Evidential Reasoning and Learning

Federico **Cerutti** • Lance M. **Kaplan**

University of Brescia (Italy) • US DEVCOM ARL (USA)

Please cite as: Federico Cerutti, Lance Kaplan, Murat Sensoy. Evidential Reasoning and Learning: a Survey. IJCAI 2022.

Announcements

Survey paper in the main conference

Federico Cerutti, Lance Kaplan, Murat Sensoy. Evidential Reasoning and Learning: a Survey.

Scheduled on July 28th at 1000h in Lehar 1 – (12 min talk)
Poster session 2, stand 318 row 9

University of Brescia, Italy, will open soon a 3-years RA/post-doc position on evidential reasoning and learning.

Introduction

Evidential Learning and Reasoning: quantifying aleatory and epistemic uncertainty in reasoning and learning while using very efficient approximations based upon the idea of updating the Bayesian posterior in light of further evidence collected in favour (or against) a hypothesis

> *Even in simple collaboration scenarios, e.g. those in which an AI system assists a human operator with predictions, the success of the team hinges on the human correctly deciding when to follow the recommendations of the AI system and when to override them. [...]*
>
> *Extracting benefits from collaboration with the AI system depends on the human developing insights (i.e., a mental model) of when to trust the AI system with its recommendations. [...]*
>
> *If the human mistakenly trusts the AI system in regions where it is likely to err, catastrophic failures may occur.*

Bansal, Gagan, et al. "Beyond Accuracy: The Role of Mental Models in Human-AI Team Performance." AAAI Conference on Human Computation and Crowdsourcing. 2019.

Guidelines for Human-AI Interaction

Initially Make clear what the system can do • Make clear how well the system can do what it can do

During interaction Time services based on context • Show contextually relevant information • Match relevant social norms • Mitigate social biases

When wrong Support efficient invocation • Support efficient dismissal • Support efficient correction • Scope services, when in doubt • Make clear why the system did what it did

Over time Remember recent interactions • Learn from user behaviour • Update and adapt cautiously • Encourage granular feedback • Convey the consequences of user actions • Provide global controls • Notify users about changes

https://aka.ms/aiguidelines

S. Amershi et. al., "Guidelines for Human-AI Interaction," CHI 2019

Misclassification of the white side of a trailer as bright sky: this caused a car operating with automated vehicle control systems (level 2) to crash against a tractor-semitrailer truck near Williston, Florida, USA on 7th May 2016.

The car driver died due to the sustained injury.

The car manufacturer stated that the "camera failed to recognize the white truck against a bright sky."*

*http://tiny.cc/2tb4uy

Requirements of Trustworthy AI

Human agency and oversight

Technical robustness and safety

Privacy and data governance

Transparency

Diversity, non-discrimination, and fairness

Societal and environmental wellbeing

Accountability

*EUROPEAN COMMISSION, 2019. High-Level Expert Group on Artificial Intelligence.

Outline

1. A primer in Bayesian Statistics:
   - Fundamentals of statistics and Bayes
   - Beta and Dirichlet distributions as uncertain probabilities.
2. Evidential Reasoning:
   - From logic to probabilistic circuits;
   - Probabilistic circuits as a unifying method for probabilistic reasoning;
   - Probabilistic circuits with uncertain probabilities.
3. Evidential Parameter Learning:
   - Learning with complete observations;
   - Learning with partial observations: preliminary proposals and discussions.
4. Ascertain Evidence from the Real World:
   - Intelligence analysis and uncertainty
   - Evidential Deep Learning;
   - Alternative proposals.
5. Summary and conclusion.

A primer in Bayesian Statistics

X (resp. Y) be a *discrete* random variable that can take values x_i with $i = 1, \ldots, M$ (resp. y_j with $j = 1, \ldots, L$).

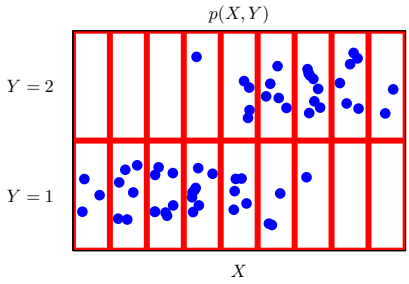
The probability that X will take the value x_i and Y will take the value y_j is written $p(X = x_i, Y = y_j)$ and is called the *joint probability* of $X = x_i$ and $Y = y_j$.

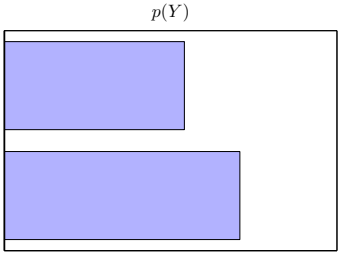
Sum rule or *marginalisation*:

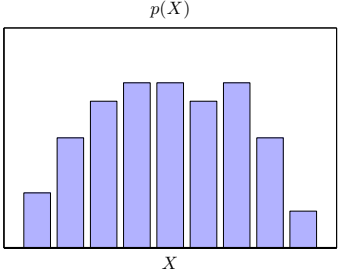
$$p(X = x_i) = \sum_{j=1}^{L} p(X = x_i, Y = y_j) \tag{5}$$

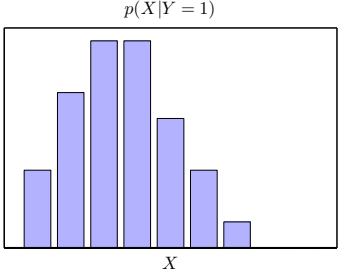
Product rule:

$$p(X = x_i, Y = y_j) = p(Y = y_j | X = x_i) p(X = x_i) \tag{6}$$









*Figs from Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Berlin, Heidelberg:

Sum and *product* rules apply to general random variables, not only discrete ones.

$$p(X) = \sum_{Y} p(X, Y) \tag{7}$$

$$p(X, Y) = p(Y|X)p(X) \tag{8}$$

X and Y are said to be independent if $p(X, Y) = p(X)p(Y)$, i.e. $p(X|Y) = p(X)$ and $p(Y|X) = p(Y)$.

X be a *continuous* random variable

$$p(x \in (a, b)) = \int_a^b p(x)\mathrm{d}x \tag{9}$$

$$p(x) \geq 0 \tag{10}$$

$$\int_{-\infty}^{\infty} p(x)\mathrm{d}x = 1 \tag{11}$$

The *cumulative distribution function* is defined by:

$$P(z) = \int_{-\infty}^{z} p(x)\mathrm{d}x \tag{12}$$

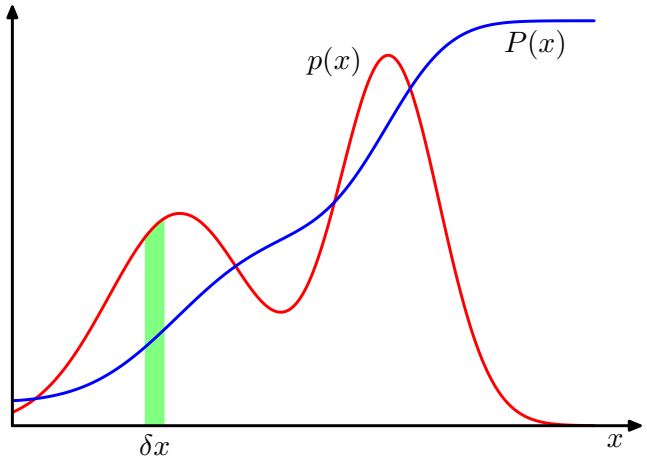


*Fig from Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Berlin, Heidelberg: Springer-Verlag.

Given several continuous variables $\boldsymbol{x} = \langle x_1, \ldots, x_D \rangle^{\mathrm{T}}$, then we can define a joint probability density $p(\boldsymbol{x}) = p(x_1, \ldots, x_D)$ such that the probability of $\boldsymbol{x}$ falling in an infinitesimal volume $\delta\boldsymbol{x}$ containing the point $\boldsymbol{x}$ is given by $p(\boldsymbol{x})\delta\boldsymbol{x}$, and

$$p(\boldsymbol{x}) \geq 0 \tag{13}$$

and

$$\int p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 1 \tag{14}$$

Sum and *product* rules for continuous random variables take the form:

$$p(x) = \int p(x, y) \mathrm{d}y \tag{15}$$

$$p(x, y) = p(y|x)p(x) \tag{16}$$

The weighted average of the function $f(x)$ under a probability distribution $p(x)$, or *expectation* of $f(x)$ is:

$$\mathbb{E}[f] = \sum_x p(x)f(x) \tag{17}$$

$$\mathbb{E}[f] = \int p(x)f(x)\mathrm{d}x \tag{18}$$

It can be approximate from N points drawn from the distribution

$$\mathbb{E}[f] \simeq \frac{1}{N}\sum_{n=1}^{N} f(x_n) \tag{19}$$

In the case of functions of several variables,

$$\mathbb{E}_x[f(x,y)] = \sum_x p(x)f(x,y) \tag{20}$$

Variance of $f(x)$

$$\text{var}[f] = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2] \tag{22}$$

$$\text{var}[f] = \mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2 \tag{23}$$

For two random variables, the *covariance* expresses the extent to which they vary together, and is defined by:

$$\text{cov}[x, y] = \mathbb{E}_{x,y}[xy] - \mathbb{E}[x]\mathbb{E}[y] \tag{24}$$

In the case of vectors of random variables:

$$\text{cov}[\boldsymbol{x}, \boldsymbol{y}] = \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[\boldsymbol{x}\boldsymbol{y}^{\mathrm{T}}] - \mathbb{E}[\boldsymbol{x}]\mathbb{E}[\boldsymbol{y}^{\mathrm{T}}] \tag{25}$$

Note that $\text{cov}[\boldsymbol{x}] \equiv \text{cov}[\boldsymbol{x}, \boldsymbol{x}]$.

