PARTIAL MARKOV CATEGORIES (DRAFT)

ELENA DI LAVORE [a], MARIO ROMÁN [b], AND PAWEŁ SOBOCIŃSKI [c]

[a] Department of Computer Science, University of Pisa, Italy

[b] Department of Computer Science, University of Oxford, United Kingdom

[c] Department of Software Science, Tallinn University of Technology, Estonia

Abstract. We introduce partial Markov categories as a synthetic framework for synthetic probabilistic inference, blending the work of Cho and Jacobs, Fritz, and Golubtsov on Markov categories with the work of Cockett and Lack on cartesian restriction categories. We describe observations, Bayes' theorem, normalisation, and both Pearl's and Jeffrey's updates in purely abstract categorical terms.

Contents

Key words and phrases: partial Markov category, Markov category, Bayesian inference.

Elena Di Lavore, Mario Román, and Paweł Sobociński were supported by the Safeguarded AI programme of the Advanced Research and Invention Agency. Mario Román was additionally supported by the Air Force Office of Scientific Research under award number FA9550-21-1-0038. This article is based upon work from COST Action EuroProofNet, CA20111 supported by COST (European Cooperation in Science and Technology).

Preprint submitted to
Logical Methods in Computer Science

1. Introduction

A good calculus for modelling decision problems needs to *(i)* express probabilistic processes, to model a stochastic environment in which the agent needs to act, *(ii)* express constraints, to restrict the model to satisfy the specifications of the decision problem, and *(iii)* explicitly capture the implicit assumptions of decision theory. We introduce partial Markov categories: both a syntax for modelling decision problems and a calculus for solving them.

Markov categories [Fri20] are an algebra for probabilistic processes where it is natural to express *conditioning*, *independence* and *Bayesian networks* [Fon13, CJ19, Fri20, JKZ21, JZ20]. However, this language does not allow the encoding of constraints, e.g., we cannot impose *a posteriori* that a certain sampling from a channel must coincide with some observation.

This limitation is due to the structure of Markov categories, which allows resources and processes to be *discarded* but only resources to be *copied*: throwing a coin twice is not the same as throwing it once and copying the result. String diagrams [Bén67] and do-notation [Pat01, HJW$^+$92, LJR24] are two internal languages for morphisms in symmetric monoidal categories [Jef97], and for Markov categories in particular. Figure 1 shows the string diagrams that discard and copy a generic morphism, $f\colon X \to Y$, in a Markov category.

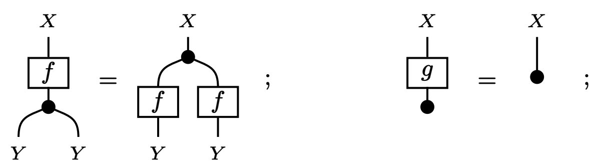


Figure 1: Stochastic processes can be discarded but not copied.

Most copy-discard categories commonly used to encode stochastic processes exhibit the additional property of having *conditionals* (Figure 2). Conditionals ensure that every joint distribution can be split into a marginal and a conditional distribution. While not necessary to encode stochastic processes, this property is essential for *reasoning* about them. In this manuscript, we always consider Markov categories to have conditionals; in standard terminology, our Markov categories are "copy and natural-discard categories with conditionals" and our partial Markov categories are "copy-discard categories with conditionals".

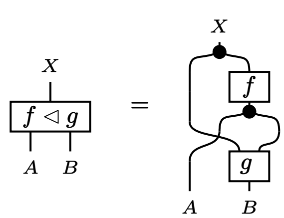


Figure 2: Conditionals require that a stochastic process $(f \triangleleft g)$ be split into a marginal f and a conditional g.

On the other hand, *cartesian restriction categories* [CL07, CO89], and discrete cartesian restriction categories in particular [CGH12, DLNS21], are a calculus of partial processes with *constraints*. Cartesian restriction categories allow the copying and discarding of resources, as in Markov categories. However, processes are now allowed to be copied but not to be discarded (Figure 1).

