Pearl's & Jeffrey's update rules in probabilistic learning

Radboud University Nijmegen
Topos Institute, online, April 17, 2025

Bart Jacobs
bart@cs.ru.nl

Outline

Where we are, so far

Challenges in probabilistic logic (from Pearl'89)

Challenges in probabilistic logic (from Pearl'89)

To those trained in traditional logics, symbolic reasoning is the standard, and nonmonotonicity a novelty. To students of probability, on the other hand, it is symbolic reasoning that is novel, not nonmonotonicity. Dealing with new facts that cause probabilities to change abruptly from very high values to very low values is a commonplace phenomenon in almost every probabilistic exercise and, naturally, has attracted special attention among probabilists. The new challenge for probabilists is to find ways of abstracting out the numerical character of high and low probabilities, and cast them in linguistic terms that reflect the natural process of accepting and retracting beliefs.

Challenges in probabilistic logic (from Pearl'89)

> *To those trained in traditional logics, symbolic reasoning is the standard, and nonmonotonicity a novelty. To students of probability, on the other hand, it is symbolic reasoning that is novel, not nonmonotonicity. Dealing with new facts that cause probabilities to change abruptly from very high values to very low values is a commonplace phenomenon in almost every probabilistic exercise and, naturally, has attracted special attention among probabilists. The new challenge for probabilists is to find ways of abstracting out the numerical character of high and low probabilities, and cast them in linguistic terms that reflect the natural process of accepting and retracting beliefs.*

Embarrassingly, there is still **no** probabilistic logic for symbolic reasoning.

Probabilistic reasoning and updating (belief revision)

Probabilistic reasoning and updating (belief revision)

Example

I may think that scientists are civilised people. But then I attend a conference dinner that ends in a fist fight.

I will update my judgement.

Probabilistic reasoning and updating (belief revision)

Example

I may think that scientists are civilised people. But then I attend a conference dinner that ends in a fist fight.

I will update my judgement.

- This is difficult in traditional, monotonic logic, where adding more information can not make true statements false.

Probabilistic reasoning and updating (belief revision)

Example

I may think that scientists are civilised people. But then I attend a conference dinner that ends in a fist fight.

I will update my judgement.

- This is difficult in traditional, monotonic logic, where adding more information can not make true statements false.
- We need to switch from truth/falsity of statement, to likelihood
  - not two-element set $\{0, 1\}$ but interval $[0, 1]$
  - not sharp but fuzzy (soft) statements

Probabilistic reasoning and updating (belief revision)

Example

I may think that scientists are civilised people. But then I attend a conference dinner that ends in a fist fight.

I will update my judgement.

- This is difficult in traditional, monotonic logic, where adding more information can not make true statements false.
- We need to switch from truth/falsity of statement, to likelihood
  - not two-element set $\{0, 1\}$ but interval $[0, 1]$
  - not sharp but fuzzy (soft) statements

  The likelihood that scientists are civilised is decreased, by the events at the conference dinner, through updating (belief revision).

Naive picture of learning

Naive picture of learning

Naive picture of learning

"Nürnberger Trichter"
(Nurnberg Funnel)

Alternative: predictive coding theory (Karl Friston et al)

Alternative: predictive coding theory (Karl Friston et al)

- The human mind is constantly active in making predictions
- These predictions are compared with what actually happens
- Mismatches (prediction errors) lead to updates in the brain

Alternative: predictive coding theory (Karl Friston et al)

- The human mind is constantly active in making predictions
- These predictions are compared with what actually happens
- Mismatches (prediction errors) lead to updates in the brain

> "The human brain is a Bayesian prediction & correction engine"

Alternative: predictive coding theory (Karl Friston et al)

- The human mind is constantly active in making predictions
- These predictions are compared with what actually happens
- Mismatches (prediction errors) lead to updates in the brain

> "The human brain is a Bayesian prediction & correction engine"

Possibly it is better to call the mind a Jeffreyan engine ...

My own (logical) interests/work

My own (logical) interests/work

- There are two update rules, by Judea Pearl (1936) and by Richard Jeffrey (1926-2002), which are not well-distinguished in the literature
  - They both have clear formulations using channels — see later
  - What are the differences? When to use which rule? Unclear!

- BJ, *The Mathematics of Changing one's Mind, via Jeffrey's or via Pearl's update rule*, Journ. of AI Research, 2019
- BJ, *Learning from What's Right and Learning from What's Wrong*, MFPS'21
- BJ & Dario Stein, *Pearl's and Jeffrey's Update as Modes of Learning in Probabilistic Programming*, MFPS'23
- BJ, *Getting Wiser from Multiple Data: Probabilistic Updating according to Jeffrey and Pearl*, 2024, 10.48550/arXiv.2405.12700

My own (logical) interests/work

- There are two update rules, by Judea Pearl (1936) and by Richard Jeffrey (1926-2002), which are not well-distinguished in the literature
  - They both have clear formulations using channels — see later
  - What are the differences? When to use which rule? Unclear!
- The topic is mathematically non-trivial
  - esp. in Jeffrey's case, as we shall see

- BJ, *The Mathematics of Changing one's Mind, via Jeffrey's or via Pearl's update rule*, Journ. of AI Research, 2019
- BJ, *Learning from What's Right and Learning from What's Wrong*, MFPS'21
- BJ & Dario Stein, *Pearl's and Jeffrey's Update as Modes of Learning in Probabilistic Programming*, MFPS'23
- BJ, *Getting Wiser from Multiple Data: Probabilistic Updating according to Jeffrey and Pearl*, 2024, 10.48550/arXiv.2405.12700

My own (logical) interests/work

- There are two update rules, by Judea Pearl (1936) and by Richard Jeffrey (1926-2002), which are not well-distinguished in the literature
  - They both have clear formulations using channels — see later
  - What are the differences? When to use which rule? Unclear!
- The topic is mathematically non-trivial
  - esp. in Jeffrey's case, as we shall see
- Intriguing question: does the human mind use Pearl's or Jeffrey's rule — within predictive coding theory
  - cognitive science may provide an answer

- BJ, *The Mathematics of Changing one's Mind, via Jeffrey's or via Pearl's update rule*, Journ. of AI Research, 2019
- BJ, *Learning from What's Right and Learning from What's Wrong*, MFPS'21
- BJ & Dario Stein, *Pearl's and Jeffrey's Update as Modes of Learning in Probabilistic Programming*, MFPS'23
- BJ, *Getting Wiser from Multiple Data: Probabilistic Updating according to Jeffrey and Pearl*, 2024, 10.48550/arXiv.2405.12700

Example I, medical test, part I

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%
- There is a test for the disease with:
  - ('sensitivity') If someone has the disease, then the test is positive with probability of 90%
  - ('specificity') If someone does not have the disease, there is a 95% chance that the test is negative.

Example I, medical test, part I

- Consider a disease with *a priori* probability (or ‘prevalence’) of 10%
- There is a test for the disease with:
  - (‘sensitivity’) If someone has the disease, then the test is positive with probability of 90%
  - (‘specificity’) If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%
- There is a test for the disease with:
  - ('sensitivity') If someone has the disease, then the test is positive with probability of 90%
  - ('specificity') If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%
- The test is performed, under unfavourable circumstances like bad light, and we are only 80% sure that the test is positive. What is the disease likelihood?

Example I, medical test, part I

- Consider a disease with *a priori* probability (or ‘prevalence’) of 10%
- There is a test for the disease with:
  - (‘sensitivity’) If someone has the disease, then the test is positive with probability of 90%
  - (‘specificity’) If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%
- The test is performed, under unfavourable circumstances like bad light, and we are only 80% sure that the test is positive. What is the disease likelihood?
- Updating with { **Pearl**'s rule gives: / **Jeffrey**'s rule gives:

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%
- There is a test for the disease with:
  - ('sensitivity') If someone has the disease, then the test is positive with probability of 90%
  - ('specificity') If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%
- The test is performed, under unfavourable circumstances like bad light, and we are only 80% sure that the test is positive. What is the disease likelihood?
- Updating with
  - **Pearl**'s rule gives: 26% disease likelihood
  - **Jeffrey**'s rule gives:

iCIS | Digital Security
Radboud University

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%
- There is a test for the disease with:
  - ('sensitivity') If someone has the disease, then the test is positive with probability of 90%
  - ('specificity') If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%
- The test is performed, under unfavourable circumstances like bad light, and we are only 80% sure that the test is positive. What is the disease likelihood?
- Updating with
  - **Pearl**'s rule gives: 26% disease likelihood
  - **Jeffrey**'s rule gives: 54%

Example I, medical test, part I

- Consider a disease with *a priori* probability (or 'prevalence') of 10%
- There is a test for the disease with:
  - ('sensitivity') If someone has the disease, then the test is positive with probability of 90%
  - ('specificity') If someone does not have the disease, there is a 95% chance that the test is negative.
- Computing the predicted positive test probability yields: 13.5%
- The test is performed, under unfavourable circumstances like bad light, and we are only 80% sure that the test is positive. What is the disease likelihood?
- Updating with
  - **Pearl**'s rule gives: 26% disease likelihood
  - **Jeffrey**'s rule gives: 54%
- Jeffrey is more than twice as high as Pearl. Which should a doctor use?