Bayes theorem

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \tag{26}$$

where

$$p(X) = \sum_{Y} p(X|Y)p(Y) \tag{27}$$

$p(H)$ Probability that a person with no known risk behaviour is infected with HIV (base rate): 0.01%

$p(T \mid H)$ Probability that a test returns positive if the person is infected with HIV (sensitivity): 99.9%

$p(\overline{T} \mid \overline{H})$ Probability that a test returns negative if the person is not infected with HIV (specificity): 99.99%

*Gerd Gigerenzer, Reckoning With Risk: Learning to Live With Uncertainty, Gardners Books, 2003

$p(H)$ Probability that a person with no known risk behaviour is infected with HIV (base rate): 0.01%

$p(T \mid H)$ Probability that a test returns positive if the person is infected with HIV (sensitivity): 99.9%

$p(\overline{T} \mid \overline{H})$ Probability that a test returns negative if the person is not infected with HIV (specificity): 99.99%

$$
\begin{aligned}
p(H \mid T) &= \frac{p(T \mid H) \cdot p(H)}{p(T)} \\
&= \frac{p(T \mid H) \cdot p(H)}{p(T \mid H)p(H) + p(T \mid \overline{H}) \cdot p(\overline{H})} \\
&= \frac{1}{1 + \frac{1}{p(T|H) \cdot p(H)} \cdot (1 - p(\overline{T} \mid \overline{H})) \cdot (1 - p(H))}
\end{aligned}
$$

*Gerd Gigerenzer, Reckoning With Risk: Learning to Live With Uncertainty, Gardners Books, 2003

$p(H)$ Probability that a person with no known risk behaviour is infected with HIV (base rate): 0.01%

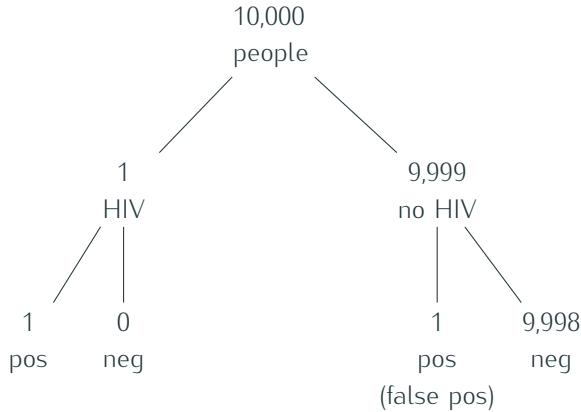
$p(T \mid H)$ Probability that a test returns positive if the person is infected with HIV (sensitivity): 99.9%

$p(\overline{T} \mid \overline{H})$ Probability that a test returns negative if the person is not infected with HIV (specificity): 99.99%

$$
\begin{aligned}
p(H \mid T) &= \frac{1}{1 + \frac{1}{p(T \mid H) \cdot p(H)} \cdot (1 - p(\overline{T} \mid \overline{H})) \cdot (1 - p(H))} \\
&= \frac{1}{1 + \frac{1}{\frac{999}{10^3} \cdot \frac{1}{10^4}} \cdot (1 - \frac{9999}{10^4}) \cdot (1 - \frac{1}{10^4})} = \frac{1}{1 + \frac{10^8}{9990} \cdot \frac{1}{10^4} \cdot \frac{9999}{10^4}} \\
&= \frac{1}{1 + \frac{9999}{9990}} \simeq \frac{1}{2}
\end{aligned}
$$

*Gerd Gigerenzer, Reckoning With Risk: Learning to Live With Uncertainty, Gardners Books, 2003

Imagine 10,000 people who are not in any known risk category. One is infected (base rate) and will test positive with practical certainty (sensitivity). Of the 9,999 people who are not infected, another one will also test positive (false positive rate). So we can expect that about two people will test positive.



*Gerd Gigerenzer, Reckoning With Risk: Learning to Live With Uncertainty, Gardners Books, 2003

Given the parameters of our model $\boldsymbol{w}$, we can capture our assumptions about $\boldsymbol{w}$, before observing the data, in the form of a prior probability distribution $p(\boldsymbol{w})$. The effect of the observed data $\mathcal{D} = \{t_1, \ldots, t_N\}$ is expressed through the conditional $p(\mathcal{D}|\boldsymbol{w})$, hence Bayes theorem takes the form:

$$p(\boldsymbol{w}|\mathcal{D}) = \frac{\overbrace{p(\mathcal{D}|\boldsymbol{w})}^{\text{likelihood}}\overbrace{p(\boldsymbol{w})}^{\text{prior}}}{p(\mathcal{D})} \tag{29}$$

$$\text{posterior} \propto \text{likelihood} \cdot \text{prior} \tag{30}$$

$$p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{w})p(\boldsymbol{w})\mathrm{d}\boldsymbol{w} \tag{31}$$

It ensures that the posterior distribution on the left-hand side is a valid probability density and integrates to one.

Frequentist paradigm

- $\boldsymbol{w}$ is considered to be a fixed parameter, whose values is determined by some form of *estimator*, e.g. the *maximum likelihood* in which $\boldsymbol{w}$ is set to the value that maximises $p(\mathcal{D}|\boldsymbol{w})$
- Error bars on this estimate are obtained by considering the distribution of possible data sets $\mathcal{D}$.
- The negative log of the likelihood function is called an *error function*: the negative log is a monotonically decreasing function hence maximising the likelihood is equivalent to minimising the error.

Bayesian paradigm

- There is only one single data set $\mathcal{D}$ (the one observed) and the uncertainty in the parameters is expressed through a probability distribution over $\boldsymbol{w}$.
- The inclusion of prior knowledge arises naturally: suppose that a fair-looking coin is tossed three times and lands heads each time. A classical maximum likelihood estimate of the probability of landing heads would give 1.
  There are cases where you want to reduce the dependence on the prior, hence using *noninformative* priors.

Binary variable: Bernoulli

Let us consider a single binary random variable $x \in \{0, 1\}$, e.g. flipping coin, not necessary fair, hence the probability is conditioned by a parameter $0 \leq \mu \leq 1$:

$$p(x = 1|\mu) = \mu \tag{32}$$

The probability distribution over x is known as the *Bernoulli* distribution:

$$\text{Bern}(x|\mu) = \mu^x(1 - \mu)^{1-x} \tag{33}$$

$$\mathbb{E}[x] = \mu \tag{34}$$

Now suppose that we have a data set of observations $\boldsymbol{x} = \langle x_1, \ldots, x_N \rangle^{\mathsf{T}}$ drawn independently from a Bernoulli distribution (iid) whose mean μ is unknown, and we would like to determine this parameter from the data set.

$$p(\mathcal{D}|\mu) = \prod_{n=1}^{N} p(x_n|\mu) = \prod_{n=1}^{N} \mu^{x_n}(1-\mu)^{1-x_n} \tag{36}$$

Let's maximise the (log)-likelihood to identify the parameter (log simplifies and reduces risks of underflow):

$$\ln p(\mathcal{D}|\mu) = \sum_{n=1}^{N} \ln p(x_n|\mu) = \sum_{n=1}^{N} \{x_n \ln \mu + (1-x_n)\ln(1-\mu)\} \tag{37}$$

The log likelihood depends on the N observations x_n only through their sum $\sum_n x_n$, hence the sum provides an example of a *sufficient statistics* for the data under this distribution

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\mu} \ln p(\mathcal{D}|\mu) &= 0 \\
\sum_{n=1}^{N} \frac{x_n}{\mu} - \frac{1-x_n}{1-\mu} &= 0 \\
\sum_{n=1}^{N} \frac{x_n - \mu}{\mu(1-\mu)} &= 0 \\
\sum_{n=1}^{N} x_n &= N\mu
\end{aligned}
\tag{38}
$$

$$
\mu_{\mathrm{ML}} = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{39}
$$

aka *sample mean*. Risk of overfit: consider to toss the coin three times and each time is head

In order to develop a Bayesian treatment to the overfit problem of the maximum likelihood estimator for the Bernoulli. Since the likelihood takes the form of the product of factors of the form $\mu^x(1-\mu)^{1-x}$, if we choose a prior to be proportional to powers of μ and $(1-\mu)$ then the posterior distribution, proportional to the product of the prior and the likelihood, will have the same functional form as the prior. This property is called *conjugacy*.

Binary variables: Beta distribution

$$\text{Beta}(\mu|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\mu^{a-1}(1-\mu)^{b-1} \tag{43}$$

with

$$\Gamma(x) \equiv \int_0^\infty u^{x-1}e^{-u}\mathrm{d}u \tag{44}$$

$$\mathbb{E}[\mu] = \frac{a}{a+b} \tag{45}$$

$$\text{var}[\mu] = \frac{ab}{(a+b)^2(a+b+1)} \tag{46}$$

a and b are *hyperparameters* controlling the distribution of parameter μ.

Considering a beta distribution prior and the Bernulli likelihood function, and given $l = N - m$

$$p(\mu|m, l, a, b) \propto \mu^{m+a-1}(1-\mu)^{l+b-1} \tag{47}$$

Hence $p(\mu|m, l, a, b)$ is another beta distribution and we can rearrange the normalisation coefficient as follows:

$$p(\mu|m, l, a, b) = \frac{\Gamma(m+a+l+b)}{\Gamma(m+a)\Gamma(l+b)}\mu^{m+a-1}(1-\mu)^{l+b-1} \tag{48}$$

# Day	Phenomenon
1	T
2	T
3	F
4	F
5	F
6	F
7	F
8	F
9	F
10	F

π: true—unknown—probability of the phenomenon in a given period of time

Let y be the number of occurrence of the phenomenon per period of time ($m = 2$)

Likelihood: $f(y|\pi) = \pi \cdot \pi \cdot (1-\pi) \cdots (1-\pi) = \pi^2 \cdot (1-\pi)^8$

$g(\pi|y) \propto g(\pi) \cdot f(y|\pi)$

The conjugate of a binomial is the Beta distribution. If:

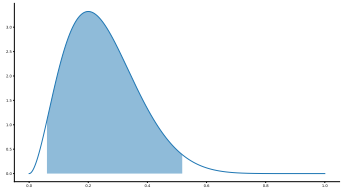
$g(\pi; a, b) = Beta(a, b) = \frac{\Gamma(a+b)}{\Gamma(a)+\Gamma(b)} \pi^{a-1}(1-\pi)^{b-1}$

then: $g(\pi|y) = Beta(m+a, l+b)$

If $a = b = 1$ (uniform prior), then $\boxed{g(\pi|y) = Beta(m+1, l+1)}$

In the example, $g(\pi|m = 2, l = 8) = Beta(3, 9)$

$X_1 \sim Beta(3, 9)$

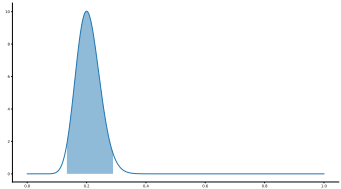


$E[X_1] = 0.2500$

$Var(X_1) = 1.4423 \cdot 10^{-2}$

95% Confidence Interval:
[0.0602, 0.5178]

$X_2 \sim Beta(21, 81)$

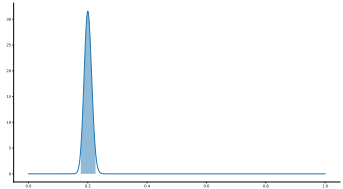


$E[X_2] = 0.2059$

$Var(X_2) = 1.5873 \cdot 10^{-3}$

95% Confidence Interval:
[0.1336, 0.2891]

$X_3 \sim Beta(201, 801)$



$E[X_3] = 0.2006$

$Var(X_3) = 1.5988 \cdot 10^{-4}$

95% Confidence Interval:
[0.1764, 0.2259]

Although $E[X_1] \simeq E[X_2] \simeq E[X_3] \simeq 0.2$

they represent remarkably different random variables

* Y-axes of the graphs are misaligned for better graphical representation.

