Examples for "An Algebraic Approach to Stochastic Answer Set Programming"

Francisco Coelho Bruno Dinis Dietmar Seipel
Salvador Abreu

May 10, 2024

Abstract

We address the problem of propagating the probabilities from the annotated facts of an answer set program to its stable models, and from there to general events in the program's domain. Our approach is algebraic in the sense that it relies on an equivalence relation over the set of events, and uncertainty is expressed with variables and polynomial expressions. We propagate probability in the space of models and events, instead of dealing with the syntax of the program. We illustrate our methods with two examples, one of which shows a connection to bayesian networks.

Keywords: Answer-Set Programming, Stable Models, Probabilistic Logic Programming

1 A Simple but Fruitful Example

Consider A to be the following Stochastic Answer Set Program (SASP):

$$\begin{aligned} a &: 0.3, \\ b \vee c &\leftarrow a \end{aligned} \tag{1}$$

which has the set $\mathcal{P} = \{a : 0.3\}$ of probabilistic facts. This program is transformed into the disjunctive logic program

$$\begin{aligned} a &\vee \overline{a}, \\ b \vee c &\leftarrow a, \end{aligned} \tag{2}$$

with the set $\mathcal{M} = \{\, \overline{a}, ab, ac \,\}$ of three stable models. We use short-hand expressions like ab to denote a set of literals such as $\{a, b\}$.

Consider the program

$$\begin{aligned} a &: 0.3, \\ b &\leftarrow a \end{aligned}$$

We don't follow the clause $b \leftarrow a$ to attribute a probability to b. Instead, that will result from considering how events are related with the stable models and these with the total choices. If one follows the steps of **??** would get

$$\begin{aligned}
&P_{\mathcal{E}}(a) = P_{\mathcal{E}}(b) = P_{\mathcal{E}}(ab) = 0.05,\\
&P_{\mathcal{E}}(\overline{a}) = P_{\mathcal{E}}(\overline{a}b) = P_{\mathcal{E}}\left(\overline{a}\overline{b}\right) = \frac{7}{60},\\
&P_{\mathcal{E}}\left(\{\}\right) = 0.5.
\end{aligned}$$

The atoms of program eq. (1) are $\mathcal{A} = \{a, b, c\}$ and the literals are

$$\mathcal{L} = \left\{ \overline{a}, \overline{b}, \overline{c}, a, b, c \right\}.$$

In this case, $\mathcal{E}$ has $2^6 = 64$ elements. Some, such as $\{\overline{a}, a, b\}$, contain an atom and its negation (a and $\overline{a}$ in that case) and are inconsistent. The set of atoms $\mathcal{A} = \{a, b, c\}$ above generates 37 inconsistent events and 27 consistent events. Notice that the empty set is an event, that we denote by λ.

As above, to simplify notation we write events as $\overline{a}ab$ instead of $\{\overline{a}, a, b\}$.

Consider the program B defined by

$$\begin{aligned}
&a : 0.3,\\
&b : 0.6,\\
&c \leftarrow a \wedge b.
\end{aligned}$$

The probabilistic facts of B are

$$\mathcal{P}_B = \{\ a : 0.3, b : 0.6\ \},$$

and the total choices are

$$\mathcal{T}_B = \left\{\ \{a, b\}, \{a, \overline{b}\}, \{\overline{a}, b\}, \{\overline{a}, \overline{b}\}\ \right\}.$$

E.g., the total choice $\{\overline{a}, b\}$ results from choosing $a' = \overline{a}$ from the probabilistic fact $a : 0.3$ and $b' = b$ from $b : 0.6$, while $\{a, b\}$ results from choosing $a' = a$ from $a : 0.3$ and $b' = b$ from $b : 0.6$.