Example II: multiple test results

Example II: multiple test results

In the same test set-up as before, you test three times and get:

two positive tests and one negative test

What is the posterior disease probability?

iCIS | Digital Security
Radboud University

Example II: multiple test results

In the same test set-up as before, you test three times and get:

two positive tests and one negative test

What is the posterior disease probability?

Updating with { **Pearl**'s rule gives:
Jeffrey's rule gives:

Example II: multiple test results

In the same test set-up as before, you test three times and get:

two positive tests and one negative test

What is the posterior disease probability?

Updating with
- **Pearl**'s rule gives: 79% disease likelihood
- **Jeffrey**'s rule gives:

Example II: multiple test results

In the same test set-up as before, you test three times and get:

two positive tests and one negative test

What is the posterior disease probability?

Updating with
- **Pearl**'s rule gives: 79% disease likelihood
- **Jeffrey**'s rule gives: 49%

Pearl & Jeffrey updating as optimisations

Pearl & Jeffrey updating as optimisations

(What is formulated informally at this stage, will be made mathematatcially precise later)

Pearl & Jeffrey updating as optimisations

(What is formulated informally at this stage, will be made mathematcially precise later)

(1) **Pearl's rule:**

(2) **Jeffrey's rule:**

Pearl & Jeffrey updating as optimisations

(What is formulated informally at this stage, will be made mathematatcially precise later)

(1) **Pearl's rule:**
- uses evidence (predicate) to update a *prior* to a *posterior*
- such that the validity (expected value) of the evidence increases
- formally: the validity of the evidence in the prediction based on the posterior is higher than in the predication based on the prior

(2) **Jeffrey's rule:**

iCIS | Digital Security
Radboud University

Pearl & Jeffrey updating as optimisations

(What is formulated informally at this stage, will be made mathematcially precise later)

(1) **Pearl's rule:**
- uses evidence (predicate) to update a *prior* to a *posterior*
- such that the validity (expected value) of the evidence increases
- formally: the validity of the evidence in the prediction based on the posterior is higher than in the predication based on the prior

(2) **Jeffrey's rule:**
- uses an observed distribution/state to update from *prior* to *posterior*
- such that the mismatch with the observation decreases
- formally: the KL-divergence between the observation and the prediction based on the posterior is lower than on the prior

Pearl & Jeffrey updating as optimisations

(What is formulated informally at this stage, will be made mathematcially precise later)

(1) **Pearl's rule:**
- uses evidence (predicate) to update a *prior* to a *posterior*
- such that the validity (expected value) of the evidence increases
- formally: the validity of the evidence in the prediction based on the posterior is higher than in the predication based on the prior

(2) **Jeffrey's rule:**
- uses an observed distribution/state to update from *prior* to *posterior*
- such that the mismatch with the observation decreases
- formally: the KL-divergence between the observation and the prediction based on the posterior is lower than on the prior

Thus, Jeffrey's rule reduces prediction errors, as in predictive coding

Where we are, so far

Comparison table about updating (with informal descriptions)

Comparison table about updating (with informal descriptions)

	Pearl's rule	Jeffrey's rule
effect		
you learn nothing from		
successive updates commute?		

Comparison table about updating (with informal descriptions)

	Pearl's rule	Jeffrey's rule
effect	increase of what's right	
you learn nothing from		
successive updates commute?		

Comparison table about updating (with informal descriptions)

	Pearl's rule	**Jeffrey's rule**
effect	increase of what's right	decrease of what's wrong
you learn nothing from		
successive updates commute?		

Comparison table about updating (with informal descriptions)

	Pearl's rule	**Jeffrey's rule**
effect	increase of what's right	decrease of what's wrong
you learn nothing from	uniformity (no differences)	
successive updates commute?		

Comparison table about updating (with informal descriptions)

	Pearl's rule	**Jeffrey's rule**
effect	increase of what's right	decrease of what's wrong
you learn nothing from	uniformity (no differences)	what you already know (can predict)
successive updates commute?		

Comparison table about updating (with informal descriptions)

	Pearl's rule	Jeffrey's rule
effect	increase of what's right	decrease of what's wrong
you learn nothing from	uniformity (no differences)	what you already know (can predict)
successive updates commute?	yes	

Comparison table about updating (with informal descriptions)

	Pearl's rule	**Jeffrey's rule**
effect	increase of what's right	decrease of what's wrong
you learn nothing from	uniformity (no differences)	what you already know (can predict)
successive updates commute?	yes	no

iCIS | Digital Security
Radboud University

Big question

Big question

- Does the human mind use Pearl's or Jeffrey's rule?

iCIS | Digital Security
Radboud University

Big question

- Does the human mind use Pearl's or Jeffrey's rule?
- My bet is on Jeffrey . . .

Big question

- Does the human mind use Pearl's or Jeffrey's rule?
- My bet is on Jeffrey . . .
- Since the human mind is very sensitive to the order of updating (priming)

Big question

- Does the human mind use Pearl's or Jeffrey's rule?
- My bet is on Jeffrey . . .
- Since the human mind is very sensitive to the order of updating (priming)

My favourite example: consider the impact of the following two sentences, in different orders.

Big question

- Does the human mind use Pearl's or Jeffrey's rule?
- My bet is on Jeffrey . . .
- Since the human mind is very sensitive to the order of updating (priming)

My favourite example: consider the impact of the following two sentences, in different orders.

Alice is sick Bob visits Alice.

Big question

- Does the human mind use Pearl's or Jeffrey's rule?
- My bet is on Jeffrey . . .
- Since the human mind is very sensitive to the order of updating (priming)

My favourite example: consider the impact of the following two sentences, in different orders.

Alice is sick　　　　Bob visits Alice.

Versus:

Bob visits Alice　　　　Alice is sick.

Where we are, so far

Distributions (finite, discrete)

Distributions (finite, discrete)

A distribution (or state) over a set X is a formal finite convex sum:

$$\sum_i r_i |\, x_i \,\rangle \;\in\; \mathcal{D}(X) \qquad \text{where} \qquad \begin{cases} r_i \in [0,1], \text{ with } \sum_i r_i = 1 \\ x_i \in X \end{cases}$$

Distributions (finite, discrete)

A distribution (or state) over a set X is a formal finite convex sum:

$$\sum_i r_i | x_i \rangle \;\in\; \mathcal{D}(X) \qquad \text{where} \qquad \begin{cases} r_i \in [0,1], \text{ with } \sum_i r_i = 1 \\ x_i \in X \end{cases}$$

- Distributions can also be described as functions $\sigma\colon X \to [0,1]$ with finite support and $\sum_x \sigma(x) = 1$

Distributions (finite, discrete)

A distribution (or state) over a set X is a formal finite convex sum:

$$\sum_i r_i | x_i \rangle \in \mathcal{D}(X) \qquad \text{where} \qquad \begin{cases} r_i \in [0,1], \text{ with } \sum_i r_i = 1 \\ x_i \in X \end{cases}$$

- Distributions can also be described as functions $\sigma \colon X \to [0,1]$ with finite support and $\sum_x \sigma(x) = 1$
- This $\mathcal{D}$ is the distribution monad on Sets

Distributions (finite, discrete)

A distribution (or state) over a set X is a formal finite convex sum:

$$\sum_i r_i| x_i \rangle \;\in\; \mathcal{D}(X) \qquad \text{where} \qquad \begin{cases} r_i \in [0,1], \text{ with } \sum_i r_i = 1 \\ x_i \in X \end{cases}$$