Epistemic vs Aleatoric uncertainty

Aleatoric uncertainty

Variability in the outcome of an experiment which is due to inherently random effects (e.g. flipping a fair coin): no additional source of information but Laplace's daemon can reduce such a variability.

Epistemic uncertainty

Epistemic state of the agent using the model, hence its lack of knowledge that—in principle—can be reduced on the basis of additional data samples.

It is a general property of Bayesian learning that, as we observe more and more data, the epistemic uncertainty represented by the posterior distribution will steadily decrease (the variance decreases).

https://tinyurl.com/5hets659

Multinomial variables: categorical distribution

Let us suppose to roll a dice with $K = 6$ faces. An observation of this variable $\boldsymbol{x}$ equivalent to $x_3 = 1$ (e.g. the number 3 with face up) can be:

$$\boldsymbol{x} = \langle 0, 0, 1, 0, 0, 0 \rangle^{\mathsf{T}} \tag{49}$$

Note that such vectors must satisfy $\sum_{k=1}^{K} x_k = 1$.

$$p(\boldsymbol{x}|\boldsymbol{\mu}) = \prod_{k=1}^{K} \mu_k^{x_k} \tag{50}$$

where $\boldsymbol{\mu} = \langle \mu_1, \ldots, \mu_K \rangle^{\mathsf{T}}$, nad the parameters μ_k are such that $\mu_k \geq 0$ and $\sum_k \mu_k = 1$.

Generalisation of the Bernoulli

$$p(\mathcal{D}|\boldsymbol{\mu}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \mu_k^{x_{nk}} \tag{52}$$

The likelihood depends on the N datapoints only through the K quantities

$$m_k = \sum_n x_{nk} \tag{53}$$

which represent the number of observations of $x_k = 1$ (e.g. with k = 3, the third face of the dice). These are called the *sufficient statistics* for this distribution.

Finding the maximum likelihood requires a Lagrange multiplier that

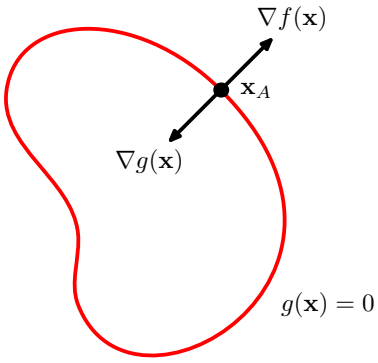
$$\sum_{k=1}^{K} m_k \ln \mu_k + \lambda \left(\sum_{k=1}^{K} \mu_k - 1 \right) \tag{54}$$

Hence

$$\mu_k^{\mathrm{ML}} = \frac{m_k}{N} \tag{55}$$

which is the fraction of N observations for which $x_k = 1$.

Lagrange multiplier



$\nabla g(\boldsymbol{x})$ must be orthogonal to the surface.
Consider a point $\boldsymbol{x}$ that lies on the surface and a nearby point $\boldsymbol{x} + \boldsymbol{\epsilon}$ that also lies on the surface.
The Taylor expansion around $\boldsymbol{x}$ gives
$g(\boldsymbol{x} + \boldsymbol{\epsilon}) \simeq g(\boldsymbol{x}) + \boldsymbol{\epsilon}^{\mathsf{T}} \nabla g(\boldsymbol{x})$.
Because both $\boldsymbol{x}$ and $\boldsymbol{x} + \boldsymbol{\epsilon}$ lie on the surface, $g(\boldsymbol{x}) \simeq g(\boldsymbol{x} + \boldsymbol{\epsilon})$, hence $\boldsymbol{\epsilon}^{\mathsf{T}} \nabla g(\boldsymbol{x}) \simeq 0$. Because $\boldsymbol{\epsilon}$ is parallel to the surface, ∇g must be normal to the surface.

Given the point $\boldsymbol{x}^*$, lying on the surface, be a maximum for f. ∇f must also be normal to the surface.

Thus ∇f and ∇g are parallel (or anti-parallel) vectors, and so there must exist a parameter λ such that $\nabla f + \lambda \nabla g = 0$.

*Fig from Bishop, Christopher M. 2006. Pattern Recognition and Machine Learning. Berlin, Heidelberg: Springer-Verlag.

Multinomial variables: the Dirichlet distribution

The *Dirichlet* distribution is the generalisation of the beta distribution to K dimensions.

$$\mathrm{Dir}(\boldsymbol{\mu}|\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\cdots\Gamma(\alpha_K)} \prod_{k=1}^{K} \mu_k^{\alpha_k - 1} \tag{56}$$

such that $\sum_k \mu_k = 1$, $\boldsymbol{\alpha} = \langle \alpha_1, \ldots, \alpha_K \rangle^{\mathsf{T}}$, $\alpha_k \geq 0$ and

$$\alpha_0 = \sum_{k=1}^{K} \alpha_k \tag{57}$$

Considering a Dirichlet distribution prior and the categorical likelihood function, the posterior is then:

$$\begin{aligned} p(\boldsymbol{\mu}|\mathcal{D}, \boldsymbol{\alpha}) &= \text{Dir}(\boldsymbol{\mu}|\boldsymbol{\alpha} + \boldsymbol{m}) = \\ &= \frac{\Gamma(\alpha_0 + N)}{\Gamma(\alpha_1 + m_1) \cdots \Gamma(\alpha_K + m_K)} \prod_{k=1}^{K} \mu_k^{\alpha_k + m_k - 1} \end{aligned} \tag{58}$$

The uniform prior is given by $\text{Dir}(\boldsymbol{\mu}|\mathbf{1})$ and the Jeffreys' non-informative prior is given by $\text{Dir}(\boldsymbol{\mu}|\langle 0.5, \ldots, 0.5\rangle^{\mathsf{T}})$.

The marginals of a Dirichlet distribution are beta distributions.

Evidential Reasoning

```
burglary.
earthquake.
hears_alarm(john).
alarm :- burglary.
alarm :- earthquake.
calls(john) :- alarm, hears_alarm(john).
evidence(calls(john)).
query(burglary).
```

$$\texttt{alarm} \leftrightarrow \texttt{burglary} \vee \texttt{earthquake}$$
$$\texttt{calls(john)} \leftrightarrow \texttt{alarm} \wedge \texttt{hears\_alarm(john)}$$
$$\texttt{calls(john)}$$

*D. Fierens, *et. al.*. 'Inference and Learning in Probabilistic Logic Programs Using Weighted Boolean Formulas'. TPLP 2015.

$$\texttt{alarm} \leftrightarrow \texttt{burglary} \vee \texttt{earthquake}$$
$$\texttt{calls(john)} \leftrightarrow \texttt{alarm} \wedge \texttt{hears\_alarm(john)}$$
$$\texttt{calls(john)}$$

$\texttt{alarm} \rightarrow \texttt{burglary} \vee \texttt{earthquake}$
$\texttt{alarm} \leftarrow \texttt{burglary} \vee \texttt{earthquake}$

$\texttt{calls(john)} \rightarrow \texttt{alarm}$
$\texttt{calls(john)} \rightarrow \texttt{hear\_alarm(john)}$
$\texttt{calls(john)} \leftarrow \texttt{alarm} \wedge \texttt{hear\_alarm(john)}$
$\texttt{calls(john)}$

$\neg\texttt{alarm} \vee \texttt{burglary} \vee \texttt{earthquake}$
$\neg\texttt{burglary} \vee \texttt{alarm}$
$\neg\texttt{earthquake} \vee \texttt{alarm}$
$\neg\texttt{calls(john)} \vee \texttt{alarm}$
$\neg\texttt{calls(john)} \vee \texttt{hear\_alarm(john)}$
$\neg\texttt{alarm} \vee \neg\texttt{hear\_alarm(john)} \vee \texttt{calls(john)}$
$\texttt{calls(john)}$

CNF

$$
\begin{aligned}
&(\texttt{alarm} \vee \neg\texttt{burglary}) \wedge \\
&(\texttt{alarm} \vee \neg\texttt{calls(john)}) \wedge \\
&(\texttt{alarm} \vee \neg\texttt{earthquake}) \wedge \\
&(\texttt{hear\_alarm(john)} \vee \neg\texttt{calls(john)}) \wedge \\
&(\texttt{burglary} \vee \texttt{earthquake} \vee \neg\texttt{alarm}) \wedge \\
&(\texttt{calls(john)} \vee \neg\texttt{alarm} \vee \neg\texttt{hear\_alarm(john)}) \wedge \\
&\texttt{calls(john)}
\end{aligned}
$$

NNF

$$
\begin{aligned}
&(\texttt{alarm} \vee (\neg\texttt{burglary} \wedge \neg\texttt{earthquake})) \wedge \\
&((\texttt{alarm} \wedge \texttt{hear\_alarm(john)}) \vee \neg\texttt{calls(john)}) \wedge \\
&(\texttt{burglary} \vee \texttt{earthquake} \vee \neg\texttt{alarm}) \wedge \\
&(\texttt{calls(john)} \vee \neg\texttt{alarm} \vee \neg\texttt{hear\_alarm(john)}) \wedge \\
&\texttt{calls(john)}
\end{aligned}
$$

A sentence in *negation normal form* (**NNF**) over a set of propositional variables $\mathcal{V}$ is a rooted, directed acyclic graph where each leaf node is labeled with true ($\top$), false ($\bot$), or a literal of a variable in $\mathcal{V}$, and each internal node with disjunction ($\vee$) or conjunction ($\wedge$).