Continuing with the program from eq. (1), the probabilities of the total choices are:

$$\begin{aligned}
P_{\mathcal{T}}\left(\{a, b\}\right) &= 0.3 \times 0.6 && = 0.18,\\
P_{\mathcal{T}}\left(\{a, \overline{b}\}\right) &= 0.3 \times \overline{0.6} && = 0.3 \times 0.4 = 0.12,\\
P_{\mathcal{T}}\left(\{\overline{a}, b\}\right) &= \overline{0.3} \times 0.6 && = 0.7 \times 0.6 = 0.42,\\
P_{\mathcal{T}}\left(\{\overline{a}, \overline{b}\}\right) &= \overline{0.3} \times \overline{0.6} && = 0.7 \times 0.4 = 0.28.
\end{aligned}$$

Suppose that in this program we change the probability in $b : 0.6$ to $b : 1.0$. Then the total choices are the same but the probabilities become

$$\begin{aligned}
P_{\mathcal{T}}\Big(\{a, b\}\Big) &= 0.3 \times 1.0 & &= 0.3,\\
P_{\mathcal{T}}\Big(\{a, \overline{b}\}\Big) &= 0.3 \times \overline{1.0} & = 0.3 \times 0.0 &= 0.0,\\
P_{\mathcal{T}}\Big(\{\overline{a}, b\}\Big) &= \overline{0.3} \times 1.0 & = 0.7 \times 1.0 &= 0.7,\\
P_{\mathcal{T}}\Big(\{\overline{a}, \overline{b}\}\Big) &= \overline{0.3} \times \overline{1.0} & = 0.7 \times 0.0 &= 0.0.
\end{aligned}$$

which, as expected from stating that $b : 1.0$, is like having b as a (deterministic) fact:

$$\begin{aligned}
&a : 0.3,\\
&b,\\
&c \leftarrow a \wedge b.
\end{aligned}$$

Continuing the study of the program eq. (1), the total choice $t = \{\overline{a}\}$ entails a single stable model, $\overline{a}$, so $\mathcal{M}\big(\{\overline{a}\}\big) = \{\overline{a}\}$ and, for $t = \{a\}$, the program has two stable models: $\mathcal{M}\big(\{a\}\big) = \{ab, ac\}$.

$t \in \mathcal{T}$	$\mathcal{M}(t)$
$\{\overline{a}\}$	$\overline{a}$
$\{a\}$	ab, ac

The second case illustrates that propagating probabilities from total choices to stable models entails a non-deterministic step: *How to propagate the probability* $P_{\mathcal{T}}\big(\{a\}\big)$ *to each one of the stable models* ab *and* ac*?*

As a first step to propagate probability from total choices to events, consider a possible propagation of $P_{\mathcal{T}} : \mathcal{T} \to [0, 1]$ from total choices to the stable models, $P_{\mathcal{M}} : \mathcal{M} \to [0, 1]$.

The stable models are the ones from its derived program (in eq. (2)):

$$\mathcal{M} = \{\overline{a}, ab, ac\}.$$

It might seem straightforward to assume $P_{\mathcal{M}}(\overline{a}) = 0.7$ but *there is no explicit way to assign values to* $P_{\mathcal{M}}(ab)$ *and* $P_{\mathcal{M}}(ac)$. Instead, we assume this lack of information and use a parameter θ as in

$$\begin{aligned}
P_{\mathcal{M}}(ab) &= 0.3\,\theta,\\
P_{\mathcal{M}}(ac) &= 0.3\,(1 - \theta)
\end{aligned}$$

to express our knowledge that ab and ac are models related in a certain way and, simultaneously, our uncertainty about that relation. In general, there might be necessary several such parameters, each associated to a stable model s and a total choice t, so we write $\theta = \theta_{s,t}$.

A value for $\theta_{s,t}$ can't be determined just with the information given in that program, but might be estimated with the help of further information, such as empirical distributions from datasets.

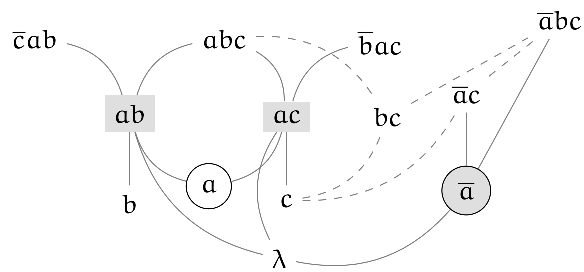


Figure 1: This (partial sub-/super-set) diagram shows some events related to the stable models of the program eq. (1). The circle nodes are total choices and shaded nodes are stable models. Solid lines represent relations with the stable models and dashed lines some sub-/super-set relations with other events. The set of events contained in all stable models, denoted by Λ, is $\{\lambda\}$ in this example, because $\overline{a} \cap ab \cap ac = \emptyset = \lambda$.