- Distributions can also be described as functions $\sigma\colon X \to [0,1]$ with finite support and $\sum_x \sigma(x) = 1$
- This $\mathcal{D}$ is the distribution monad on Sets
- A Kleisli map $X \to \mathcal{D}(Y)$ is also called a channel, and written as $X \rightsquigarrow Y$, with special arrow. Channels capture conditional probabilities $p(Y|X)$ in a graphical calculus, via string diagrams

Distributions (finite, discrete)

A distribution (or state) over a set X is a formal finite convex sum:

$$\sum_i r_i |\, x_i \,\rangle \;\in\; \mathcal{D}(X) \qquad \text{where} \qquad \begin{cases} r_i \in [0,1], \text{ with } \sum_i r_i = 1 \\ x_i \in X \end{cases}$$

- Distributions can also be described as functions $\sigma \colon X \to [0,1]$ with finite support and $\sum_x \sigma(x) = 1$
- This $\mathcal{D}$ is the distribution monad on Sets
- A Kleisli map $X \to \mathcal{D}(Y)$ is also called a channel, and written as $X \rightsquigarrow Y$, with special arrow. Channels capture conditional probabilities $p(Y|X)$ in a graphical calculus, via string diagrams
- For $\sigma \in \mathcal{D}(X)$ and $c \colon X \rightsquigarrow Y$ we have Kleisli extension / bind / state transformation / prediction: $c_*(\sigma) \in \mathcal{D}(Y)$. Explicitly, if $\sigma = \sum_i r_i |\, x_i \,\rangle$, prediction along channel c is:

$$c_*(\sigma) := \sum_i r_i \cdot c(x_i) = \sum_{y \in Y} \Big(\sum_i r_i \cdot c(x_i)(y) \Big) \,|\, y \,\rangle.$$

The disease-test example: state & channel

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:

$$prior = \tfrac{1}{10}|\, d\,\rangle + \tfrac{9}{10}|\, d^{\perp}\,\rangle.$$

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:
$$prior = \tfrac{1}{10}|\, d\,\rangle + \tfrac{9}{10}|\, d^{\perp}\,\rangle.$$
- Testing is done via the channel $test: D \to \mathcal{D}(T)$ with:

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:

$$\mathit{prior} = \tfrac{1}{10}|\, d\,\rangle + \tfrac{9}{10}|\, d^{\perp}\,\rangle.$$

- Testing is done via the channel $\mathit{test}\colon D \to \mathcal{D}(T)$ with:

$$\mathit{test}(d) = \tfrac{9}{10}|\, p\,\rangle + \tfrac{1}{10}|\, n\,\rangle \qquad \text{and } \mathit{test}(d^{\perp}) = \tfrac{1}{20}|\, p\,\rangle + \tfrac{19}{20}|\, n\,\rangle.$$

(Recall: sensitivity is $90\% = \frac{9}{10}$, specificity is $95\% = \frac{19}{20}$)

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:

$$prior = \frac{1}{10}|\,d\,\rangle + \frac{9}{10}|\,d^{\perp}\,\rangle.$$

- Testing is done via the channel $test \colon D \to \mathcal{D}(T)$ with:

$$test(d) = \frac{9}{10}|\,p\,\rangle + \frac{1}{10}|\,n\,\rangle \qquad \text{and } test(d^{\perp}) = \frac{1}{20}|\,p\,\rangle + \frac{19}{20}|\,n\,\rangle.$$

(Recall: sensitivity is $90\% = \frac{9}{10}$, specificity is $95\% = \frac{19}{20}$)

- The predicted test distribution is:

$$test_{*}(prior)$$

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:

$$prior = \tfrac{1}{10}|\, d\,\rangle + \tfrac{9}{10}|\, d^{\perp}\,\rangle.$$

- Testing is done via the channel $test\colon D \to \mathcal{D}(T)$ with:

$$test(d) = \tfrac{9}{10}|\, p\,\rangle + \tfrac{1}{10}|\, n\,\rangle \qquad \text{and } test(d^{\perp}) = \tfrac{1}{20}|\, p\,\rangle + \tfrac{19}{20}|\, n\,\rangle.$$

(Recall: sensitivity is $90\% = \frac{9}{10}$, specificity is $95\% = \frac{19}{20}$)

- The predicted test distribution is:

$$test_*(prior) = \tfrac{27}{200}|\, p\,\rangle + \tfrac{173}{200}|\, n\,\rangle = 0.135|\, p\,\rangle + 0.865|\, n\,\rangle.$$

iCIS | Digital Security
Radboud University

The disease-test example: state & channel

- Use sets $D = \{d, d^{\perp}\}$ for disease (or not) and $T = \{p, n\}$ for positive and negative test outcomes
- The prevalence state / distribution is:

$$prior = \tfrac{1}{10}|\, d\,\rangle + \tfrac{9}{10}|\, d^{\perp}\,\rangle.$$

- Testing is done via the channel $test\colon D \to \mathcal{D}(T)$ with:

$$test(d) = \tfrac{9}{10}|\, p\,\rangle + \tfrac{1}{10}|\, n\,\rangle \qquad \text{and } test(d^{\perp}) = \tfrac{1}{20}|\, p\,\rangle + \tfrac{19}{20}|\, n\,\rangle.$$

(Recall: sensitivity is $90\% = \frac{9}{10}$, specificity is $95\% = \frac{19}{20}$)

- The predicted test distribution is:

$$test_*(prior) = \tfrac{27}{200}|\, p\,\rangle + \tfrac{173}{200}|\, n\,\rangle = 0.135|\, p\,\rangle + 0.865|\, n\,\rangle.$$

This gives the 13.5% likelihood of positive tests.

Divergence between states

Divergence between states

For $\omega, \rho \in \mathcal{D}(X)$ the Kullback-Leibler divergence, or *KL-divergence*, or simply *divergence*, of ω from ρ is:

$$D_{KL}(\omega, \rho) \coloneqq \sum_{x \in X} \omega(x) \cdot \log\left(\frac{\omega(x)}{\rho(x)}\right).$$

Divergence between states

For $\omega, \rho \in \mathcal{D}(X)$ the Kullback-Leibler divergence, or *KL-divergence*, or simply *divergence*, of ω from ρ is:

$$D_{KL}(\omega, \rho) \coloneqq \sum_{x \in X} \omega(x) \cdot \log\left(\frac{\omega(x)}{\rho(x)}\right).$$

It is one standard way to compare states.

Divergence between states

For $\omega, \rho \in \mathcal{D}(X)$ the Kullback-Leibler divergence, or *KL-divergence*, or simply *divergence*, of ω from ρ is:

$$D_{KL}(\omega, \rho) \coloneqq \sum_{x \in X} \omega(x) \cdot \log\left(\frac{\omega(x)}{\rho(x)}\right).$$

It is one standard way to compare states.

Lemma (Basic divergence properties)

(1) $D_{KL}(\omega, \rho) \geq 0$, *with* $D_{KL}(\omega, \rho) = 0$ *iff* $\omega = \rho$

(2) *But:* $D_{KL}(\omega, \rho) \neq D_{KL}(\rho, \omega)$, *in general*

(3) *Also (but not used):* $D_{KL}\big(c_*(\omega),\, c_*(\rho)\big) \leq D_{KL}(\omega, \rho)$

(4) *And:* $D_{KL}\big(\omega \otimes \omega',\, \rho \otimes \rho'\big) = D_{KL}\big(\omega, \rho\big) + D_{KL}\big(\omega', \rho'\big)$

Predicates and transformations

Predicates and transformations

A predicate on a set X is a function $p\colon X \to [0,1]$.

Predicates and transformations

A predicate on a set X is a function $p\colon X \to [0,1]$.

- Each subset/event $E \subseteq X$ forms a 'sharp' predicate, via the indicator function $\mathbf{1}_E\colon X \to [0,1]$
- For each $x \in X$ write $\mathbf{1}_x = \mathbf{1}_{\{x\}}$ for the point predicate, sending $x' \neq x$ to 0 and x to 1.

Predicates and transformations

A predicate on a set X is a function $p \colon X \to [0,1]$.

- Each subset/event $E \subseteq X$ forms a 'sharp' predicate, via the indicator function $1_E \colon X \to [0,1]$
- For each $x \in X$ write $1_x = 1_{\{x\}}$ for the point predicate, sending $x' \neq x$ to 0 and x to 1.