Decomposable: for each conjunction node no two children ϕ_i and ϕ_j share any variable.

Deterministic: for each disjunction node each pair of different children ϕ_i and ϕ_j is logically contradictory, that is $\phi_i \wedge \phi_j \models \bot$ for $i \neq j$; i.e., only one child can be true at any time.

Smooth: for each disjunction nodeeach disjunct ϕ_i mentions the same variables, $\mathit{Vars}(\phi_i) = \mathit{Vars}(\phi_j)$ for $i \neq j$.

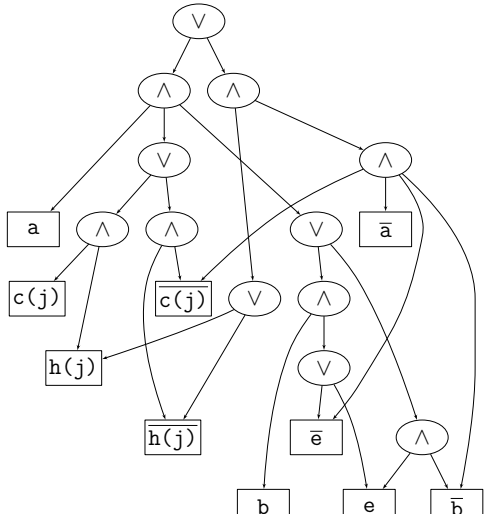
It is hard to ensure decomposability. It is also hard to ensure determinism while preserving decomposability. Yet any sentence in **NNF** can be smoothed in polytime, while preserving decomposability and determinism.

*Darwiche and Marquis, A Knowledge Compilation Map, JAIR 17 (2002) 229–264

Decomposable: for each conjunction node no two children ϕ_i and ϕ_j share any variable.

Deterministic: for each disjunction node each pair of different children ϕ_i and ϕ_j is logically contradictory, that is $\phi_i \wedge \phi_j \models \bot$ for $i \neq j$; i.e., only one child can be true at any time.

Smooth: for each disjunction node each disjunct ϕ_i mentions the same variables, $\mathit{Vars}(\phi_i) = \mathit{Vars}(\phi_j)$ for $i \neq j$.



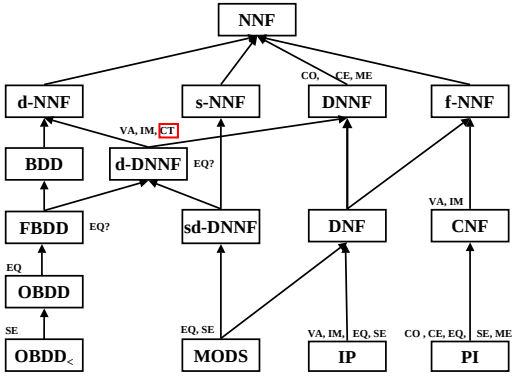
*Darwiche and Marquis, A Knowledge Compilation Map, JAIR 17 (2002) 229–264

Model counting (CT) is important

Given a query $q \subseteq \mathcal{L}$ and $\mathcal{I}(q) = \{I \mid I \in \mathcal{M}(\mathcal{T}) \ \wedge \ q \subseteq I\}$ the set of interpretations where the query is true, then the probabilistic inference task is:

$$\text{PROB}(q) = \sum_{I \in \mathcal{I}(q)} \prod_{l \in I} p(l). \tag{1}$$

*A. Kimmig, G. Van den Broeck, and L. De Raedt. Algebraic model counting. Journal of Applied Logic, 22:46–62, 2017.

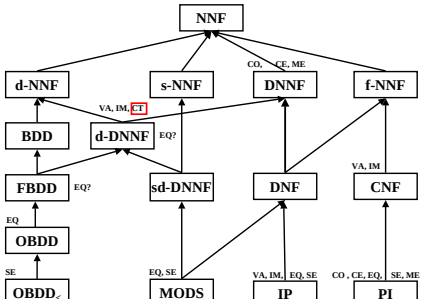


$L_1 \rightarrow L_2$ means that L_1 is a proper subset of L_2.
Next to each subset, polytime queries supported by the subset but not by any of its proper supersets.

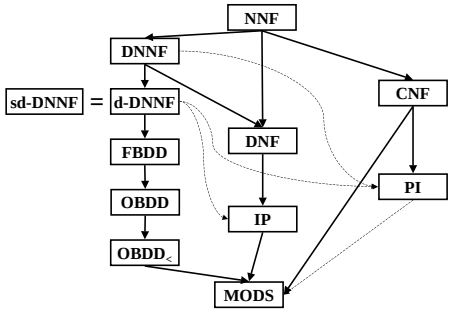
Notation	Query
CO	polytime consistency check
VA	polytime validity check
CE	polytime clausal entailment check
IM	polytime implicant check
EQ	polytime equivalence check
SE	polytime sentential entailment check
CT	polytime model counting
ME	polytime model enumeration

L	CO	VA	CE	IM	EQ	SE	CT	ME
NNF	○	○	○	○	○	○	○	○
DNNF	√	○	√	○	○	○	○	√
d-NNF	○	○	○	○	○	○	○	○
s-NNF	○	○	○	○	○	○	○	○
f-NNF	○	○	○	○	○	○	○	○
d-DNNF	√	√	√	√	?	○	√	√
sd-DNNF	√	√	√	√	?	○	√	√
BDD	○	○	○	○	○	○	○	○
FBDD	√	√	√	√	?	○	√	√
OBDD	√	√	√	√	√	○	√	√
OBDD$_<$	√	√	√	√	√	√	√	√
DNF	√	○	√	○	○	○	○	√
CNF	○	√	○	√	○	○	○	○
PI	√	√	√	√	√	√	○	√
IP	√	√	√	√	√	√	○	√
MODS	√	√	√	√	√	√	√	√

*Darwiche and Marquis, A Knowledge Compilation Map, JAIR 17 (2002) 229–264



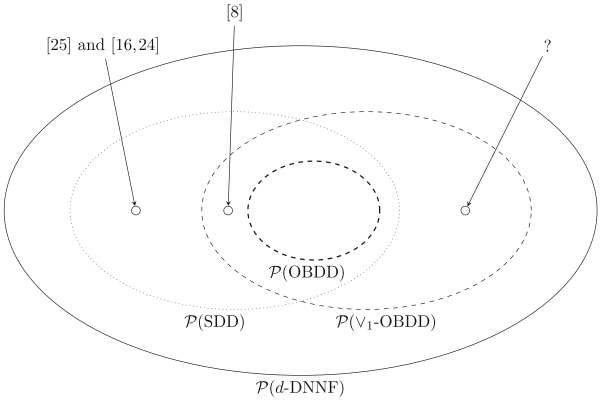
$L_1 \rightarrow L_2$ means that L_1 is a proper subset of L_2.
Next to each subset, polytime queries supported by the subset but not by any of its proper supersets.



$L_1 \rightarrow L_2$ indicates that L_1 is strictly more succint (space efficient) than L_2. Dotted arrows indicate unkwnown relationships.

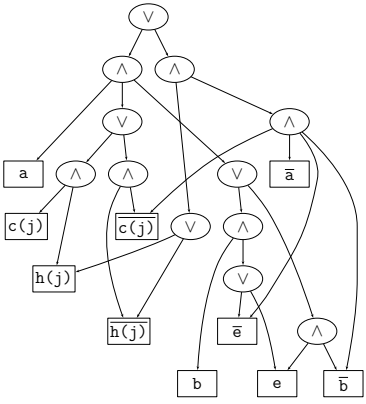
*Darwiche and Marquis, A Knowledge Compilation Map, JAIR 17 (2002) 229–264

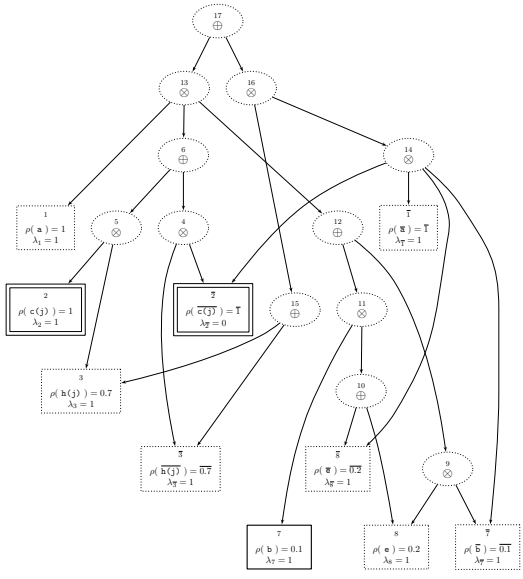
Sentential Decision Diagrams



SDDs provide a canonical representation of a propositional sentence, i.e. there is a unique SDD for any propositional sentence under a given variable order.
Manipulating SDDs is thus easier.
However, they are less succint than `d-DNF`s.

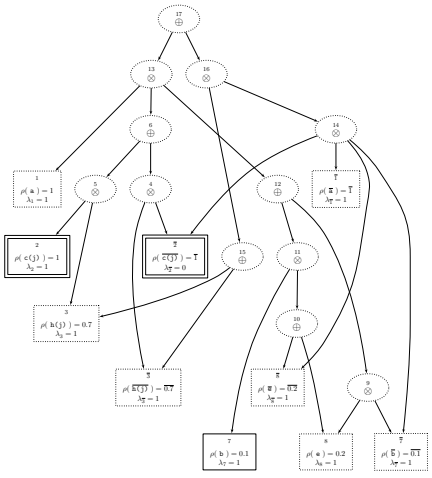
*Fig. from Bollig, B., Buttkus, M. On the Relative Succinctness of Sentential Decision Diagrams. Theory Comput Syst 63, 1250–1277 (2019).