Continuing eq. (1), depicted in figs. 1 to 3, and $\mathcal{M} = \{ab, ac, \overline{a}\}$, consider the following stable cores of some events:

$$\begin{aligned}
[\![a]\!] &= \{s \in \mathcal{M} \mid s \subseteq a \vee a \subseteq s\} && = \{ab, ac\} \\
[\![abc]\!] &= \{s \in \mathcal{M} \mid s \subseteq abc \vee abc \subseteq s\} && = \{ab, ac\} \\
[\![\overline{c}ab]\!] &= \{s \in \mathcal{M} \mid s \subseteq \overline{c}ab \vee \overline{c}ab \subseteq s\} && = \{ab\} \\
[\![bc]\!] &= \{s \in \mathcal{M} \mid s \subseteq bc \vee bc \subseteq s\} && = \emptyset \\
[\![\lambda]\!] &= \{s \in \mathcal{M} \mid s \subseteq \emptyset \vee \emptyset \subseteq s\} = \{ab, ac, \overline{a}\} && = \Lambda
\end{aligned}$$

Events a and abc have the same stable core (SC), while $\overline{c}ab$ has a different SC. Also, bc is *independent of* (*i.e.* not related to) any stable model. Since events are sets of literals, the empty set is an event and a subset of any SM.

Consider again eq. (1). As previously stated, the stable models are the elements of $\mathcal{M} = \{\overline{a}, ab, ac\}$ so the quotient set of this relation is

$$[\mathcal{E}]_\sim = \left\{ \begin{array}{lll} \bot, & \Diamond, & [\![\overline{a}]\!]\,, \\ [\![ab]\!]\,, & [\![ac]\!]\,, & [\![\overline{a}, ab]\!]\,, \\ [\![\overline{a}, ac]\!]\,, & [\![ab, ac]\!]\,, & \Lambda \end{array} \right\},$$

where $\Diamond$ denotes the class of *independent events* e such that $[\![e]\!] = \{\emptyset\}$, while $\Lambda = [\![\mathcal{M}]\!]$

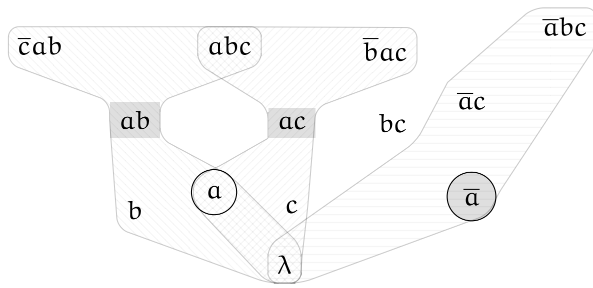


Figure 2: Classes of (consistent) events related to the stable models of eq. (1) are defined through sub-/super-set relations. In this picture we can see, for example, that $\{\overline{c}ab, ab, b\}$ and $\{a, abc\}$ are part of different classes, represented by different fillings. As before, the circle nodes are total choices and shaded nodes are stable models. Notice that bc is not in a shaded area.

is the set of events related with all SMs. We have:

$[\![e]\!]$	$[e]_\sim$	$\#[e]_\sim$
$\bot$	$\overline{a}a, \ldots$	37
$\Diamond$	$\overline{b}, \overline{c}, bc, \overline{b}a, \overline{b}c, \overline{bc}, \overline{c}a, \overline{c}b, \overline{bc}a$	9
$\overline{a}$	$\overline{a}, \overline{a}b, \overline{a}c, \overline{ab}, \overline{ac}, \overline{a}bc, \overline{ac}b, \overline{ab}c, \overline{abc}$	9
ab	$b, ab, \overline{c}ab$	3
ac	$c, ac, \overline{b}ac$	3
$\overline{a}, ab$		0
$\overline{a}, ac$		0
ab, ac	a, abc	2
Λ	λ	1
$[\mathcal{E}]_\sim$	$\mathcal{E}$	64

Notice that $bc \in \Diamond$, as hinted by figs. 1 and 2.