Discrete cartesian restriction categories additionally possess an *equality constraint* morphism, $\mu_X \colon X \otimes X \to X$, for each object X. Its axioms say that copying resources and then checking that they are equal should be the same as the identity process (Figure 3).

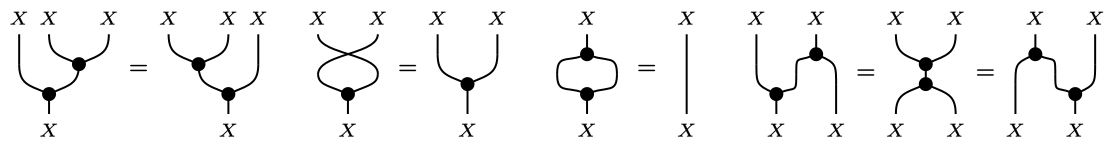


Figure 3: Equality constraint structure and its axioms in a discrete cartesian restriction category.

We introduce *partial Markov categories* and *discrete partial Markov categories*, which extend Markov categories and (discrete) cartesian restriction categories to encode both *probabilistic reasoning* and *constraints*. Discrete partial Markov categories allow the modelling of inference problems: they can express the constraints that we observe via the comparator map.

We claim that the algebra of partial Markov categories is a good theoretical framework for synthetic probabilistic inference: it provides both a convenient syntax in terms of string diagrams that translates to the computations that solve the given inference problem, and a calculus for reasoning about them.

1.1. **Contributions.** Our main contribution is the algebra of partial Markov categories (Definition 3.1) and discrete partial Markov categories (Definition 4.2). Our main result is the construction of a partial Markov category on top of any Markov category such that exact observations are expressed via conditionals of the base Markov category (Theorems 5.3 and 5.4).

We introduce a synthetic description of normalisation (Definition 3.8), and we prove its compositional properties in the abstract setting of partial Markov categories: for instance, $\mathtt{n}(f \fatsemi g) = \mathtt{n}(\mathtt{n}(f) \fatsemi g)$ (Proposition 3.12) and, almost surely, $\mathtt{n}(\mathtt{n}(f)) = \mathtt{n}(f)$ (Proposition 3.11).

We introduce partial Markov categories and discrete partial Markov categories in Sections 3 and 4, where we prove a synthetic version of Bayes' theorem that holds in any discrete partial Markov category (Theorem 4.6). We apply this framework to compare Pearl'sand Jeffrey's update rules (Definitions 4.7 and 4.8 and Proposition 4.9). We prove that the Kleisli category of the *Maybe* monad over a Markov category is a partial Markov category (Section 3.5).

Finally, in Section 5, we provide a calculus for reasoning with exact observations (Definition 5.1). The construction from Theorems 5.3 and 5.4 allows us to express deterministic observations even in non-discrete cases like that of continuous probabilistic processes.

1.2. **Related work.**

Markov categories. The categorical approach to probability theory based on Markov categories [Fri20] has led to the abstraction of various results [FR20, FPR21, FP19, FGP21]. Markov categories have also been applied in the formalisation of Bayes networks and other kinds of probabilistic reasoning in categorical terms [Fon13, JZ20, JKZ21]. The breadth of results and applications of Markov categories suggest that there can be an equally rich landscape for their partial counterpart.

Categories of partial maps. Partiality has long been studied in Computer Science, and even categorical approaches to it date back to the works of Carboni [Car87], Di Paola and Heller [DPH87], Robinson and Rosolini [RR88], and Curien and Obtułowicz [CO89]. However, our categorical structures are more related to more recent work on *restriction categories* by Cockett and Lack [CL02, CL03], and, in particular, *cartesian restriction categories* [CL07] and *discrete cartesian restriction categories* [CGH12, DLNS21].

Copy-discard categories. Categories of partial probabilistic processes—and even the *comparator* morphism—have been previously considered [Pan99, Jac18, CJ19, Ste21], but no comprehensive presentation was given. Bayesian inversion for compact closed copy-discard categories appears in the work of Coecke and Spekkens [CS12]; the relationship between that definition and Definition 3.3 might involve normalisation. Copy-discard categories have been applied to graph rewriting [CG99], where they are called *GS-monoidal*.