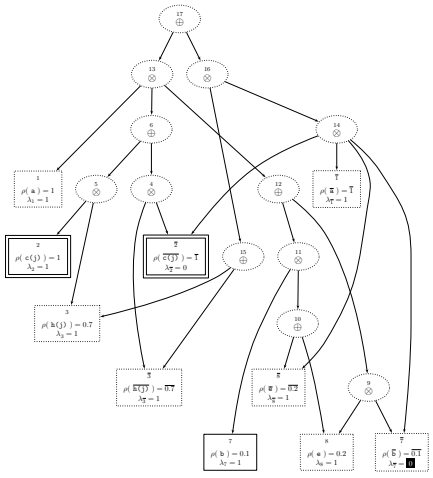





```
0.1::burglary.          ...
0.2::earthquake.        evidence(calls(john)).
0.7::hears_alarm(john). query(burglary).
```

*D. Fierens, *et. al.*. 'Inference and Learning in Probabilistic Logic Programs Using Weighted Boolean Formulas'. TPLP 2015.





```
0.1::burglary.    0.7::hears_alarm(john).  evidence(calls(john)).
0.2::earthquake.  ...                      query(burglary).
```

$$p(\texttt{burglary} \mid \texttt{calls(john)}) = \frac{p(\texttt{burglary} \wedge \texttt{calls(john)})}{p(\texttt{calls(john)})}$$

*D. Fierens, *et. al.*. 'Inference and Learning in Probabilistic Logic Programs Using Weighted Boolean Formulas'. TPLP 2015.

Connection with Sum-Product Networks

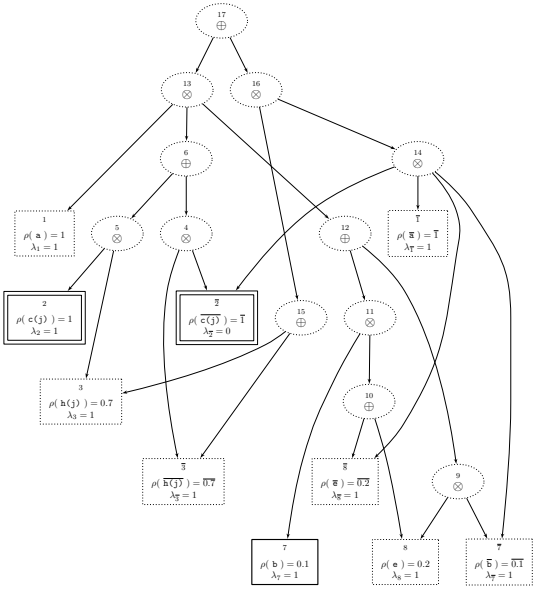
image: Freepik.com

An SPN consists of a rooted, directed, acyclic graph. Each leaf in the SPN graph is a tractable distribution over a single random variable. Each interior node is either a sum node, which computes a weighted sum of its children in the graph, or a product node, which computes the product of its children.

For discrete domains, every decomposable and smooth NNF can be represented as an equivalent SPN with fewer or equal nodes and edges.

1. If the root is not a sum node, set the root to a new sum node whose single child is the former root.
2. For each sum node, set the initial weights of all outgoing edges to 1.
3. For each leaf, find the first sum node on each path to the root and multiply its outgoing edge weight along that path by the parameter value. (Do not multiply the same edge weight by any given parameter more than once, even if that edge occurs in multiple paths to the root.)
4. Replace each leaf with a deterministic univariate distribution, $P(v) = 1$.

*Rooshenas and Lowd, Learning Sum-Product Networks with Direct and Indirect Variable Interactions, ICML2014.


17 ⊕
13 ⊗
16 ⊗
6 ⊕
14 ⊗
1 ρ(a) = 1 λ1 = 1
5 ⊗
4 ⊗
12 ⊕
T̄ ρ(x) = T̄ λT̄ = 1
2 ρ(c(j)) = 1 λ2 = 1
x̄ ρ(c(j)) = T̄ λx̄ = 0
15 ⊕
11 ⊗
3 ρ(b(j)) = 0.7 λ3 = 1
10 ⊕
b̄ ρ(b(j)) = 0.7 λb̄ = 1
x̄ ρ(x) = 0.2 λx̄ = 1
9 ⊗
7 ρ(b) = 0.1 λ7 = 1
8 ρ(e) = 0.2 λ8 = 1
b̄ ρ(b) = 0.1 λb̄ = 1

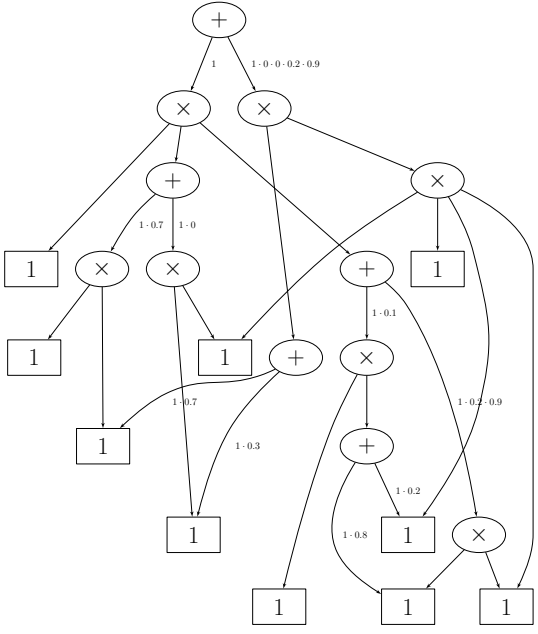

+
1
1 · 0 · 0 · 0.2 · 0.9
×
×
+
×
1 · 0.7
1 · 0
1
×
×
+
1
1
1
+
×
1 · 0.1
1
1 · 0.7
1 · 0.2 · 0.9
1
1 · 0.3
+
1 · 0.2
1 · 0.8
1
×
1
1
1

image: Freepik.com

More and more research in the field...

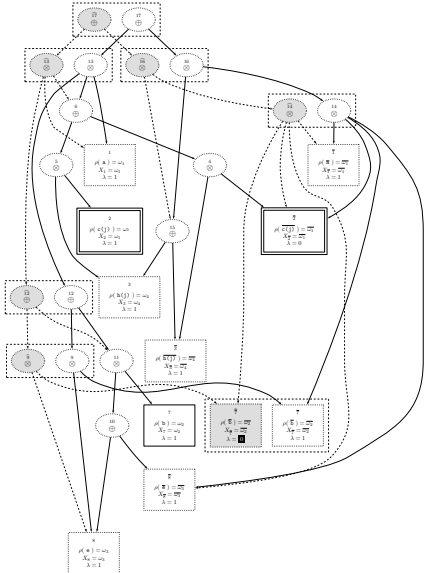
Kisa, D., Van den Broeck, G., Choi, A. and Darwiche, A., 2014, May. Probabilistic sentential decision diagrams. In Fourteenth International Conference on the Principles of Knowledge Representation and Reasoning.

Dang, M., Vergari, A., Broeck, G.. (2020). Strudel: Learning Structured-Decomposable Probabilistic Circuits. Proceedings of the 10th International Conference on Probabilistic Graphical Models, 138:137-148.

```
ω2 :: burglary.
ω3 :: earthquake.
ω4 :: hears_alarm(john).
alarm :- burglary.
alarm :- earthquake.
calls(john) :- alarm, \
       hears_alarm(john).
evidence(calls(john)).
query(burglary).
```

Identifier	Beta parameters	
ω_1	Beta$(\infty, 1)$	*False*
$\overline{\omega_1}$	Beta$(1, \infty)$	*True*
ω_2	Beta$(2, 18)$	*A burglary happened*
$\overline{\omega_2}$	Beta$(18, 2)$	*A burglary did not happen*
ω_3	Beta$(2, 8)$	...
$\overline{\omega_3}$	Beta$(8, 2)$	
ω_4	Beta$(3.5, 1.5)$	
$\overline{\omega_4}$	Beta$(1.5, 3.5)$	

*Cerutti, Kaplan, Kimmig, Şensoy, Handling Epistemic and Aleatory Uncertainties in Probabilistic Circuits, Machine Learning, 2022



Let n be a $\oplus$-gate over C nodes, its children

...

Let n be a $\otimes$-gate over C nodes, its children

$$\mathbb{E}[X_n] = \prod_{c \in C} \mathbb{E}[X_c],$$

$$\mathrm{cov}[X_n] \simeq \sum_{c \in C} \sum_{c' \in C} \frac{\mathbb{E}[X_n]^2}{\mathbb{E}[X_c]\mathbb{E}[X_{c'}]} \mathrm{cov}[X_c, X_{c'}],$$

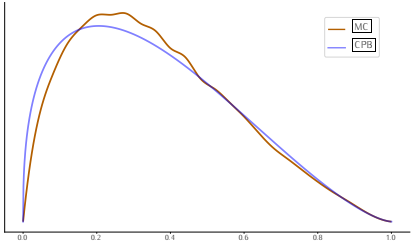
$$\mathrm{cov}[X_n, X_z] \simeq \sum_{c \in C} \frac{\mathbb{E}[X_n]}{\mathbb{E}[X_c]} \mathrm{cov}[X_c, X_z] \quad \text{for } z \in \widehat{N_A} \setminus \{n\}.$$

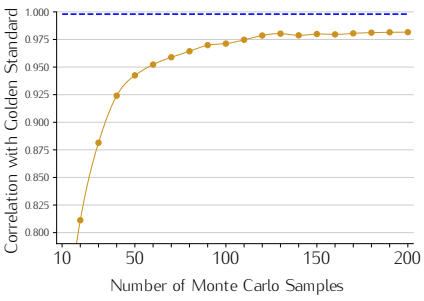
Conditioning:

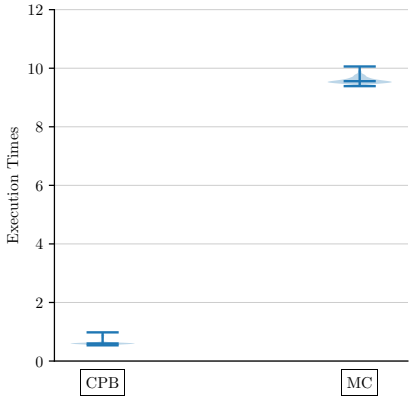
$$\mathbb{E}\left[\frac{X_r}{X_{\hat{r}}}\right] \simeq \frac{\mathbb{E}[X_r]}{\mathbb{E}[X_{\hat{r}}]},$$

$$\mathrm{cov}\left[\frac{X_r}{X_{\hat{r}}}\right] \simeq \frac{1}{\mathbb{E}[X_{\hat{r}}]^2}\mathrm{cov}[X_r] + \frac{\mathbb{E}[X_r]^2}{\mathbb{E}[X_{\hat{r}}]^4}\mathrm{cov}[X_{\hat{r}}] - 2\frac{\mathbb{E}[X_r]}{\mathbb{E}[X_{\hat{r}}]^3}\mathrm{cov}[X_r, X_{\hat{r}}].$$

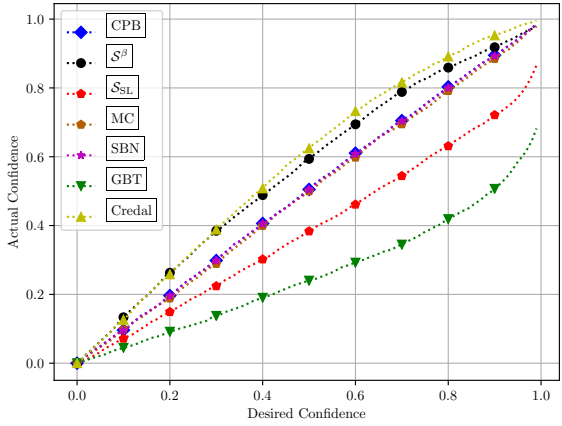
*Cerutti, Kaplan, Kimmig, Şensoy, Handling Epistemic and Aleatory Uncertainties in Probabilistic Circuits, Machine Learning, 2022







*Cerutti, Kaplan, Kimmig, Şensoy, Handling Epistemic and Aleatory Uncertainties in Probabilistic Circuits, Machine Learning, 2022



(best closest to the diagonal)

Experimental evaluation of the quality of the approximation in assessing epistemic uncertainty

It provides a methodology for calibration of the uncertainty estimation

Theoretical guarantees (or even whether they are possible) still unclear and requires further investigations

*Cerutti, Kaplan, Kimmig, Şensoy, Handling Epistemic and Aleatory Uncertainties in Probabilistic Circuits, Machine Learning, 2022

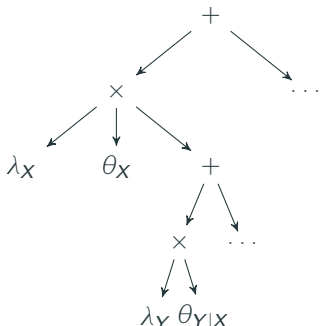
Kaplan and Ivanovska: Efficient belief propagation in second-order Bayesian networks for singly-connected graphs. Int. J. Approx. Reason. 93: 132-152 (2018)

- It tackles second-order correlation by looking at the covariances
- Leaves are still assumed to be independent. If that is not the case, leaves should become Dirichlet distributions (see for instance Lance M. Kaplan, Magdalena Ivanovska: Efficient belief propagation in second-order Bayesian networks for singly-connected graphs. Int. J. Approx. Reason. 93: 132-152 (2018))

What about the dependencies in learning?

Evidential Parameter Learning

$\theta_X \to p(X = 1)$, $\theta_{Y|X} \to p(Y = 1 \mid X = 1)$, and $\theta_{Y|\overline{X}} \to p(Y = 1 \mid X = 0)$

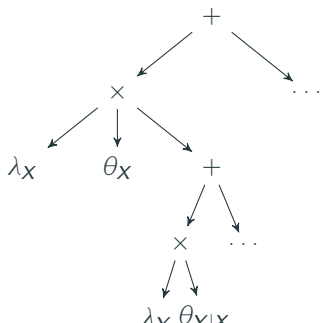


X	Y	Likelihood
0	1	$\theta_{Y\|\overline{X}}\ (1-\theta_X)$
0	0	$(1-\theta_{Y\|\overline{X}})\ (1-\theta_X)$
1	1	$\theta_{Y\|X}\ \theta_X$
0	0	$(1-\theta_{Y\|\overline{X}})\ (1-\theta_X)$
1	1	$\theta_{Y\|X}\ \theta_X$
1	0	$(1-\theta_{Y\|X})\ \theta_X$
1	1	$\theta_{Y\|X}\ \theta_X$
⋮	⋮	⋮

MAP from complete data leads to simple decomposition

$$p(\theta_X, \theta_{Y|X}, \theta_{Y|\overline{X}}) \propto \theta_X^{n_{YX}+n_{\overline{Y}X}}\ (1-\theta_X)^{n_{Y\overline{X}}+n_{\overline{Y}\overline{X}}}\ \theta_{Y|X}^{n_{YX}}\ (1-\theta_{Y|X})^{n_{\overline{Y}X}}\ \theta_{Y|\overline{X}}^{n_{Y\overline{X}}}\ (1-\theta_{Y|\overline{X}})^{n_{\overline{Y}\overline{X}}}$$

$\theta_X \to p(X = 1)$, $\theta_{Y|X} \to p(Y = 1 \mid X = 1)$, and $\theta_{Y|\overline{X}} \to p(Y = 1 \mid X = 0)$



X	Y	Likelihood
0	1	$\theta_{Y\|\overline{X}}\ (1-\theta_X)$
0	?	$(1-\theta_X)$
?	1	$\theta_{Y\|X}\ \theta_X + \theta_{Y\|\overline{X}}\ (1-\theta_X)$
?	0	$1-(\theta_{Y\|X}\ \theta_X + \theta_{Y\|\overline{X}}\ (1-\theta_X))$
1	1	$\theta_{Y\|X}\ \theta_X$
1	0	$(1-\theta_{Y\|X})\ \theta_X$
1	?	θ_X
⋮	⋮	⋮

MAP from complete data does not lead to simple decomposition

$$p(\theta_X, \theta_{Y|X}, \theta_{Y|\overline{X}}) \propto \theta_X^{n_X + n_{YX} + n_{\overline{Y}X}}\ (1-\theta_X)^{n_{\overline{X}} + n_{Y\overline{X}} + n_{\overline{Y}\overline{X}}}\ \theta_{Y|X}^{n_{YX}}\ (1-\theta_{Y|X})^{n_{\overline{Y}X}}\ \theta_{Y|\overline{X}}^{n_{Y\overline{X}}}\ (1-\theta_{Y|\overline{X}})^{n_{\overline{Y}\overline{X}}} \cdot \cdot\ (\theta_{Y|X}\ \theta_X + \theta_{Y|\overline{X}}\ (1-\theta_X))^{n_Y}\ (1-(\theta_{Y|X}\ \theta_X + \theta_{Y|\overline{X}}\ (1-\theta_X)))^{n_{\overline{Y}}}$$

Learning with incomplete information:

- Bayesian Moment Matching: A. Rashwan, H. Zhao, and P. Poupart, "Online and distributed Bayesian moment matching for parameter learning in sum-product networks," AISTAT 2016.

- EM + Gaussian approximation: Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. Bayesian data analysis. Chapman and Hall/CRC, 1995.

- EM + Fisher Information: Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. Bayesian data analysis. Chapman and Hall/CRC, 1995.

Bayesian Moment Matching

After t instantiation, let us presume the posterior is a product of Dirichlets, i.e.,

$f^{(t)}(\theta) \approx f(\theta|\{e_{t'}\}_{t'=1}^{t})$ where $f^{(t)}(\theta) = \prod_{i=1}^{n} \prod_{pa_i \in \mathbb{PA}_i} Dir(\theta_{X_i|pa_i}; \alpha^{(t)}_{X_i|pa_i})$

and fitting a product of Dirichlets to the posterior after $t+1$ instantiations, i.e.,

$$f^{(t+1)}(\theta) = \underbrace{\left(\sum_{x_i \sim e_{t+1}} \sum_{pa_i \sim e_{t+1}} \theta_{x_i|pa_i} \frac{\partial p(e;\theta)}{\partial \theta_{x_i|pa_i}} \right)}_{p(e_{t+1};\theta)} f^{(t)}(\theta), \text{ via the method of moments.}$$

The first and second order moments of the parameters are $m^{(t)}_{x_i|pa_i} = E[\theta_{x_i|pa_i}] = \frac{Z[1]}{Z[0]}$, $v^{(t)}_{x_i|pa_i} = E[\theta^2_{x_i|pa_i}] = \frac{Z[2]}{Z[0]}$, where $Z[k;\theta_{x_i|pa_i}] = \int \theta^k_{x_i|pa_i} p(e_{t+1};\theta) f^{(t)}(\theta) d\theta$ and can be computed in closed form by leveraging the properties of Dirichlet distributions.

*A. Rashwan *et. al.*, "Online and distributed Bayesian moment matching for parameter learning in sum-product networks," AISTAT 2016

EM for estimating means

Estimate the parameters as the maximum a posteriori (MAP) estimate
$\theta = \text{argmax}_\theta \log((P(X|\theta)f(\theta))$

In the case of incomplete data, the logs do not break into chains of simple additions, hence the need for a 2-step EM algorithm.

Step 1: Expectation. $Q(\theta; \theta^{(t)}) = \sum_{X_\ell \in \mathbb{X}_\ell} \log(P(X_o, X_\ell; \theta)f(\theta))P(X_\ell|X_o; \theta^{(t)})$

where X_ℓ are the unobserved latent variables and X_o are the observed variables.

Step 2: Maximisation step, which updates the estimated parameters.
$\theta^{(t+1)} = \text{argmax}_\theta Q(\theta; \theta^{(t)})$.

*Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. Bayesian data analysis. Chapman and Hall/CRC, 1995.