The program from eq. (1) has no information about the probabilities of the stable models that result from the total choice $t = \{a\}$. These models are $\mathcal{M}(\{a\}) = \{ab, ac\}$ so we need two parameters $\theta_{ab,\{a\}}, \theta_{ac,\{a\}} \in [0, 1]$ and such that

$$\theta_{ab,\{a\}} + \theta_{ac,\{a\}} = 1.$$

If we let $\theta = \theta_{ab,\{a\}}$ then

$$\theta_{ac,\{a\}} = 1 - \theta = \overline{\theta}.$$

Also

$$\theta_{ab,\{\overline{a}\}} = 0,$$
$$\theta_{ac,\{\overline{a}\}} = 0$$

because $ab, ac \notin \mathcal{M}(\overline{a})$.

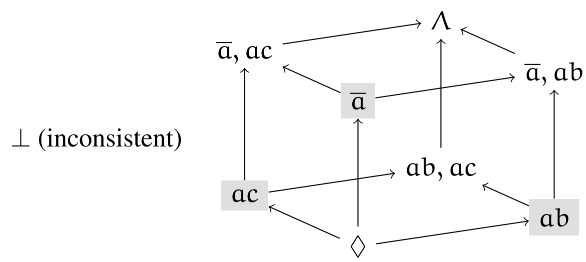


Figure 3: Lattice of the stable cores from eq. (1). In this diagram the nodes are the different stable cores that result from the stable models plus the *inconsistent* class ($\bot$). The bottom node ($\Diamond$) is the class of *independent* events, those that have no sub-/superset relation with the SMs and the top node (Λ) represents events related with all the SMs. As in previous diagrams, shaded nodes represent the SMs.

	A	B	C	D	E	F	G	H	I	J
	$[\![e]\!]$	$\mu_{\mathcal{M}}(s, \{\overline{a}\})$			$\mu_{\mathcal{M}}(s, \{a\})$			$\mu_C([e]_\sim, \{\overline{a}\})$	$\mu_C([e]_\sim, \{a\})$	$\mu_C([e]_\sim)$
		$\overline{a}$	ab	ac	$\overline{a}$	ab	ac	$\mu_{\mathcal{T}}(\{\overline{a}\})$	$\mu_{\mathcal{T}}(\{a\})$	
		1	0	0	0	θ	$\overline{\theta}$	0.7	0.3	
1	$\overline{a}$	1			0			1	0	0.7
2	ab		0			θ		0	θ	0.3θ
3	ac			0			$\overline{\theta}$	0	$\overline{\theta}$	$0.3\overline{\theta}$
4	$\overline{a}, ab$	1	0		0	θ		1	θ	$0.7 + 0.3\theta$
5	$\overline{a}, ac$	1		0	0		$\overline{\theta}$	1	$\overline{\theta}$	$0.7 + 0.3\overline{\theta}$
6	ab, ac		0	0		θ	$\overline{\theta}$	0	$\theta + \overline{\theta} = 1$	0.3
7	Λ	1	0	0	0	θ	$\overline{\theta}$	1	$\theta + \overline{\theta} = 1$	1

Continuing eq. (1), we show the propagation of $\mu_{\mathcal{T}}$ to $\mu_{\mathcal{M}}$ and then to μ_C. The table above resumes the calculations to compute $\mu_C([e]_\sim)$ for each $e \in \mathcal{E}$. For example, $e = abc$ the calculation of $J6 = \mu_C([abc]_\sim)$ follows these steps:

1. $[\![abc]\!] = \{ab, ac\}$ — is in line 6 of the table.
2. Since $\mathcal{T} = \{\{a\}, \{\overline{a}\}\}$, we need to calculate $I6 = \mu_C([abc]_\sim, \{a\})$ and $H6 = \mu_C([abc]_\sim, \{\overline{a}\})$. Now:

$$H6 = \mu_C([abc]_\sim, \{\overline{a}\}) = \sum_{s \in [\![abc]\!]} \mu_{\mathcal{M}}(s, \{\overline{a}\}) = \mu_{\mathcal{M}}(ab, \{\overline{a}\}) + \mu_{\mathcal{M}}(ac, \{\overline{a}\})$$

$$I6 = \mu_C([abc]_\sim, \{a\}) = \sum_{s \in [\![abc]\!]} \mu_{\mathcal{M}}(s, \{a\}) = \mu_{\mathcal{M}}(ab, \{a\}) + \mu_{\mathcal{M}}(ac, \{a\})$$

3. The $\mu_{\mathcal{M}}(s, t)$ above result from the non-empty cells in columns $B:D$ and $E:G$:

$$\begin{aligned} C6 &= \mu_{\mathcal{M}}\big(ab, \{\overline{a}\}\big) &&= 0 \\ D6 &= \mu_{\mathcal{M}}\big(ac, \{\overline{a}\}\big) &&= 0 \\ F6 &= \mu_{\mathcal{M}}\big(ab, \{a\}\big) &&= \theta \\ G6 &= \mu_{\mathcal{M}}\big(ac, \{a\}\big) &&= \overline{\theta} \end{aligned}$$

4. So we have, in columns H, I:

$$\begin{aligned} H6 &= \mu_C\big([abc]_\sim, \{\overline{a}\}\big) &&= 0 + 0 &&= 0 \\ I6 &= \mu_C\big([abc]_\sim, \{a\}\big) &&= \theta + \overline{\theta} &&= 1 \end{aligned}$$

5. At last, in columns H, I and J:

$$\begin{aligned} J6 = \mu_C\big([abc]_\sim\big) &= \sum_{t \in \mathcal{M}} \mu_C\big([abc]_\sim, t\big)\, \mu_{\mathcal{T}}(t) \\ &= \mu_C\big([abc]_\sim, \{\overline{a}\}\big)\, \mu_{\mathcal{T}}\big(\{\overline{a}\}\big) + \mu_C\big([abc]_\sim, \{a\}\big)\, \mu_{\mathcal{T}}\big(\{a\}\big) \\ &= 0\overline{\theta} + 1\theta = 0 \times 0.7 + 1 \times 0.3 = 0.3 \end{aligned}$$

We determined $\mu_C\big([e]_\sim, t\big)$ and also $\mu_C\big([e]_\sim\big)$, the measure of each class, by marginalizing the total choices.

$[\![e]\!]$	μ_C	$\#[e]_\sim$	$\mu_{\mathcal{E}}$	$P_{\mathcal{E}}$
$\perp$	0	37	0	0
$\Diamond$	0	9	0	0
$\overline{a}$	$\frac{7}{10}$	9	$\frac{7}{90}$	$\frac{7}{207}$
ab	$\frac{3}{10}\theta$	3	$\frac{1}{10}\theta$	$\frac{1}{23}\theta$
ac	$\frac{3}{10}\overline{\theta}$	3	$\frac{1}{10}\overline{\theta}$	$\frac{1}{23}\overline{\theta}$
$\overline{a}, ab$	$\frac{7+3\theta}{10}$	0	0	0
$\overline{a}, ac$	$\frac{7+3\overline{\theta}}{10}$	0	0	0
ab, ac	$\frac{3}{10}$	2	$\frac{3}{20}$	$\frac{3}{46}$
Λ	1	1	1	$\frac{10}{23}$
	$Z = \frac{23}{10}$		$\frac{\mu_C}{\#[e]_\sim}$	$\frac{\mu_{\mathcal{E}}}{Z}$

From there we can calculate the measure $\mu_{\mathcal{E}}(e, t)$ of each event given t, by simply dividing $\mu_C\big([e]_\sim, t\big)$ by $\#[e]_\sim$, the total number of elements in $[e]_\sim$. Then we marginalize t in $\mu_{\mathcal{E}}(e, t)$ to get $\mu_{\mathcal{E}}(e)$. Finally, the normalization factor provides a coherent *prior* probability for each event.