Categorical semantics of probabilistic programming. There exists a vast literature on categorical semantics for probabilistic programming languages (for some related, see e.g. [Has97, SV13, SWY$^+$16, HKSY17, EPT17, DK19, VKS19]). However, while the internal language of Markov categories has been studied [FL22], the notion of partial Markov category and its diagrammatic syntax have remained unexplored. Stein and Staton [SS21, Ste21] have recently presented the **Cond** construction for exact conditioning, which could be related to our construction of a partial Markov category of exact observations in Definition 5.1 via normalisation.

Evidential Decision Theory. Existing formalisations of decision theories exist mainly in philosophical terms [Lew81, Joy99, GW06, Ahm14, Gre13, YS17]. We provide a categorical formalisation of Evidential Decision Theory.

2. Background on Markov Categories

This section recalls the categorical approach to probability theory using copy-discard categories. Copy-discard categories are an algebra of processes that compose sequentially and in parallel, with nodes that "copy and discard" the outputs of the processes. They possess a convenient sound and complete syntax in terms of string diagrams [JS91, FL22]. In particular, copy-discard categories and the more specialized Markov categories allow us to reason about probabilistic processes.

2.1. Copy-Discard Categories.

Copy-Discard Categories. Copy-discard categories are categories where every object X has a "copy" morphism, $\nu_X \colon X \to X \otimes X$, and a "discard" morphism $\varepsilon_X \colon X \to I$, forming a commutative comonoid (Figure 4)[1]. The comonoid on the tensor of two objects, $X \otimes Y$, must be the tensor of their comonoid structures: $\nu_{X\otimes Y} = (\nu_X \otimes \nu_Y) ⨟ (\mathrm{id} \otimes \sigma \otimes \mathrm{id})$ and $\varepsilon_{X\otimes Y} = \varepsilon_X \otimes \varepsilon_Y$. The comonoid on the unit object must be the identity, $\nu_I = \mathrm{id}_I$ and $\varepsilon_I = \mathrm{id}_I$.

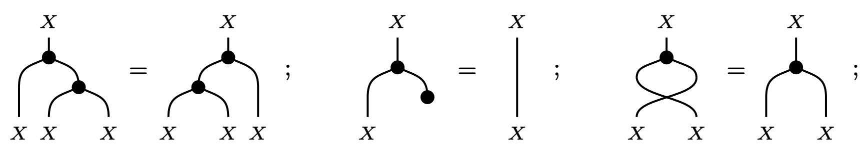


Figure 4: Generators and axioms of a commutative comonoid.

Some morphisms will be copied and discarded by the *copy* and *discard* morphisms: we call these *deterministic* and *total*, respectively. However, contrary to what happens in cartesian categories, *copy* and *discard* are not necessarily natural transformations: morphisms are not always deterministic nor total.

Definition 2.1 (Deterministic and total morphisms)**.** A morphism $f \colon X \to Y$ in a copy-discard category is called *deterministic* if $f ⨟ \nu_Y = \nu_X ⨟ (f \otimes f)$ (Figure 5, left); a morphism $g \colon X \to Y$ is called *total* if $g ⨟ \varepsilon_Y = \varepsilon_X$ (Figure 5, right).

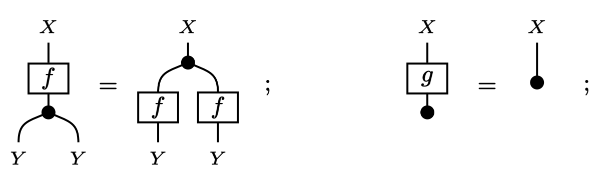


Figure 5: Deterministic morphism (left) and total morphism (right).

The simple algebra of copy-discard categories is already enough to discuss the first basic concepts from probability theory: *marginals*, *conditionals*, and *almost sure equality*.

[1] Copy-discard categories have been called *GS-monoidal categories* when applied to graph rewriting [CG99] and CD-categories when applied to unnormalised probabilistic processes [CJ19]. In categorical quantum mechanics, they are *copy and delete* [HV19]. Fritz and Liang [FL22, Remark 2.2] expand on the history of this structure.