Given a channel $c \colon X \to\!\!\!\!\circ\; Y$ and a predicate q on Y, one defines predicate transformation $c^*(q)$, as predicate on X.

iCIS | Digital Security
Radboud University

Predicates and transformations

A predicate on a set X is a function $p\colon X \to [0,1]$.

- Each subset/event $E \subseteq X$ forms a 'sharp' predicate, via the indicator function $1_E\colon X \to [0,1]$
- For each $x \in X$ write $1_x = 1_{\{x\}}$ for the point predicate, sending $x' \neq x$ to 0 and x to 1.

Given a channel $c\colon X \rightsquigarrow Y$ and a predicate q on Y, one defines predicate transformation $c^*(q)$, as predicate on X.

Explicitly, on $x \in X$,

$$c^*(q)(x) \coloneqq \sum_{y \in Y} c(x)(y) \cdot q(y).$$

iCIS | Digital Security
Radboud University

Predicates and transformations

A predicate on a set X is a function $p\colon X \to [0,1]$.

- Each subset/event $E \subseteq X$ forms a 'sharp' predicate, via the indicator function $1_E\colon X \to [0,1]$
- For each $x \in X$ write $1_x = 1_{\{x\}}$ for the point predicate, sending $x' \neq x$ to 0 and x to 1.

Given a channel $c\colon X \rightsquigarrow Y$ and a predicate q on Y, one defines predicate transformation $c^*(q)$, as predicate on X.

Explicitly, on $x \in X$,

$$c^*(q)(x) \coloneqq \sum_{y \in Y} c(x)(y) \cdot q(y).$$

Note: state tranformation c_* goes in forward direction, along the channel, and predicate transformation c^* goes backward.

Validity and conditioning

Validity and conditioning

(1) For a state ω on a set X, and a predicate p on X define validity as:

$$\omega \models p \quad := \quad \sum_{x \in X} \omega(x) \cdot p(x) \;\; \in \;\; [0,1]$$

It describes the expected value of p in ω.

Validity and conditioning

(1) For a state ω on a set X, and a predicate p on X define validity as:

$$\omega \models p \quad := \quad \sum_{x \in X} \omega(x) \cdot p(x) \;\in\; [0,1]$$

It describes the expected value of p in ω.

(2) If $\omega \models p$ is non-zero, we define the conditional distribution $\omega|_p$ as:

$$\omega|_p(x) \;:=\; \frac{\omega(x) \cdot p(x)}{\omega \models p} \qquad \text{that is} \qquad \omega|_p \;=\; \sum_{x \in X} \frac{\omega(x) \cdot p(x)}{\omega \models p} \big| x \big\rangle.$$

This normalised product $\omega|_p$ of ω and p is the Bayesian update.

Validity and conditioning example

Validity and conditioning example

- Take $X = \{1, 2, 3, 4, 5, 6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|\,1\,\rangle + \frac{1}{6}|\,2\,\rangle + \frac{1}{6}|\,3\,\rangle + \frac{1}{6}|\,4\,\rangle + \frac{1}{6}|\,5\,\rangle + \frac{1}{6}|\,6\,\rangle$

Validity and conditioning example

- Take $X = \{1,2,3,4,5,6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$
- Take the predicate $evenish \colon X \to [0,1]$

$$
\begin{array}{lll}
evenish(1) = \frac{1}{5} & evenish(3) = \frac{1}{10} & evenish(5) = \frac{1}{10} \\
evenish(2) = \frac{9}{10} & evenish(4) = \frac{9}{10} & evenish(6) = \frac{4}{5}
\end{array}
$$

Validity and conditioning example

- Take $X = \{1, 2, 3, 4, 5, 6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$
- Take the predicate $evenish\colon X \to [0,1]$

$$evenish(1) = \tfrac{1}{5} \qquad evenish(3) = \tfrac{1}{10} \qquad evenish(5) = \tfrac{1}{10}$$
$$evenish(2) = \tfrac{9}{10} \qquad evenish(4) = \tfrac{9}{10} \qquad evenish(6) = \tfrac{4}{5}$$

- The validity of $evenish$ for our fair dice is:

$$dice \models evenish = \sum_x dice(x) \cdot evenish(x) = \tfrac{1}{2}.$$

Validity and conditioning example

- Take $X = \{1, 2, 3, 4, 5, 6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$
- Take the predicate $evenish \colon X \to [0, 1]$

$$\begin{array}{lll} evenish(1) = \frac{1}{5} & evenish(3) = \frac{1}{10} & evenish(5) = \frac{1}{10} \\ evenish(2) = \frac{9}{10} & evenish(4) = \frac{9}{10} & evenish(6) = \frac{4}{5} \end{array}$$

- The validity of $evenish$ for our fair dice is:

$$dice \models evenish = \sum_x dice(x) \cdot evenish(x) = \frac{1}{2}.$$

- If we take $evenish$ as evidence, we can update our $dice$ state and get:

$$dice\big|_{evenish} = \sum_x \frac{dice(x) \cdot evenish(x)}{dice \models evenish} \, \big| x \big\rangle$$

Validity and conditioning example

- Take $X = \{1,2,3,4,5,6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$
- Take the predicate $evenish\colon X \to [0,1]$

$$\begin{array}{lll} evenish(1) = \frac{1}{5} & evenish(3) = \frac{1}{10} & evenish(5) = \frac{1}{10} \\ evenish(2) = \frac{9}{10} & evenish(4) = \frac{9}{10} & evenish(6) = \frac{4}{5} \end{array}$$

- The validity of $evenish$ for our fair dice is:

$$dice \models evenish = \sum_x dice(x) \cdot evenish(x) = \tfrac{1}{2}.$$

- If we take $evenish$ as evidence, we can update our $dice$ state and get:

$$dice\big|_{evenish} = \sum_x \frac{dice(x)\cdot evenish(x)}{dice \models evenish}\,|x\rangle$$

$$= \tfrac{1/6\cdot 1/5}{1/2}|1\rangle + \tfrac{1/6\cdot 9/10}{1/2}|2\rangle + \tfrac{1/6\cdot 1/10}{1/2}|3\rangle + \tfrac{1/6\cdot 9/10}{1/2}|4\rangle + \tfrac{1/6\cdot 1/10}{1/2}|5\rangle + \tfrac{1/6\cdot 4/5}{1/2}|6\rangle$$

Validity and conditioning example

- Take $X = \{1,2,3,4,5,6\}$ with state dice $\in \mathcal{D}(X)$
  - Explicitly: $dice = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$
- Take the predicate $evenish \colon X \to [0,1]$

$$\begin{array}{lll} evenish(1) = \frac{1}{5} & evenish(3) = \frac{1}{10} & evenish(5) = \frac{1}{10} \\ evenish(2) = \frac{9}{10} & evenish(4) = \frac{9}{10} & evenish(6) = \frac{4}{5} \end{array}$$

- The validity of $evenish$ for our fair dice is:

$$dice \models evenish = \sum_x dice(x) \cdot evenish(x) = \tfrac{1}{2}.$$

- If we take $evenish$ as evidence, we can update our $dice$ state and get:

$$\begin{aligned} dice\big|_{evenish} &= \sum_x \tfrac{dice(x)\cdot evenish(x)}{dice \models evenish} \big| x \big\rangle \\ &= \tfrac{1/6 \cdot 1/5}{1/2}\big|1\big\rangle + \tfrac{1/6 \cdot 9/10}{1/2}\big|2\big\rangle + \tfrac{1/6 \cdot 1/10}{1/2}\big|3\big\rangle + \tfrac{1/6 \cdot 9/10}{1/2}\big|4\big\rangle + \tfrac{1/6 \cdot 1/10}{1/2}\big|5\big\rangle + \tfrac{1/6 \cdot 4/5}{1/2}\big|6\big\rangle \\ &= \tfrac{1}{15}\big|1\big\rangle + \tfrac{3}{10}\big|2\big\rangle + \tfrac{1}{30}\big|3\big\rangle + \tfrac{3}{10}\big|4\big\rangle + \tfrac{1}{30}\big|5\big\rangle + \tfrac{4}{15}\big|6\big\rangle. \end{aligned}$$

Two basic results about validity $\models$

Two basic results about validity $\models$

Theorem (Validity and transformation)

For channel $c \colon X \to Y$, state σ on X, predicate q on Y,

$$c_*(\sigma) \models q \quad = \quad \sigma \models c^*(q)$$

Two basic results about validity $\models$

Theorem (Validity and transformation)

For channel $c\colon X \rightsquigarrow Y$, *state* σ *on* X, *predicate* q *on* Y,

$$c_*(\sigma) \models q \;=\; \sigma \models c^*(q)$$

Theorem (Validity increase)

For a state ω *and predicate* p *(on the same set, with non-zero validity),*

$$\omega|_p \models p \;\geq\; \omega \models p$$

Two basic results about validity $\models$

Theorem (Validity and transformation)

For channel $c\colon X \rightsquigarrow Y$, state σ on X, predicate q on Y,

$$c_*(\sigma) \models q \;=\; \sigma \models c^*(q)$$

Theorem (Validity increase)

For a state ω and predicate p (on the same set, with non-zero validity),

$$\omega|_p \models p \;\geq\; \omega \models p$$

Informally, absorbing evidence p into state ω, makes p more true.