In summary, the coherent *prior* probability of events of program eq. (1) is

$$\begin{array}{c|ccccccccc} [\![e]\!] & \perp & \Diamond & \overline{a} & ab & ac & \overline{a}, ab & \overline{a}, ac & ab, ac & \Lambda \\ \hline P_{\mathcal{E}}(e) & 0 & 0 & \frac{7}{207} & \frac{1}{23}\theta & \frac{1}{23}\overline{\theta} & 0 & 0 & \frac{3}{46} & \frac{10}{23} \end{array} \tag{3}$$

Say what is $P(a)$, $P(b)$, etc... Note that $P(a) + P(\overline{a}) \neq 1$ (explain why)

$[\![e]\!]$	#Z	$P_S([e]_\sim)$	$P_\mathcal{E}([e]_\sim)$
$\bot$	0	0	0
$\Diamond$	24	$\frac{24}{1000}$	0
$\overline{a}$	647	$\frac{647}{1000}$	$\frac{7}{23}$
ab	66	$\frac{66}{1000}$	$\frac{3}{23}\theta$
ac	231	$\frac{231}{1000}$	$\frac{3}{23}\overline{\theta}$
$\overline{a}, ab$	0	0	0
$\overline{a}, ac$	0	0	0
ab, ac	7	$\frac{7}{1000}$	$\frac{3}{23}$
$\overline{a}, ab, ac$	25	$\frac{25}{1000}$	$\frac{10}{23}$
	$n = 1000$		

Table 1: *Experiment 1: Bias to* ac. Results from an experiment where $n = 1000$ samples where generated following the *Model+Noise* procedure with parameters $\alpha = 0.1, \beta = 0.3, \gamma = 0.2$. Column #Z lists the number of observations on each class, the *empirical* distribution is represented by P_S and the *prior*, as before, is denoted by $P_\mathcal{E}$.

1.1 Estimating Parameters of the Fruitful Example

Table 1 lists the empirical results from an experiment where samples are classified according to the classes of eq. (1). These results can be *generated by simulation* in a two-step process, where (1) a "system" is *simulated*, to gather some "observations" and then (2) empirical distributions from those samples are *related* with the prior distributions from section 1. Tables 1 and 2 summarize some of those tests, where datasets of $n = 1000$ observations are generated and analyzed.

Simulating a System. Following some criteria, more or less related to the given program, a set of events, that represent observations, is generated. Possible simulation procedures include:

- *Random.* Each sample is a Random Set of Literals (RSL). Additional sub-criteria may require, for example, consistent events, a Random Consistent Event (RCE) simulation.
- *Model+Noise.* Gibbs' sampling (Geman and Geman 1984) tries to replicate the program model and also to add some noise. For example, let $\alpha, \beta, \gamma \in [0, 1]$ be some parameters to control the sample generation. The first parameter, α is the "out of model" samples ratio; β represents the choice a or $\overline{a}$ (explicit in the model) and γ is the simulation representation of θ. A single sample is then

$[\![e]\!]$	$\#_{0.2}$	$\#_{0.8}$	$\#_{0.5}$
$\bot$	0	0	0
$\Diamond$	24	28	23
$\overline{a}$	647	632	614
ab	66	246	165
ac	231	59	169
$\overline{a}, ab$	0	0	0
$\overline{a}, ac$	0	0	0
ab, ac	7	8	4
$\overline{a}, ab, ac$	25	27	25

Table 2: *Experiments 2 and 3.* Results from experiments, each with $n = 1000$ samples generated following the *Model+Noise* procedure, with parameters $\alpha = 0.1, \beta = 0.3, \gamma = 0.8$ (Experiment 2: bias to ab.) and $\gamma = 0.5$ (Experiment 3: balanced ab and ac.). Empirical distributions are represented by the random variables $S_{0.8}$ and $S_{0.5}$ respectively. Data from experience table 1 is also included, and denoted by $S_{0.2}$, to provide reference. Columns $\#_{0.2}$, $\#_{0.8}$ and $\#_{0.5}$ contain $\#\left\{S_{0.2} \in [e]_\sim\right\}$, $\#\left\{S_{0.8} \in [e]_\sim\right\}$ and $\#\left\{S_{0.5} \in [e]_\sim\right\}$, the respective number of events in each class.

generated following the probabilistic choices below:

$$\begin{cases} \alpha & \text{by RCE} \\ & \begin{cases} \beta & \overline{a} \\ & \begin{cases} \gamma & ab \\ & ac \end{cases} \end{cases} \end{cases},$$

where

$$\begin{cases} p & x \\ & y \end{cases}$$

denotes "*the value of* x *with probability* p*, otherwise* y" — notice that y might entail x and *vice-versa*: E.g. some ab can be generated in the RCE.