2.2. **Marginals and conditionals.** This section introduces conditionals and their properties (Definition 2.5); their definition needs an auxiliary operation—*marginal composition*—that extends a marginal with a given morphism (Definition 2.3, Figure 6).[2]

Definition 2.2 (Marginals)**.** The *projections* from every pair of objects, X and Y, are the morphisms $\pi_1\colon X \otimes Y \to X$ and $\pi_2\colon X \otimes Y \to Y$ defined by the equations, $\pi_1 = (\mathrm{id}_X \otimes \varepsilon_Y)$ and $\pi_2 = (\varepsilon_X \otimes \mathrm{id}_Y)$. The *marginals* of a two-output morphism, $f\colon X \to A \otimes B$, are the two morphisms resulting from postcomposition with the two projections, $f \fatsemi \pi_1\colon X \to A$ and $f \fatsemi \pi_2\colon X \to B$.

Thus, postcomposition with the (second) projection, $(\bullet \fatsemi \pi_2)$, computes the (second) marginal of a morphism in a copy-discard category. This operation has a section that brings a morphism $f\colon X \to Y$ into a morphism $\varphi(f)\colon X \to X \otimes Y$ and is defined by precomposition with the *copy* morphism, $\varphi(f) = \nu_X \fatsemi (\mathrm{id}_X \otimes f)$. Indeed, we may check that $\varphi(f) \fatsemi \pi_2 = f$, while, in general, $\varphi(f \fatsemi \pi_2) \neq f$. Still, this section-retraction pair induces a new operation.

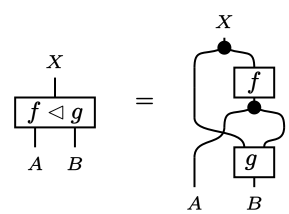


Figure 6: Conditional composition of f and g.

Definition 2.3 (Conditional composition)**.** In a copy-discard category, the *conditional composition* of a morphism $f\colon X \to Y$ with a morphism $g\colon X \otimes Y \to Z$ is the morphism $(f \lhd g)\colon X \to Y \otimes$ defined by the diagram in Figure 6 or, equivalently, by the formula

$$(f \lhd g) = \varphi(f) \fatsemi \varphi(g) \fatsemi \pi_2.$$

Conditional composition is unital: because $\varphi(\varepsilon_X) = \mathrm{id}_X$, it follows that $f \lhd \varepsilon_{X \otimes Y} = f$ and $\varepsilon_X \lhd f = f$. Conditional composition is also associative: note that $\varphi(f \lhd g) = \varphi(f) \fatsemi \varphi(g)$ follows by the associativity of the comonoid and, by a general argument,

$$(f \lhd g) \lhd h = \pi(\varphi((f \lhd g) \lhd h)) = \pi(\varphi(f) \fatsemi \varphi(g) \fatsemi \varphi(h)) = \pi\varphi(f \lhd (g \lhd h)) = f \lhd (g \lhd h).$$

In probability theory, we assume that every morphism can be factored through any of its marginals: for each $f\colon X \to Y \otimes Z$, there exists some $c\colon X \otimes Y \to Z$—its *conditional*—such that $f = (f \fatsemi \pi_1) \lhd c$. This assumption defines Markov categories and partial Markov categories.

[2]Markov categories [Fri20] appear in the literature without the requirement of having conditionals – they are defined as copy-discard categories of total maps. In this text, we call *Markov categories* only to those with conditionals: the Markov categories better suited for synthetic probability all have conditionals. Our change of convention makes the parallel with cartesian categories more explicit: Markov categories are cartesian categories with a weaker splitting condition, instead of any category with copy and uniform discard maps.

2.3. **Markov Categories.** Markov categories are a variant of copy-discard categories, specialized for probability theory. Synthetic probability theory needs—apart from the axioms of copy-discard categories—two extra principles. The first is that every morphism is total: a probabilistic choice that does not affect any outcome could as well not have happened. The second is that every joint morphism can be factored through its marginal: we assume the existence of a morphism such that, when composed with the marginal of a morphism, yields the original morphism. Explicitly, synthetic probability theory assumes the existence of *conditionals* [CJ19, Fri20].