The "dagger" of a channel: Bayesian inversion

The “dagger” of a channel: Bayesian inversion

Assume a channel $c\colon X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

The "dagger" of a channel: Bayesian inversion

Assume a channel $c\colon X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:

The "dagger" of a channel: Bayesian inversion

Assume a channel $c: X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:
  (1) the point predicate 1_y on Y

The "dagger" of a channel: Bayesian inversion

Assume a channel $c\colon X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:

  (1) the point predicate 1_y on Y
  (2) its transformation $c^*(1_y)$ along c, as predicate on X

The "dagger" of a channel: Bayesian inversion

Assume a channel $c\colon X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:
  (1) the point predicate 1_y on Y
  (2) its transformation $c^*(1_y)$ along c, as predicate on X
  (3) the updated state $\sigma|_{c^*(1_y)} \in \mathcal{D}(X)$.

The "dagger" of a channel: Bayesian inversion

Assume a channel $c\colon X \rightarrow\!\!\!\circ\, Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:

  (1) the point predicate 1_y on Y
  (2) its transformation $c^*(1_y)$ along c, as predicate on X
  (3) the updated state $\sigma|_{c^*(1_y)} \in \mathcal{D}(X)$.

- This yields an inverted channel, the "dagger"

$$Y \xrightarrow{c^\dagger_\sigma} X \qquad \text{with} \qquad c^\dagger_\sigma(y) \coloneqq \sigma|_{c^*(1_y)}$$

The “dagger” of a channel: Bayesian inversion

Assume a channel $c\colon X \rightsquigarrow Y$ and a state $\sigma \in \mathcal{D}(X)$.

- For an element $y \in Y$ we can form:
  (1) the point predicate 1_y on Y
  (2) its transformation $c^*(1_y)$ along c, as predicate on X
  (3) the updated state $\sigma|_{c^*(1_y)} \in \mathcal{D}(X)$.
- This yields an inverted channel, the “dagger”

$$Y \xrightarrow{c^\dagger_\sigma} X \qquad \text{with} \qquad c^\dagger_\sigma(y) \coloneqq \sigma|_{c^*(1_y)}$$

- This forms a dagger functor on a symmetric monoidal category.
  - see e.g. Clerc, Dahlqvist, Danos, Garnier in FoSSaCS 2017
  - with disintegration: Cho-Jacobs in MSCS'19; Fritz in AIM'20.

Pearl and Jeffrey, formulated via channels (JAIR'19)

Pearl and Jeffrey, formulated via channels (JAIR'19)

Set-up:

- a channel $c\colon X \rightsquigarrow Y$ with a (prior) state $\sigma \in \mathcal{D}(X)$ on the domain
- evidence on Y, that we wish to use to update σ

Pearl and Jeffrey, formulated via channels (JAIR'19)

Set-up:

- a channel $c\colon X \rightsquigarrow Y$ with a (prior) state $\sigma \in \mathcal{D}(X)$ on the domain
- evidence on Y, that we wish to use to update σ

- **Pearl's update rule**

- **Jeffrey's update rule**

Pearl and Jeffrey, formulated via channels (JAIR'19)

Set-up:

- a channel $c \colon X \rightsquigarrow Y$ with a (prior) state $\sigma \in \mathcal{D}(X)$ on the domain
- evidence on Y, that we wish to use to update σ

- **Pearl's update rule**
  (1) Evidence is a predicate q on Y
  (2) Updated state:

$$\sigma_P \coloneqq \sigma|_{c^*(q)}$$

- **Jeffrey's update rule**

Pearl and Jeffrey, formulated via channels (JAIR'19)

Set-up:

- a channel $c\colon X \rightsquigarrow Y$ with a (prior) state $\sigma \in \mathcal{D}(X)$ on the domain
- evidence on Y, that we wish to use to update σ

- **Pearl's update rule**
  (1) Evidence is a predicate q on Y
  (2) Updated state:
$$\sigma_P \;\coloneqq\; \sigma|_{c^*(q)}$$

- **Jeffrey's update rule**
  (1) Evidence is state τ on Y
  (2) Updated state:
$$\sigma_J \;\coloneqq\; \Big(c^\dagger_\sigma\Big)_*(\tau) = \sum_{y \in Y} \tau(y) \cdot \Big(\sigma|_{c^*(1_y)}\Big)$$

Main optimisation results

Main optimisation results

Theorem

Let $c\colon X \rightsquigarrow Y$ *be a channel, with prior state* $\sigma \in \mathcal{D}(X)$.

Main optimisation results

Theorem

Let $c\colon X \rightsquigarrow Y$ be a channel, with prior state $\sigma \in \mathcal{D}(X)$.

(1) *Pearl increases validity of evidence predicate q on Y,*

$$c_*(\sigma_P) \models q \;\geq\; c_*(\sigma) \models q \quad \text{for} \quad \sigma_P = \sigma|_{c^*(q)}.$$

Main optimisation results

Theorem

Let $c\colon X \rightsquigarrow Y$ be a channel, with prior state $\sigma \in \mathcal{D}(X)$.

(1) *Pearl increases validity of evidence predicate q on Y,*

$$c_*(\sigma_P) \models q \;\geq\; c_*(\sigma) \models q \quad \textit{for} \;\; \sigma_P = \sigma|_{c^*(q)}.$$

(2) *Jeffrey decreases divergence from evidence distribution τ on Y,*

$$D_{KL}\big(\tau, c_*(\sigma_J)\big) \leq D_{KL}\big(\tau, c_*(\sigma)\big) \qquad \textit{for} \; \sigma_J = \Big(c_\sigma^\dagger\Big)_*(\tau).$$

Main optimisation results

Theorem

Let $c\colon X \to Y$ be a channel, with prior state $\sigma \in \mathcal{D}(X)$.

(1) *Pearl increases validity of evidence predicate q on Y,*

$$c_*(\sigma_P) \models q \;\geq\; c_*(\sigma) \models q \quad \text{for} \quad \sigma_P = \sigma|_{c^*(q)}.$$

(2) *Jeffrey decreases divergence from evidence distribution τ on Y,*

$$D_{KL}\big(\tau, c_*(\sigma_J)\big) \leq D_{KL}\big(\tau, c_*(\sigma)\big) \qquad \text{for } \sigma_J = \Big(c^\dagger_\sigma\Big)_*(\tau).$$

- The proof of Pearly is easy, but for Jeffrey it is remarkably hard.
- Jeffrey's KL-decrease is missing in the predictive coding literature — although it forms the basis of error reduction

The disease-test example: Pearl and Jeffrey

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**:
- **Jeffrey**:

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**: Take predicate $q = \frac{8}{10}1_p + \frac{2}{10}1_n$. Then:

$$pearl := prior\big|_{test^*(q)} = \frac{74}{281}|\,d\,\rangle + \frac{207}{281}|\,d^{\perp}\,\rangle \approx 0.263|\,d\,\rangle + 0.737|\,d^{\perp}\,\rangle.$$

- **Jeffrey**:

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**: Take predicate $q = \frac{8}{10} 1_p + \frac{2}{10} 1_n$. Then:

$$pearl := prior\big|_{test^*(q)} = \tfrac{74}{281}|\, d\,\rangle + \tfrac{207}{281}|\, d^{\perp}\,\rangle \approx 0.263|\, d\,\rangle + 0.737|\, d^{\perp}\,\rangle.$$

  There is a validity increase from

$$test_*(prior) \models q = 0.281 \quad \text{to} \quad test_*(pearl) \models q = 0.364.$$

- **Jeffrey**:

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**: Take predicate $q = \frac{8}{10}\mathbf{1}_p + \frac{2}{10}\mathbf{1}_n$. Then:

$$pearl \coloneqq prior\big|_{test^*(q)} = \frac{74}{281}|\,d\,\rangle + \frac{207}{281}|\,d^\perp\,\rangle \approx 0.263|\,d\,\rangle + 0.737|\,d^\perp\,\rangle.$$

  There is a validity increase from

$$test_*(prior) \models q = 0.281 \quad \text{to} \quad test_*(pearl) \models q = 0.364.$$

- **Jeffrey**: Take distribution $\tau = \frac{8}{10}|\,p\,\rangle + \frac{2}{10}|\,n\,\rangle$. Then:

$$jeffrey \coloneqq \left(test^\dagger_{prior}\right)_*(\tau) = \frac{278}{519}|\,d\,\rangle + \frac{241}{519}|\,d^\perp\,\rangle \approx 0.536|\,d\,\rangle + 0.464|\,d^\perp\,\rangle.$$

iCIS | Digital Security
Radboud University

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**: Take predicate $q = \frac{8}{10}1_p + \frac{2}{10}1_n$. Then:

$$pearl := prior\big|_{test^*(q)} = \frac{74}{281}|\,d\,\rangle + \frac{207}{281}|\,d^\perp\,\rangle \approx 0.263|\,d\,\rangle + 0.737|\,d^\perp\,\rangle.$$

  There is a validity increase from

$$test_*(prior) \models q = 0.281 \quad \text{to} \quad test_*(pearl) \models q = 0.364.$$

- **Jeffrey**: Take distribution $\tau = \frac{8}{10}|\,p\,\rangle + \frac{2}{10}|\,n\,\rangle$. Then:

$$jeffrey := \Big(test^\dagger_{prior}\Big)_*(\tau) = \frac{278}{519}|\,d\,\rangle + \frac{241}{519}|\,d^\perp\,\rangle \approx 0.536|\,d\,\rangle + 0.464|\,d^\perp\,\rangle.$$

  There is a divergence decrease from

$$D_{KL}\Big(\tau,\ test_*(prior)\Big) = 1.13 \quad \text{to} \quad D_{KL}\Big(\tau,\ test_*(jeffrey)\Big) = 0.186.$$

The disease-test example: Pearl and Jeffrey

Recall there is 80% certainty about a positive test

- **Pearl**: Take predicate $q = \frac{8}{10}1_p + \frac{2}{10}1_n$. Then:

$$pearl := prior\big|_{test^*(q)} = \frac{74}{281}|\, d\,\rangle + \frac{207}{281}|\, d^\perp\,\rangle \approx 0.263|\, d\,\rangle + 0.737|\, d^\perp\,\rangle.$$

  There is a validity increase from

$$test_*(prior) \models q = 0.281 \quad \text{to} \quad test_*(pearl) \models q = 0.364.$$

- **Jeffrey**: Take distribution $\tau = \frac{8}{10}|\, p\,\rangle + \frac{2}{10}|\, n\,\rangle$. Then:

$$jeffrey := \Big(test^\dagger_{prior}\Big)_*(\tau) = \frac{278}{519}|\, d\,\rangle + \frac{241}{519}|\, d^\perp\,\rangle \approx 0.536|\, d\,\rangle + 0.464|\, d^\perp\,\rangle.$$

  There is a divergence decrease from

$$D_{KL}\Big(\tau,\ test_*(prior)\Big) = 1.13 \quad \text{to} \quad D_{KL}\Big(\tau,\ test_*(jeffrey)\Big) = 0.186.$$

Possible interpretation: in Pearl's case the tester sets the evidence uncertainty, whereas in Jeffrey's case the evaluater sets the uncertainy.

Where we are, so far

Expectation Maximisation I

Expectation Maximisation I

- Expectation Maximisation is one of the core algorithms of (unsupervised) machine learning
  - it tries to recognise structure in multiple data points

Expectation Maximisation I

- Expectation Maximisation is one of the core algorithms of (unsupervised) machine learning
  - it tries to recognise structure in multiple data points
- These data points are organised as a multiset, also written via kets, e.g. as for an urn with coloured balls $3|R\rangle + 4|G\rangle + 5|B\rangle$

Expectation Maximisation I

- Expectation Maximisation is one of the core algorithms of (unsupervised) machine learning
  - it tries to recognise structure in multiple data points
- These data points are organised as a multiset, also written via kets, e.g. as for an urn with coloured balls $3|R\rangle + 4|G\rangle + 5|B\rangle$
- We write $\mathcal{M}(Y)$ for the set of multisets with elements from Y
  - $\psi \in \mathcal{M}(Y)$ is $\sum_i n_i|y_i\rangle$, with multiplicity $n_i \in \mathbb{N}$ for $y_i \in Y$
  - or as $\psi\colon Y \to \mathbb{N}$, finite support $supp(\psi) = \{y \in Y \mid \psi(y) \neq 0\}$

Expectation Maximisation I

- Expectation Maximisation is one of the core algorithms of (unsupervised) machine learning
  - it tries to recognise structure in multiple data points
- These data points are organised as a multiset, also written via kets, e.g. as for an urn with coloured balls $3|R\rangle + 4|G\rangle + 5|B\rangle$
- We write $\mathcal{M}(Y)$ for the set of multisets with elements from Y
  - $\psi \in \mathcal{M}(Y)$ is $\sum_i n_i|y_i\rangle$, with multiplicity $n_i \in \mathbb{N}$ for $y_i \in Y$
  - or as $\psi\colon Y \to \mathbb{N}$, finite support $supp(\psi) = \{y \in Y \mid \psi(y) \neq 0\}$
- Each (non-empty) multiset can be turned into a distribution
  - this works via normalisation, called frequentist learning $Flrn$
  - Explicitly, for non-zero $\psi \in \mathcal{M}(Y)$,

$$Flrn(\psi) := \sum_{y \in Y} \frac{\psi(y)}{\|\psi\|}|y\rangle \quad \text{where } \|\psi\| := \sum_{y \in Y} \psi(y).$$

Expectation Maximisation I

- Expectation Maximisation is one of the core algorithms of (unsupervised) machine learning
  - it tries to recognise structure in multiple data points
- These data points are organised as a multiset, also written via kets, e.g. as for an urn with coloured balls $3|R\rangle + 4|G\rangle + 5|B\rangle$
- We write $\mathcal{M}(Y)$ for the set of multisets with elements from Y
  - $\psi \in \mathcal{M}(Y)$ is $\sum_i n_i|y_i\rangle$, with multiplicity $n_i \in \mathbb{N}$ for $y_i \in Y$
  - or as $\psi\colon Y \to \mathbb{N}$, finite support $supp(\psi) = \{y \in Y \mid \psi(y) \neq 0\}$
- Each (non-empty) multiset can be turned into a distribution
  - this works via normalisation, called frequentist learning $Flrn$
  - Explicitly, for non-zero $\psi \in \mathcal{M}(Y)$,

$$Flrn(\psi) \coloneqq \sum_{y\in Y} \frac{\psi(y)}{\|\psi\|}\big|y\big\rangle \quad \text{where } \|\psi\| \coloneqq \sum_{y\in Y} \psi(y).$$

  - E.g. $Flrn\Big(3|R\rangle + 4|G\rangle + 5|B\rangle\Big) = \frac{1}{4}|R\rangle + \frac{1}{3}|G\rangle + \frac{5}{12}|B\rangle$.