Gaussian Approximation

Approximate the covariance matrix as

$$R = D^T(DHD^T)^{-1}D,$$

where

$$H \approx J_0 + \sum_t \frac{1}{p^2(e_t)} \nabla_\theta p(e_t) \nabla_\theta^T p(e_t).$$

Fisher Information

Approximate the covariance matrix as

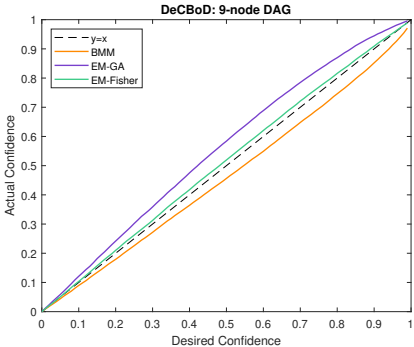
$$R = D^T(DJD^T)^{-1}D.$$

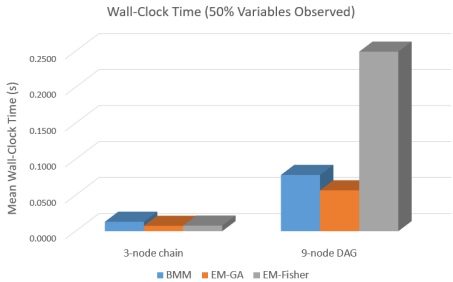
where

$$J = J_0 + \sum_t \sum_{e' \in \mathbb{E}_t} \frac{1}{p^2(e')} \nabla_\theta p(e') \nabla_\theta^T p(e')$$

is the Fisher Information Matrix.

*Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. Bayesian data analysis. Chapman and Hall/CRC, 1995.



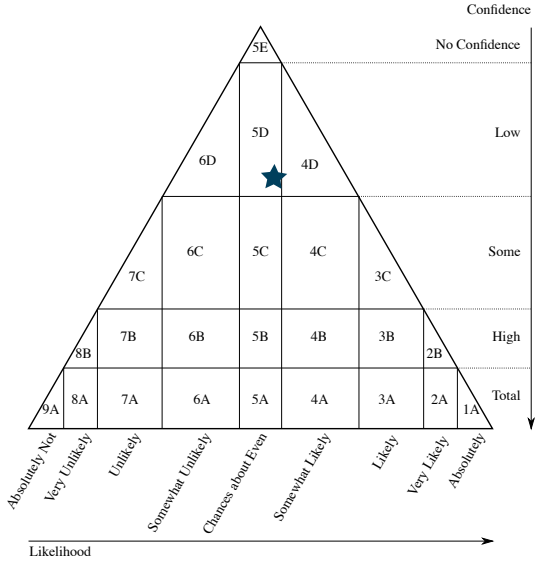


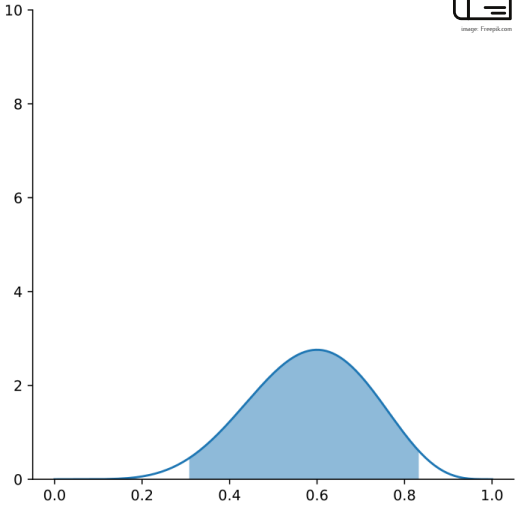
*Conrad D. Hougen, Lance M. Kaplan, Federico Cerutti, Alfred O. Hero III: Uncertain Bayesian Networks: Learning from Incomplete Data. MLSP 2021: 1-6

Ascertain Evidence from the Real World

> *"…cumulative net CO2 emissions over the last decade (2010–2019) are about the same size as the 11 remaining carbon budget likely to limit warming to 1.5C (medium confidence)."*

*PCC. Climate Change 2022: Impacts, Adaptation, and Vulnerability. Cambridge University Press; In press





*Josang, Audun. Subjective Logic: A Formalism for Reasoning under Uncertainty. Springer, 2016.

Uncertainty-Awareness

Change the loss function so to output pieces of evidences in favour of different classes that should then be considered through Bayesian update resulting into a Dirichlet Distribution

*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. "Evidential deep learning to quantify classification uncertainty." Advances in Neural Information Processing Systems. 2018.

From Evidence to Dirichlet

Let us now assume a Dirichlet distribution over K classes that is the result of Bayesian update with N observations and starting with a uniform prior:

$$\mathrm{Dir}(\boldsymbol{\mu} \mid \boldsymbol{\alpha}) = \mathrm{Dir}(\boldsymbol{\mu} \mid \langle e_1 + 1, e_2 + 2, \ldots, e_K + 1 \rangle)$$

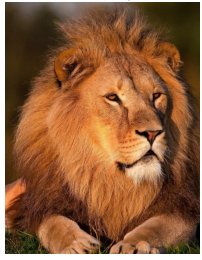
where e_i is the number of observations (evidence) for the class k, and $\sum_k e_k = N$.

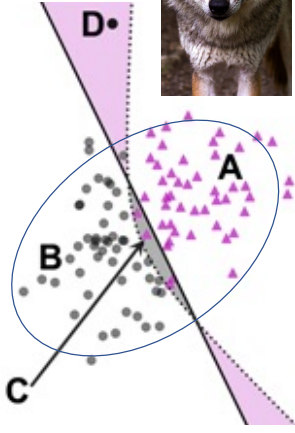
Intuition.

Pieces of evidence for the various classes should be representative of the training samples *nearby* (geodesic space, euclidian space, ...) a test sample

D. High epistemic uncertainty

B. Low aleatoric and Low epistemic uncertainty





A. Low aleatoric and Low epistemic

C. High aleatoric and Low epistemic uncertainty

Dirichlet and Epistemic Uncertainty

The epistemic uncertainty associated to a Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\mu} \mid \boldsymbol{\alpha})$ can be estimated by

$$u = \frac{K}{S}$$

with K the number of classes and $S = \alpha_0 = \sum_{k=1}^{K} \alpha_k$ is the Dirichlet strength.

Note that if the Dirichlet has been computed as the resulting of Bayesian update from a uniform prior, $0 \leq u \leq 1$, and $u = 1$ implies that we are considering the uniform distribution (an extreme case of Dirichlet distribution).

Let us denote with $\widehat{\mu}_k \triangleq \frac{\alpha_k}{S}$.

Loss function: two components

Multiple loss functions introduced, each with two components:

- one aims at minimising the prediction error;
- the other the number of pieces of evidence generated for each class, thus learning to say *I do not known* when facing ambiguous datapoints.

*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. "Evidential deep learning to quantify classification uncertainty." Advances in Neural Information Processing Systems. 2018.

Loss function: minimising the prediction error

If we then consider $\text{Dir}(\boldsymbol{\mu}_i \mid \boldsymbol{\alpha}_i)$ as the prior for a Multinomial $\boldsymbol{p}(\boldsymbol{y}_i \mid \boldsymbol{\mu}_i)$, we can then compute the expected squared error (aka Brier score)

$$
\begin{aligned}
\mathbb{E}[\left\|\boldsymbol{y}_i - \boldsymbol{\mu}_i\right\|_2^2] &= \sum_{k=1}^{K} \mathbb{E}[y_{i,k}^2 - 2y_{i,k}\mu_{i,k} + \mu_{i,k}^2] = \sum_{k=1}^{X} y_{i,k}^2 - 2y_{i,k}\mathbb{E}[\mu_{i,k}] + \mathbb{E}[\mu_{i,k}^2] = \\
&= \sum_{k=1}^{K} y_{i,k}^2 - 2y_{i,k}\mathbb{E}[\mu_{i,k}] + \mathbb{E}[\mu_{i,k}]^2 + \text{var}[\mu_{i,k}] = \\
&= \sum_{k=1}^{K} (y_{i,k} - \mathbb{E}[\mu_{i,k}])^2 + \text{var}[\mu_{i,k}] = \\
&= \sum_{k=1}^{K} \left(y_{i,k} - \frac{\alpha_{i,k}}{S_i} \right)^2 + \frac{\alpha_{i,k}(S_i - \alpha_{i,k})}{S_i^2(S_i + 1)} = \\
&= \sum_{k=1}^{K} (y_{i,k} - \widehat{\mu_{i,k}})^2 + \frac{\widehat{\mu_{i,k}}(1 - \widehat{\mu_{i,k}})^2}{S_i + 1}
\end{aligned}
$$

The authors provide other loss functions to minimise the prediction error.

*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. "Evidential deep learning to quantify classification uncertainty." Advances in Neural Information Processing Systems. 2018.

Learning to say “I don’t know”

To avoid generating evidence for all the classes when the network cannot classify a given sample (epistemic uncertainty), we introduce a term in the loss function that penalises the divergence from the uniform distribution:

$$\mathcal{L} \quad = \quad \sum_{i=1}^{N} \mathcal{L}_i(\theta) + \lambda_t \sum_{i=1}^{N} \mathrm{KL}\left(\mathrm{Dir}(\boldsymbol{\mu}_i \mid \widetilde{\boldsymbol{\alpha}}_i) \;||\; \mathrm{Dir}(\boldsymbol{\mu}_i \mid 1) \right)$$

where:

- λ_t is another hyperparameter, and the suggestion is to use it parametric on the number of training epochs, e.g. $\lambda_t = \min\left(1, \frac{t}{\mathrm{CONST}}\right)$ with t the number of current training epoch, so that the effect of the KL divergence is gradually increased to avoid premature convergence to the uniform distribution in the early epoch where the learning algorithm still needs to explore the parameter space;
- $\widetilde{\boldsymbol{\alpha}}_i = \boldsymbol{y}_i + (1 - \boldsymbol{y}_i) \cdot \alpha_i$ are the Dirichlet parameters the neural network in a forward pass has put on the wrong classes, and the idea is to minimise them as much as possible.

*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. “Evidential deep learning to quantify classification uncertainty.” Advances in Neural Information Processing Systems. 2018.