Definition 2.4 (Markov category)**.** A *Markov category* is a copy-discard category with conditionals where all morphisms are total.

Definition 2.5 (Conditionals)**.** A copy-discard category $\mathbb{C}$ has *conditionals* if each morphism $f\colon X \to Y_1 \otimes Y_2$ can be factored through its marginal:

$$f = (f \mathbin{⨟} \pi_1) \triangleleft c_1$$

for some $c_1\colon Y_1 \otimes X \to Y_2$. In other words, some morphism satisfies the equation in Figure 2. In this situation, we say that $c_1\colon Y_1 \otimes X \to Y_2$ is a *conditional* of f with respect to Y_1.

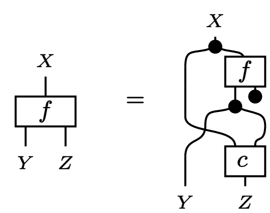


Figure 7: Definition of conditional.

In general, conditionals are not unique: unique conditionals only exist in posetal categories ([FP19, Proposition 11.15]). However, conditionals are "*almost surely* unique" with respect to the marginal. Indeed, many notions in Markov categories are better behaved up to synthetic almost sure equality [FP19, Section 13].[3]

Definition 2.6 (Almost-sure equality)**.** Two morphisms, $g_1, g_2\colon A \otimes X \to B$, are *almost surely equal* with respect to $f\colon X \to A$ (or, f-*almost surely equal*) whenever $f \triangleleft g_1 = f \triangleleft g_2$.

We denote the conditional of a morphism $f\colon X \to Y \otimes Z$ by $\boldsymbol{c}(f)\colon X \otimes Y \to Z$; note that it is defined merely up to $(f \mathbin{⨟} \pi_1)$-almost sure equality. In a certain sense, conditionals generalize how deterministic morphisms split in terms of their projections, $h = \nu \mathbin{⨟} ((h \mathbin{⨟} \pi_1) \otimes (h \mathbin{⨟} \pi_2))$.

Proposition 2.7 (Conditional of a deterministic morphism)**.** *Let $h\colon X \to Y \otimes Z$ be a deterministic morphism. Its conditional can be constructed from composing with projections,*

$$\boldsymbol{c}(h) = \pi_1 \mathbin{⨟} h \mathbin{⨟} \pi_2, \text{ when } h \text{ is deterministic.}$$

[3]There exists a more general notion of almost sure equality [FGL+23, Definition 2.1.1] and conditional composition [DdFR22, Remark A.14]; for simplicity, we restrict it to the case that concerns us in this work.

2.4. **Some Markov categories.** This section gives examples of Markov categories. The canonical example of Markov category is the monoidal Kleisli category of the finitary distribution monad [Fri20], Stoch (Proposition 2.9). Standard Borel spaces and kernels between them form a Markov category that models continuous probability, BorelStoch (Definition 2.11).

Definition 2.8. A *finitely supported distribution* on a set X is a function $\sigma \colon X \to [0,1]$ such that the set $\{x \in X : \sigma(x) \neq 0\}$ is finite and $\sum_{x \in X} \sigma(x) = 1$. We indicate the set of finitely supported distributions on a set X as $\mathbf{D}(X)$.

This mapping, $\mathbf{D}$, can be extended to a functor $\mathbf{D} \colon \mathsf{Set} \to \mathsf{Set}$ and to a monad, called the *finitary distribution monad*. For any function, $f \colon X \to Y$, the corresponding $\mathbf{D}(f) \colon \mathbf{D}(X) \to \mathbf{D}(Y)$ associates to a distribution $\sigma \in \mathbf{D}(X)$ a new distribution, $\mathbf{D}(f)(\sigma) \in \mathbf{D}(Y)$, defined by