- *Other Processes.* Besides the two sample generations procedures above, any other processes and variations can be used. For example, requiring that one of $x, \overline{x}$ literals is always in a sample or using specific distributions to guide the sampling of literals or events.

Relating the Empirical and the Prior Distributions. The data from the simulated observations is used to test the prior distribution. Consider the prior, $P_\mathcal{E}$, and the empirical, P_S, distributions and the following error function:

$$\operatorname{err}(\theta) := \sum_{e \in \mathcal{E}} \left(P_\mathcal{E}(e) - P_S(e)\right)^2 . \tag{4}$$

Since $P_\mathcal{E}$ depends on θ, one can ask how does the error varies with θ, what is the *optimal* (i.e. minimum) error value

$$\hat{\theta} := \arg\min_{\theta} \mathrm{err}(\theta) \tag{5}$$

and what does it tell us about the program.

In order to illustrate this analysis, consider the experiment summarized in table 1.

1. Equation (4) becomes

$$\mathrm{err}(\theta) = \frac{20869963}{66125000} + \frac{477}{52900}\theta + \frac{18}{529}\theta^2.$$

2. The minimum of $\mathrm{err}(\theta)$ is at $\frac{477}{52900} + 2\frac{18}{529}\theta = 0$. Since this leads to a negative θ value $\theta \in [0, 1]$, it must be $\hat{\theta} = 0$, and

$$\mathrm{err}\left(\hat{\theta}\right) = \frac{20869963}{66125000} \approx 0.31561.$$

The parameters α, β, γ of that experiment favour ac over ab. In particular, setting $\gamma = 0.2$ means that in the simulation process, choices between ab and ac favour ac, 4 to 1. For completeness sake, we also describe one experiment that favours ab over ac (setting $\gamma = 0.8$) and one balanced ($\gamma = 0.5$).

For $\gamma = 0.8$, the error function is

$$\begin{aligned}\mathrm{err}(\theta) &= \frac{188207311}{529000000} - \frac{21903}{264500}\theta + \frac{18}{529}\theta^2 \\ &\approx 0.35579 - 0.08281\theta + 0.034403\theta^2\end{aligned}$$

and, with $\theta \in [0, 1]$ the minimum is at $-0.08281 + 0.068805\theta = 0$, *i.e.*:

$$\hat{\theta} : \frac{0.08281}{0.06805} \approx 1.21683 \quad > 1.$$

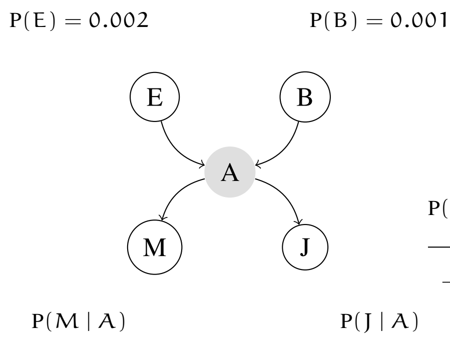
So, $\hat{\theta} = 1, \mathrm{err}\left(\hat{\theta}\right) \approx 0.30699.$

For $\gamma = 0.5$, the error function is

$$\begin{aligned}\mathrm{err}(\theta) &= \frac{10217413}{33062500} - \frac{2181}{66125}\theta + \frac{18}{529}\theta^2 \\ &\approx 0.30903 - 0.032298\theta + 0.034402\theta^2\end{aligned}$$

and, with $\theta \in [0, 1]$ the minimum is at $-0.03298 + 0.068804\theta = 0$, *i.e.*:

$$\begin{aligned}\hat{\theta} &\approx \frac{0.03298}{0.06804} \approx 0.48471 \approx \frac{1}{2}, \\ \mathrm{err}\left(\hat{\theta}\right) &\approx 0.30104\end{aligned}$$