Expectation Maximisation II

Expectation Maximisation II

General goal: given a datapoints multiset $\psi \in \mathcal{M}(Y)$, find a mixture of distributions:

$$\omega := r_1 \cdot \omega_1 + \cdots + r_N \cdot \omega_N \qquad \text{with minimal} \qquad D_{KL}\big(Flrn(\psi), \omega\big)$$

Expectation Maximisation II

General goal: given a datapoints multiset $\psi \in \mathcal{M}(Y)$, find a mixture of distributions:

$$\omega := r_1 \cdot \omega_1 + \cdots + r_N \cdot \omega_N \qquad \text{with minimal} \qquad D_{KL}\big(Flrn(\psi), \omega\big)$$

- More explicitly, $\omega_i \in \mathcal{D}(Y)$ and $r_i \in [0, 1]$ with $\sum_i r_i = 1$

Expectation Maximisation II

General goal: given a datapoints multiset $\psi \in \mathcal{M}(Y)$, find a mixture of distributions:

$$\omega := r_1 \cdot \omega_1 + \cdots + r_N \cdot \omega_N \qquad \text{with minimal} \qquad D_{KL}\big(Flrn(\psi), \omega\big)$$

- More explicitly, $\omega_i \in \mathcal{D}(Y)$ and $r_i \in [0,1]$ with $\sum_i r_i = 1$
- The number N is fixed in advance. Once can form:

$$X := \{1, 2, \ldots, N\}$$
$$\sigma := \sum_{1 \leq i \leq N} r_i |\, i \,\rangle \in \mathcal{D}(X)$$

$$c \colon X \rightsquigarrow Y \quad \text{with} \quad c(i) := \omega_i$$

Expectation Maximisation II

General goal: given a datapoints multiset $\psi \in \mathcal{M}(Y)$, find a mixture of distributions:

$$\omega := r_1 \cdot \omega_1 + \cdots + r_N \cdot \omega_N \qquad \text{with minimal} \qquad D_{KL}\big(Flrn(\psi), \omega\big)$$

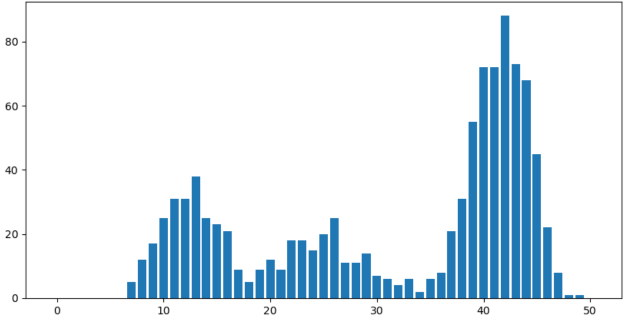
- More explicitly, $\omega_i \in \mathcal{D}(Y)$ and $r_i \in [0,1]$ with $\sum_i r_i = 1$
- The number N is fixed in advance. Once can form:

$$X := \{1, 2, \ldots, N\}$$
$$\sigma := \sum_{1 \leq i \leq N} r_i |\, i \,\rangle \;\in\; \mathcal{D}(X)$$
$$c \colon X \rightsquigarrow Y \;\text{ with }\; c(i) := \omega_i$$

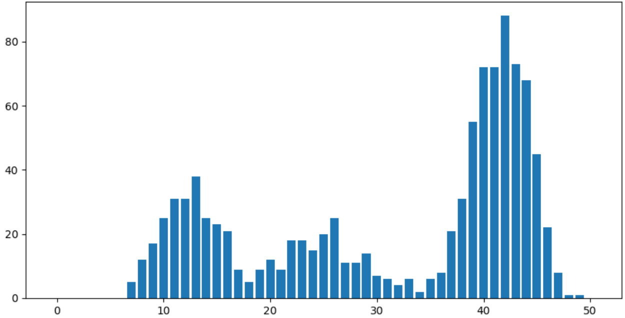
- The goal is then to minimise $D_{KL}\big(Flrn(\psi),\, c_*(\sigma)\big)$
  - this is the same goal of Jeffrey's update rule
  - but now we wish to learn both a distribution σ and a channel c

Running example

Running example

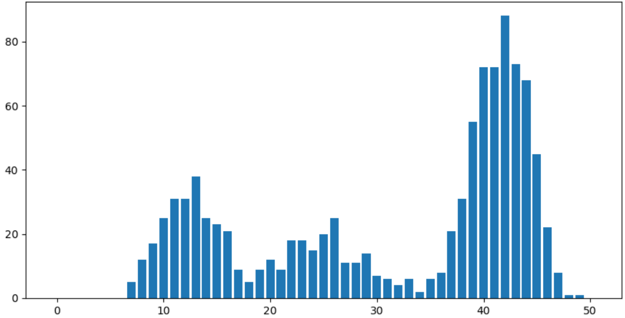


Running example



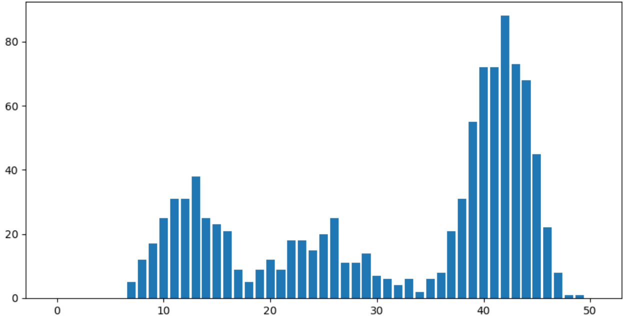
- A multiset of size 1000 on $\{0, 1, \ldots, 50\}$, plotted as a histogram

Running example



- A multiset of size 1000 on $\{0, 1, \ldots, 50\}$, plotted as a histogram
- One may recognise a mixture of three binomial distributions

Running example



- A multiset of size 1000 on $\{0, 1, \ldots, 50\}$, plotted as a histogram
- One may recognise a mixture of three binomial distributions
- The aim of Expectation Maximisation is to uncover the mixture distribution and also the (means of the) three binomials

Setting of EM

Setting of EM

- We have:
  - data multiset $\psi \in \mathcal{M}(Y)$
  - mixture distribution $\sigma \in \mathcal{D}(X)$
  - a channel $c\colon X \rightsquigarrow Y$

iCIS | Digital Security
Radboud University

Setting of EM

- We have:
  - data multiset $\psi \in \mathcal{M}(Y)$
  - mixture distribution $\sigma \in \mathcal{D}(X)$
  - a channel $c \colon X \rightsquigarrow Y$
- The aim of one iteration of the EM-algorithm is:
  - to find new σ' and c', such that:
  - $D_{KL}\big(Flrn(\psi), \, c'_*(\sigma')\big) \,\leq\, D_{KL}\big(Flrn(\psi), \, c_*(\sigma)\big)$

Setting of EM

- We have:
  - data multiset $\psi \in \mathcal{M}(Y)$
  - mixture distribution $\sigma \in \mathcal{D}(X)$
  - a channel $c \colon X \rightsquigarrow Y$
- The aim of one iteration of the EM-algorithm is:
  - to find new σ' and c', such that:
  - $D_{KL}\big(Flrn(\psi), {c'}_*(\sigma')\big) \leq D_{KL}\big(Flrn(\psi), c_*(\sigma)\big)$
- This is iterated until some (divergence) fixed point is reached.

Ideal EM

Ideal EM

Let $\psi \in \mathcal{M}(Y)$, $\sigma \in \mathcal{D}(X)$ and $c\colon X \rightsquigarrow Y$ be given.

Ideal EM

Let $\psi \in \mathcal{M}(Y)$, $\sigma \in \mathcal{D}(X)$ and $c\colon X \rightsquigarrow Y$ be given.

Theorem

Form:

- *the dagger* $d := c^{\dagger}_{\sigma}\colon Y \rightsquigarrow X$
- *the Jeffrey update* $\sigma' := d_{*}(Flrn(\psi)) \in \mathcal{D}(X)$
- *the double dagger* $c' := d^{\dagger}_{Flrn(\psi)}\colon X \rightsquigarrow Y$

Then: $c'_{*}(\sigma') = Flrn(\psi)$.

Ideal EM

Let $\psi \in \mathcal{M}(Y)$, $\sigma \in \mathcal{D}(X)$ and $c\colon X \rightsquigarrow Y$ be given.

Theorem

Form:

- *the dagger* $d := c^{\dagger}_{\sigma}\colon Y \rightsquigarrow X$
- *the Jeffrey update* $\sigma' := d_*(Flrn(\psi)) \in \mathcal{D}(X)$
- *the double dagger* $c' := d^{\dagger}_{Flrn(\psi)}\colon X \rightsquigarrow Y$

Then: $c'_*(\sigma') = Flrn(\psi)$.

In this way one gets a perfect match, in one iteration.

EM for running, binomials example

EM for running, binomials example

We force the double dagger to be of the right binomial shape. This is not "perfect", so needs several iterations.

iCIS | Digital Security
Radboud University

EM for running, binomials example

We force the double dagger to be of the right binomial shape. This is not "perfect", so needs several iterations.