$$\mathbf{D}(f)(\sigma)(y) = \sum_{x \in X, f(x) = y} \sigma(x).$$

The monad multiplication, $\mu_X \in \mathbf{D}(\mathbf{D}(X)) \to \mathbf{D}(X)$, associates to a distribution $\sigma \in \mathbf{D}(\mathbf{D}(X))$ the new distribution $\mu_X(\sigma) \in \mathbf{D}(X)$ defined by $\mu_X(\sigma)(x) = \sum_{\tau \in \mathbf{D}(X)} \tau(x) \cdot \sigma(\tau)$. The monad unit, $\delta \colon X \to \mathbf{D}(X)$, associates to each element, $x \in X$, its Dirac delta, δ_x. The finitary distribution monad is monoidal with structural transformation $t_{X,Y} \colon \mathbf{D}(X) \otimes \mathbf{D}(Y) \to \mathbf{D}(X \otimes Y)$ given by $t_{X,Y}(\sigma, \tau)(x, y) = \sigma(x) \cdot \tau(y)$.

Proposition 2.9. *The Kleisli category of the finitary distribution monad,* $\mathbf{D}$*, is a copy-discard category with conditionals, called* Stoch*. Its copy and discard morphisms are lifted from* Set*, by post-composing with the unit of the monad. A morphism* $f \colon X \to Y$ *in* Stoch *represents a stochastic channel: we interpret the value of* $f(x)$ *in* $y \in Y$ *as the probability of* y *given* x *according to the channel* f*, and we indicate it as* $f(y \mid x)$*.*

Remark 2.10 (Reading string diagrams in Stoch)**.** The copy-discard category structure of Stoch allows an intuitive reading of a string diagram in terms of its components. The value of a morphism is the multiplication of the values of all the components summed over all wires that are not inputs nor outputs. For example, the morphism in Figure 8 evaluates to the formula

$$f(z_1, z_2 \mid x) = \sum_{y \in Y} g(y \mid x) \cdot h_1(z_1 \mid y) \cdot h_2(z_2 \mid y).$$

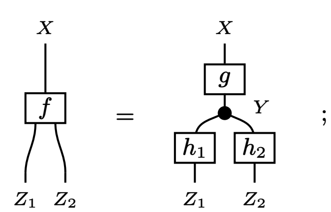


Figure 8: String diagrammatic equation for the formula in Remark 2.10.

Standard Borel spaces. Measurable maps between measurable spaces form a copy-discard category. However, this category does not have conditionals. Instead, we consider its subcategory on standard Borel spaces, which has conditionals and is a Markov category [Fri20].

Definition 2.11 (Category of standard Borel spaces)**.** The category BorelStoch has standard Borel spaces (X, Σ_X) as objects, where X is a set and Σ_X is a σ-algebra on X.

A morphism of Borel spaces, $f\colon (X, \Sigma_X) \to (Y, \Sigma_Y)$, is a function $f\colon \Sigma_Y \times X \to [0, 1]$ such that, for each subset of the codomain σ-algebra, $T \in \Sigma_Y$, its probability given an input, $f(T \mid -)\colon X \to [0, 1]$ defines a measurable function, and such that, for each input $x \in X$, the function $f(- \mid x)\colon \Sigma_Y \to [0, 1]$ is a probability measure.

The composition of morphisms of Borel spaces is given by the following Lebesgue integral [Pan09]; the identity morphism, $\mathrm{id}_X\colon (X, \Sigma_X) \to (X, \Sigma_X)$, is given by the Dirac measure,

$$(f \fatsemi g)(T \mid x) = \int_{y \in Y} g(T|y) \cdot f(dy|x); \qquad \mathrm{id}_X(T \mid x) = \begin{cases} 1 & \text{if } x \in T, \\ 0 & \text{otherwise.} \end{cases}$$

for any two morphisms $f\colon (X, \Sigma_X) \to (Y, \Sigma_Y)$ and $g\colon (Y, \Sigma_Y) \to (Z, \Sigma_Z)$. The category BorelStoch is moreover monoidal. The monoidal product is defined on objects using the σ-algebra generated by the cartesian product of elements of its factors, $(X, \Sigma_X) \otimes (Y, \Sigma_Y) = (X \otimes Y, \Sigma_X \otimes \Sigma_Y)$, where

$$\Sigma_X \otimes \Sigma_Y = \langle S_1 \times S_2 \mid S_1 \in \Sigma_X, S_2 \in \Sigma_Y \rangle.$$

The monoidal tensor is defined on morphisms by

$$(f \otimes f')(T \mid x, x') = \int_{(y,y') \in T} f(dy \mid x) \cdot f'(dy' \mid x').$$

The monoidal unit is the one-element set.