KL recap

Consider some unknown distribution $p(\boldsymbol{x})$ and suppose that we have modelled this using $q(\boldsymbol{x})$. If we use $q(\boldsymbol{x})$ instead of $p(\boldsymbol{x})$ to represent the true values of $\boldsymbol{x}$, the average *additional* amount of information required is:

$$\begin{aligned} \mathrm{KL}(p||q) &= -\int p(\boldsymbol{x}) \ln q(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \left(-\int p(\boldsymbol{x}) \ln p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right) \\ &= -\int p(\boldsymbol{x}) \ln \left\{ \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} \right\} \mathrm{d}\boldsymbol{x} \\ &= -\mathbb{E}\left[\ln \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} \right] \end{aligned} \tag{2}$$

This is known as the *relative entropy* or *Kullback-Leibler divergence*, or *KL divergence* between the distributions $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$.

Properties:

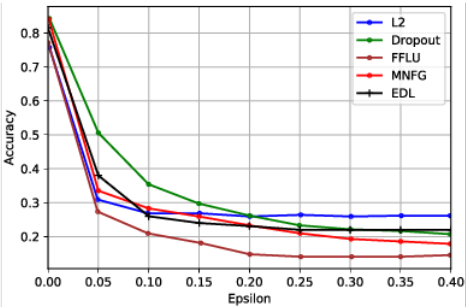
- $\mathrm{KL}(p||q) \not\equiv \mathrm{KL}(q||p)$;
- $\mathrm{KL}(p||q) \geq 0$ and $\mathrm{KL}(p||q) = 0$ if and only if $p = q$

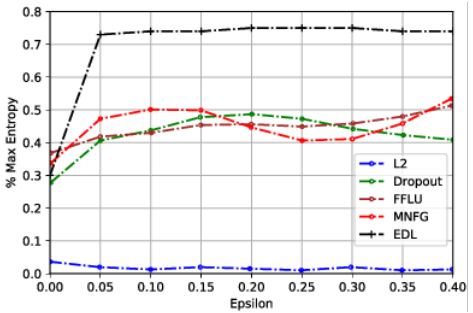
$$\mathrm{KL}\left(\ \mathrm{Dir}(\boldsymbol{\mu}_i \mid \widetilde{\boldsymbol{\alpha}}_i) \ \|\ \mathrm{Dir}(\boldsymbol{\mu}_i \mid \mathbf{1})\ \right) = \ln\left(\frac{\Gamma(\sum_{k=1}^{K}\widetilde{\alpha}_{i,k})}{\Gamma(K)\prod_{k=1}^{K}\Gamma(\widetilde{\alpha}_{i,k})}\right)+\sum_{k=1}^{K}(\widetilde{\alpha}_{i,k}-1)\left[\psi(\widetilde{\alpha}_{i,k})-\psi\left(\sum_{j=1}^{K}\widetilde{\alpha}_{i,j}\right)\right]$$

where $\psi(x) = \dfrac{\mathrm{d}}{\mathrm{d}x}\ln\left(\ \Gamma(x)\ \right)$ is the *digamma* function

*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. "Evidential deep learning to quantify classification uncertainty." Advances in Neural Information Processing Systems. 2018.

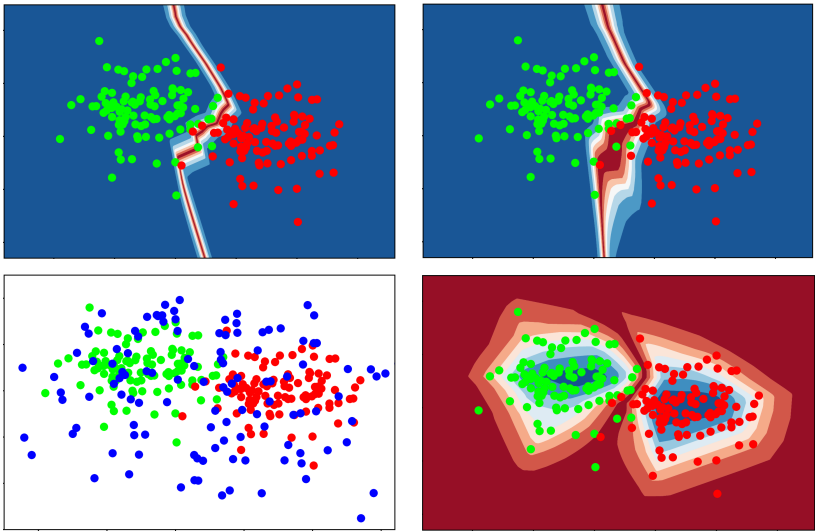
EDL and robustness to FGS



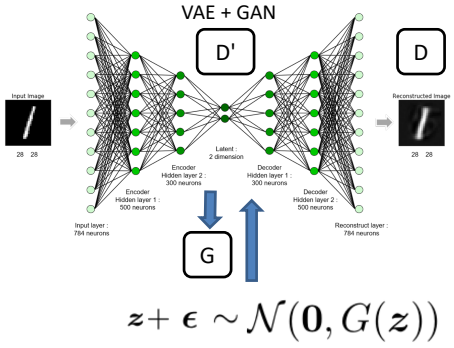


*Sensoy, Murat, Lance Kaplan, and Melih Kandemir. "Evidential deep learning to quantify classification uncertainty." Advances in Neural Information Processing Systems. 2018.

EDL + GAN for adversarial training



*Sensoy, Murat, et al. "Uncertainty-Aware Deep Classifiers using Generative Models." AAAI 2020



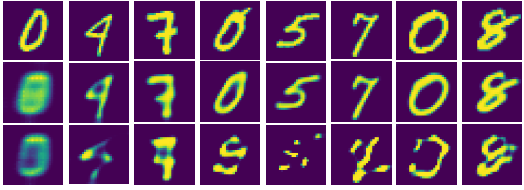


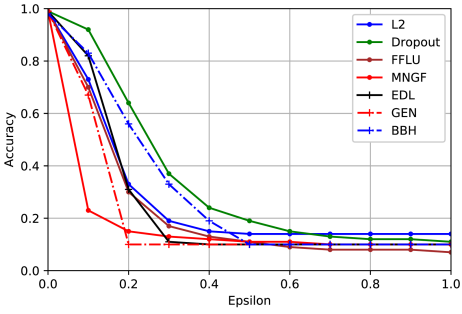
image: Freepik.com

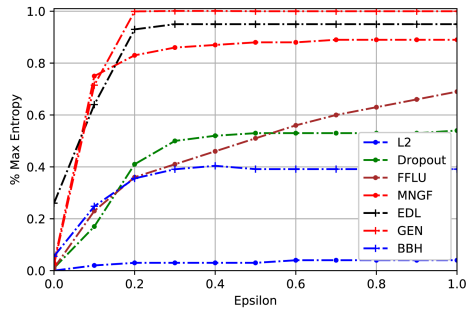
For each $\boldsymbol{x}_i$ in training set, we sample a latent point $\boldsymbol{z}$ from $q_\theta(\boldsymbol{z} \mid \boldsymbol{x}_i)$ and perturb it by $\boldsymbol{\epsilon} \sim \mathcal{N}(0, G(\boldsymbol{z}))$, where $G(\cdot)$ is a GAN with two discriminators D and D'.

The generated points are forced to be similar to the real latent points through making them indistinguishable by D' in the latent space of the VAE, while having the generated samples to be distinguishable by D in the input space.

*Sensoy, Murat, et al. "Uncertainty-Aware Deep Classifiers using Generative Models." AAAI 2020

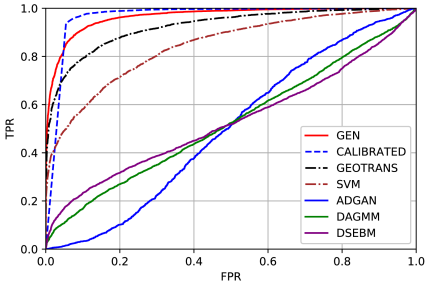
Robustness against FGS



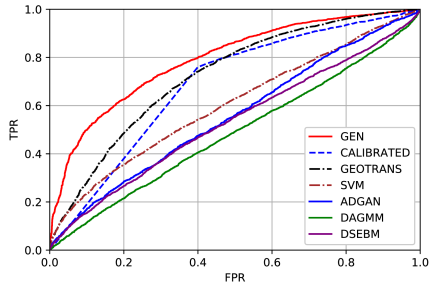


*Sensoy, Murat, et al. "Uncertainty-Aware Deep Classifiers using Generative Models." AAAI 2020

Anomaly detection



(mnist)



(cifar10)

*Sensoy, Murat, et al. "Uncertainty-Aware Deep Classifiers using Generative Models." AAAI 2020

Many other approaches

Prior networks: auxiliary dataset for out-of-distributions.
A. Malinin and M. Gales. Reverse KL-Divergence Training of Prior Networks: Improved Uncertainty and Adversarial Robustness. In NeurIPS, 2019.

Posterior networks: using normalising flow for learning a latent representation of the input.
B. Charpentier, D. Zügner, and S. Günnemann. Posterior Network: Uncertainty Estimation without OOD Samples via Density-Based Pseudo-Counts. In NeurIPS, pages 1356–1367, 2020.

Tutorials/Reviews
M. Abdar, *et. al.* A review of uncertainty quantification in deep learning: Techniques, applications and challenges. Information Fusion, 76:243–297, 2021.
E. Hüllermeier and W. Waegeman. Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods. Mach. Learn., 110(3):457–506, 2021.

Summary and Conclusions

- Effective approximations for quantifying aleatory and epistemic uncertainty in reasoning and learning
- Evidential reasoning introduces the idea of choosing either beta or Dirichlet distributions to represent uncertain probabilities and then using efficient methods—such as the the moment matching—for manipulating them
- Several research questions are left unanswered
  - Efficient algorithms, in particular when it comes to parameter (and structure) learning in probabilistic circuits
  - When dealing with real-world problems, how to deal with an input which is classified with high epistemic uncertainty: does it identify a new class?
  - Evidential learning and reasoning in neuro-symbolic/neuro-programming architectures

Announcements

Survey paper in the main conference

Federico Cerutti, Lance Kaplan, Murat Sensoy. Evidential Reasoning and Learning: a Survey.

Scheduled on July 28th at 1000h in Lehar 1 – (12 min talk)
Poster session 2 at stand 318 row 9

University of Brescia, Italy, will open soon a 3-years RA/post-doc position on evidential reasoning and learning.