The EM-algorithm can be described in a few lines:

```
def BinomialMixEM (dist, chan):
    dagger = chan†_dist
    # E-step, as Jeffrey update
    new_dist = dagger_*(Flrn(ψ))
    # M-part, via means of double dagger
    double_dagger = dagger†_Flrn(ψ)
    def new_chan(x) = bn[K]( mean( double_dagger(x) ) / K )
    return (new_dist, new_chan)
```

Running EM example: results

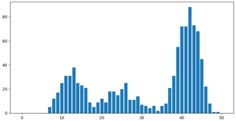
Running EM example: results

- We start from the mixture distribution

$$\tfrac{1}{4}\cdot binom[K]\big(\tfrac{1}{4}\big) \;+\; \tfrac{1}{6}\cdot binom[K]\big(\tfrac{1}{2}\big) \;+\; \tfrac{7}{12}\cdot binom[K]\big(\tfrac{5}{6}\big)$$

Running EM example: results

- We start from the mixture distribution
$$\tfrac{1}{4} \cdot binom[K]\big(\tfrac{1}{4}\big) + \tfrac{1}{6} \cdot binom[K]\big(\tfrac{1}{2}\big) + \tfrac{7}{12} \cdot binom[K]\big(\tfrac{5}{6}\big)$$
- We sample 1000 points from this distribution, giving the earlier histogram:

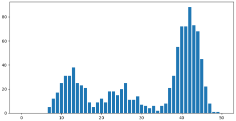


Running EM example: results

- We start from the mixture distribution

$$\frac{1}{4} \cdot binom[K]\left(\frac{1}{4}\right) + \frac{1}{6} \cdot binom[K]\left(\frac{1}{2}\right) + \frac{7}{12} \cdot binom[K]\left(\frac{5}{6}\right)$$

- We sample 1000 points from this distribution, giving the earlier histogram:



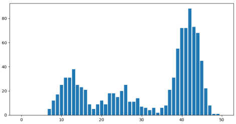
The aim is to reconstruct the original mixture from these data alone

Running EM example: results

- We start from the mixture distribution

$$\tfrac{1}{4} \cdot binom[K]\big(\tfrac{1}{4}\big) \;+\; \tfrac{1}{6} \cdot binom[K]\big(\tfrac{1}{2}\big) \;+\; \tfrac{7}{12} \cdot binom[K]\big(\tfrac{5}{6}\big)$$

- We sample 1000 points from this distribution, giving the earlier histogram:



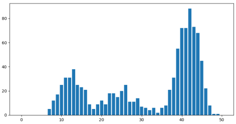
The aim is to reconstruct the original mixture from these data alone

- After 10 EM-iterations the divergence stabilises at 0.026.

	mixture	**means**
original		
via EM:		

Running EM example: results

- We start from the mixture distribution
$$\tfrac{1}{4} \cdot binom[K]\big(\tfrac{1}{4}\big) + \tfrac{1}{6} \cdot binom[K]\big(\tfrac{1}{2}\big) + \tfrac{7}{12} \cdot binom[K]\big(\tfrac{5}{6}\big)$$
- We sample 1000 points from this distribution, giving the earlier histogram:



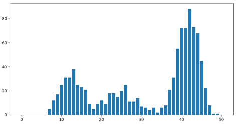
The aim is to reconstruct the original mixture from these data alone

- After 10 EM-iterations the divergence stabilises at 0.026.

	mixture	means		
original	$\frac{1}{4}\vert 1\rangle + \frac{1}{6}\vert 2\rangle + \frac{7}{12}\vert 3\rangle \approx$ $0.25\vert 1\rangle + 0.167\vert 2\rangle + 0.583\vert 3\rangle$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{5}{6}$
via EM:				

Running EM example: results

- We start from the mixture distribution
$$\tfrac{1}{4} \cdot binom[K]\big(\tfrac{1}{4}\big) \;+\; \tfrac{1}{6} \cdot binom[K]\big(\tfrac{1}{2}\big) \;+\; \tfrac{7}{12} \cdot binom[K]\big(\tfrac{5}{6}\big)$$
- We sample 1000 points from this distribution, giving the earlier histogram:



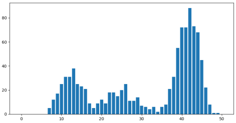
The aim is to reconstruct the original mixture from these data alone

- After 10 EM-iterations the divergence stabilises at 0.026.

	mixture	means		
original	$\frac{1}{4}\vert 1\rangle + \frac{1}{6}\vert 2\rangle + \frac{7}{12}\vert 3\rangle \approx$ $0.25\vert 1\rangle + 0.167\vert 2\rangle + 0.583\vert 3\rangle$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{5}{6}$
via EM:	$0.574\vert 1\rangle + 0.175\vert 2\rangle + 0.252\vert 3\rangle$			

Running EM example: results

- We start from the mixture distribution
$$\tfrac{1}{4} \cdot binom[K]\big(\tfrac{1}{4}\big) + \tfrac{1}{6} \cdot binom[K]\big(\tfrac{1}{2}\big) + \tfrac{7}{12} \cdot binom[K]\big(\tfrac{5}{6}\big)$$
- We sample 1000 points from this distribution, giving the earlier histogram:



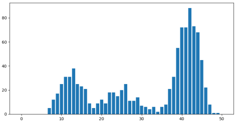
The aim is to reconstruct the original mixture from these data alone

- After 10 EM-iterations the divergence stabilises at 0.026.

	mixture	**means**		
original	$\frac{1}{4}\vert 1\rangle + \frac{1}{6}\vert 2\rangle + \frac{7}{12}\vert 3\rangle \approx$ $0.25\vert 1\rangle + 0.167\vert 2\rangle + 0.583\vert 3\rangle$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{5}{6}$
via EM:	$0.574\vert 1\rangle + 0.175\vert 2\rangle + 0.252\vert 3\rangle$	0.831	0.505	0.254

Running EM example: results

- We start from the mixture distribution
$$\tfrac{1}{4}\cdot binom[K]\big(\tfrac{1}{4}\big) \;+\; \tfrac{1}{6}\cdot binom[K]\big(\tfrac{1}{2}\big) \;+\; \tfrac{7}{12}\cdot binom[K]\big(\tfrac{5}{6}\big)$$
- We sample 1000 points from this distribution, giving the earlier histogram:



The aim is to reconstruct the original mixture from these data alone

- After 10 EM-iterations the divergence stabilises at 0.026.

	mixture	means		
original	$\frac{1}{4}\vert 1\rangle + \frac{1}{6}\vert 2\rangle + \frac{7}{12}\vert 3\rangle \approx$ $0.25\vert 1\rangle + 0.167\vert 2\rangle + 0.583\vert 3\rangle$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{5}{6}$
via EM:	$0.574\vert 1\rangle + 0.175\vert 2\rangle + 0.252\vert 3\rangle$	0.831	0.505	0.254

- Outcomes are swapped; the mixture has no order

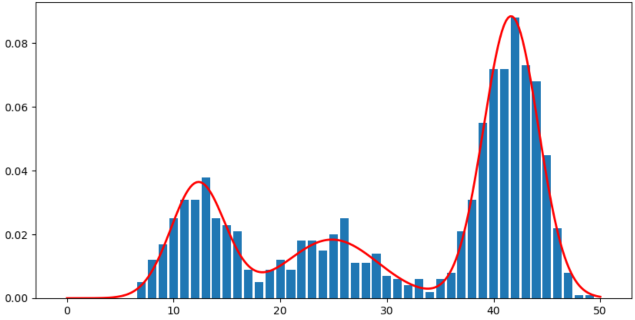
Additional point

Additional point

- The same techniques work in a continuous setting
- they can give for instance a mixture of Gaussians matching the same data:

Additional point

- The same techniques work in a continuous setting
- they can give for instance a mixture of Gaussians matching the same data:



Where we are, so far

Concluding remarks

Concluding remarks

- Updating is one of the magical things in probabilistic logic
  - it is a pillar of the AI-revolution
  - it requires a proper logic, for causality and for 'XAI'
- The two update rules of Pearl and Jeffrey:
  - can give wildly different outcomes
  - are not so clearly distinguished in the literature — probably because fuzzy / soft predicates are not standard
  - have clear formulations/properties in terms of channels: Pearl increases validity, Jeffrey decreases divergence
- The difference Pearl / Jeffrey is of wider significance
  - e.g. EM decreases divergence via Jeffrey, see Wollic'23
  - daggers and double daggers are actually useful
- Challenge: connecting to cognition theory community
  - that's hard, because of differences in language/methods