Definition 2.12 (Giry monad)**.** The Giry functor, $\mathbf{G}\colon \mathsf{Meas} \to \mathsf{Meas}$, assigns to a set X the set of probability measures on it. Given a morphism $f\colon (X, \Sigma_X) \to (Y, \Sigma_Y)$, it is defined by $\mathbf{G}(f)(T \mid \sigma) = \sigma(f^{-1}(T))$.

This functor is a monad [Gir82]: its unit associates to each $x \in X$ its Dirac measure, $\eta_{(X,\Sigma_X)}(x) = \delta_x$; its multiplication is defined as

$$\mu_{(X,\sigma_X)}(T|p) = \int_{\tau \in \mathbf{G}(X,\sigma_X)} \tau(T) \cdot p(d\tau).$$

The Giry monad is monoidal with a structural transformation analogous to that of the finitary distribution monad: $t_{X,Y}(\sigma, \tau)(S, T) = \sigma(S) \cdot \tau(T)$.

Remark 2.13. The category BorelStoch can also be seen as the Kleisli category of the Giry monad $\mathbf{G}\colon \mathsf{Meas} \to \mathsf{Meas}$ restricted to standard Borel spaces [Gir82, Fri20].

2.5. **Subdistributions.** Let us now develop two non-examples of Markov category. A *subdistribution* over X is a distribution whose total probability is allowed to be less than 1 [Jac18, CJ19]. In other words, it is a distribution over $X + 1$. This means that a morphism $f\colon X \to Y$ in subStoch represents a *substochastic channel*: a stochastic channel that has some probability of failure.

The symmetric monoidal Kleisli category of the finitary subdistribution monoidal monad, $\mathbf{D}_{\leq 1}$, is the main example for (discrete) partial Markov categories. It is the semantic universe where we compute solutions to discrete inference problems.

Definition 2.14. A *finitely supported subdistribution* on a set X is a function $\sigma\colon X \to [0,1]$ such that the set $\{x \in X : \sigma(x) \neq 0\}$ is finite and $\sum_{x \in X} \sigma(x) \leq 1$. We indicate the set of subdistributions on a set X as $\mathbf{D}_{\leq 1}(X)$.

Proposition 2.15. *The mapping $\mathbf{D}_{\leq 1}$ can be extended to a functor $\mathbf{D}_{\leq 1}\colon \mathsf{Set} \to \mathsf{Set}$ and to a monad, called the* finitary subdistribution monad. *For a function $f\colon X \to Y$, $\mathbf{D}_{\leq 1}(f)$ is defined by*

$$\mathbf{D}_{\leq 1}(f)(y|\sigma) = \sum_{f(x)=y} \sigma(x),$$

for any subdistribution $\sigma \in \mathbf{D}_{\leq 1}(X)$ and any element $y \in Y$.

The monad multiplication $\mu_X\colon \mathbf{D}_{\leq 1}(\mathbf{D}_{\leq 1}(X)) \to \mathbf{D}_{\leq 1}(X)$ and unit $\eta_X\colon X \to \mathbf{D}_{\leq 1}(X)$ are defined analogously to those of the finitary distribution monad $\mathbf{D}$ (Definition 2.8). Explicitly, the monad multiplication is defined by

$$\mu_X(x|p) = \sum_{\sigma \in \mathbf{D}_{\leq 1}(X)} p(\sigma) \cdot \sigma(x);$$

and the monad unit is defined by $\eta_X(x|x') = \delta_{x'}(x)$, where $\delta_x \in \mathbf{D}(X)$ is the Dirac distribution that assigns probability 1 to x and 0 to everything else.

