $\theta \in [0, 2\pi)$ instead. In the absence of performance benefits (like needing to regularly use the y-coordinate and not wanting to compute a sine), why would we take the first approach, introducing a potentially complex data-invariant, when we could avoid it?

This line of thought, though common and defensible, is flawed if we are not perfectly confident in the design of both our system and the ways it can interact with the outside world. Using similar logic, we might ask

ourselves: Why ask programmers for type annotations when all instructions are operationally well-defined at run-time? Why use extra training data if there's already enough there to specify a function? Why estimate a quantity in two ways when they will yield different answers? Why repeat and rephrase your ideas when this could make you contradict yourself? Why write test cases when they could fail and make the project inconsistent? Why conduct an experiment if it could just end up contradicting your current knowledge?

These questions may seem silly, but there is a satisfying information theoretic answer to all of them: redundancy, though costly, is the primary tool that we use to combat the possibility of being wrong. Maintaining data invariants can be expensive but provides diagnostic information; in the example above, settings of x and y that don't lie on the unit circle provide diagnostic information that something has gone wrong. In many cases, it is also possible to paper over problems by forcibly re-instating local data invariants: for instance, we could re-normalize any values of x and y (so long as $xy \neq 0$; we can chose an arbitrary point otherwise) at every step. While this would reduce inconsistency, it also hides red flags.

Using a Bayesian Network to represent a probability distribution is like representing a circle with $\theta \in [0, 2\pi)$. By construction, the result must be a distribution, and nothing can possibly go wrong so long as we can always decide on exactly one distribution which is sufficient for our purposes.

The process of mechanistically forcing invariants is homologous to the standard practice for factor graphs: practitioners will often just assume that the density it defines is normalizable, and either forcibly re-normalize or cleverly avoid computing the normalization constant while still assuming that one exists; behavior is usually left unspecified in the unlikely event that it is not defined or zero.