$P(A \mid B \wedge E)$

		a	$\neg a$
b	e	0.95	0.05
b	$\neg e$	0.94	0.06
$\neg b$	e	0.29	0.71
$\neg b$	$\neg e$	0.001	0.999

$P(M \mid A)$

	m	$\neg m$
a	0.9	0.1
$\neg a$	0.05	0.95

$P(J \mid A)$

	j	$\neg j$
a	0.7	0.3
$\neg a$	0.01	0.99

Figure 4: The Earthquake, Burglary, Alarm model

These experiments show that data can indeed be used to estimate the parameters of the model. However, we observe that the estimated $\hat{\theta}$ has a tendency to over- or under-estimate the θ used to generate the samples. More precisely, in experiment 1 data is generated with $\gamma = 0.2$ (the surrogate of θ) which is under-estimated with $\hat{\theta} = 0$ while in experiment 2, $\gamma = 0.8$ leads the over-estimation $\hat{\theta} = 1$. This suggests that we might need to refine the error estimation process. However, experiment 3 data results from $\gamma = 0.5$ and we've got $\hat{\theta} \approx 0.48471 \approx 0.5$, which is more in line with what is to be expected.

1.2 An Example Involving Bayesian Networks

As it turns out, our framework is suitable to deal with more sophisticated cases, in particular cases involving bayesian networks. In order to illustrate this, in this section we see how the classical example of the Burglary, Earthquake, Alarm (Pearl 1988) works in our setting. This example is a commonly used example in bayesian networks because it illustrates reasoning under uncertainty. The gist of the example is given in fig. 4. It involves a simple network of events and conditional probabilities.

The events are: Burglary (B), Earthquake (E), Alarm (A), Mary calls (M) and John calls (J). The initial events B and E are assumed to be independent events that occur with probabilities $P(B)$ and $P(E)$, respectively. There is an alarm system that can be triggered by either of the initial events B and E. The probability of the alarm going off is a conditional probability given that B and E have occurred. One denotes these probabilities, as per usual, by $P(A \mid B)$, and $P(A \mid E)$. There are two neighbors, Mary and John who have agreed to call if they hear the alarm. The probability that they do actually call is also a conditional probability denoted by $P(M \mid A)$ and $P(J \mid A)$, respectively.

We follow the convention of representing the (upper case) random variable X by the (lower case) positive literal x. Considering the probabilities given in fig. 4 we obtain the following specification:

$$b : 0.001,$$
$$e : 0.002,$$

For the table giving the probability $P(M \mid A)$ we obtain the program:

$$\begin{aligned} p_{m|a} &: 0.9, \\ p_{m|\overline{a}} &: 0.05, \\ m &\leftarrow a \wedge p_{m|a}, \\ m &\leftarrow \neg a \wedge p_{m|\overline{a}}. \end{aligned}$$

The latter program can be simplified (abusing notation) by writing $m : 0.9 \leftarrow a$ and $m : 0.05 \leftarrow \overline{a}$.

Similarly, for the probability $P(J \mid A)$ we obtain

$$\begin{aligned} j &: 0.7 \leftarrow a, \\ j &: 0.01 \leftarrow \neg a, \end{aligned}$$

Finally, for the probability $P(A \mid B \wedge E)$ we obtain

$$\begin{aligned} a : 0.95 \leftarrow b \wedge e, &\quad a : 0.94 \leftarrow b \wedge \overline{e}, \\ a : 0.29 \leftarrow \overline{b} \wedge e, &\quad a : 0.001 \leftarrow \overline{b} \wedge \overline{e}. \end{aligned}$$

One can then proceed as in the previous subsection and analyze this example. The details of such analysis are not given here since they are analogous, albeit admittedly more cumbersome.

References

Geman, S. and Geman, D.: 1984, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-6**(6), 721–741.

Pearl, J.: 1988, *Probabilistic reasoning in intelligent systems: networks of plausible inference*, The Morgan Kaufmann Series in Representation and Reasoning, Morgan Kaufmann, San Mateo, CA.