Remark 2.16. The fact that $\mathbf{D}_{\leq 1}$ is a functor and a monad can be proved from a distributive law between the *Maybe* monad $(- + 1)$ with the finitary *distribution* monad $\mathbf{D}$: the category Set of sets and functions is distributive. This implies that $(- + 1)$ can be lifted to the Kleisli category $\mathsf{kl}(\mathbf{D})$ and that there is a distributive law between $\mathbf{D}$ and $(- + 1)$. Their composition is the finitary subdistribution monad $\mathbf{D}_{\leq 1} = \mathbf{D}(- + 1)$. The distributive law $\mathsf{d}\colon \mathbf{D}(-) + 1 \to \mathbf{D}(- + 1)$ is defined by $\mathsf{d}_X(\sigma) = \sigma^*$ and $\mathsf{d}_X(\perp) = \delta_\perp$, where $\sigma \in \mathbf{D}(X)$ and σ^* is σ extended to $X + 1$ by $\sigma^*(\perp) = 0$. See the work of Jacobs for details [Jac18, Section 4].

Call $\mathsf{subStoch}$ the Kleisli category of the subdistribution monad, $\mathbf{D}_{\leq 1}$. Even when this category has conditionals, not every map is total, which prevents it from being a Markov category.

Proposition 2.17. $\mathsf{SubStoch}$ *is a copy-discard category with conditionals.*

Standard Borel spaces. The analogue of $\mathsf{subStoch}$ for Borel spaces is $\mathsf{BorelStoch}_{\leq 1}$. A morphism $f\colon X \to Y$ in $\mathsf{BorelStoch}_{\leq 1}$ represents a stochastic channel that has some probability of failure, i.e. $f(x)$ is a subprobability measure.

Definition 2.18 (Subprobability measure)**.** A *subprobability measure* p on a measurable space (X, σ_X) is a measurable function such that $p(X) \leq 1$.

Definition 2.19 (Category subdistributions between standard Borel spaces)**.** The objects of $\mathsf{BorelStoch}_{\leq 1}$ are standard Borel spaces, (X, Σ_X), where X is a set and Σ_X is a σ-algebra on X.

A morphism $f\colon (X, \sigma_X) \to (Y, \sigma_Y)$ in $\mathsf{BorelStoch}_{\leq 1}$ is a function $f\colon \Sigma_Y \times X \to [0,1]$ such that, for each measurable set $T \in \sigma_Y$, the conditional $f(T \mid -)\colon X \to [0,1]$ is a measurable function, and, for each element $x \in X$, the function $f(-|x)\colon \Sigma_Y \to [0,1]$ forms a subprobability measure.

Identities and composition are defined as in $\mathsf{BorelStoch}$.

Remark 2.20 (Panangaden's monad [Pan99])**.** The category $\mathsf{BorelStoch}_{\leq 1}$ arises as the Kleisli category of Panangaden's monad, $\mathbf{G}_{\leq 1}$. Its underlying functor is the composition of the Giry functor and the Maybe functor: $\mathbf{G}_{\leq 1} = \mathbf{G}(- + 1)$. There is a candidate distributive law $\mathsf{d}\colon \mathbf{G}(-) + 1 \to \mathbf{G}(- + 1)$ defined by $\mathsf{d}_X(\sigma) = \sigma^*$ and $\mathsf{d}_X(\perp) = \delta_\perp$, where σ^* is the extension of σ to $X + 1$ by $\sigma^*(\{\perp\}) = 0$. Proposition 2.21 shows that this is indeed a distributive law between the Giry monad and the Maybe monad.

Proposition 2.21. *There is a distributive law between the Giry monad and the Maybe monad:* $\mathsf{d}\colon \mathbf{G}(-) + 1 \to \mathbf{G}(- + 1)$.

Proof. We will deduce the existence of the distributive law from the fact that the composite functor is a monad with the multiplication and units given by the potential distributive law.

The composite functor $\mathbf{G}(- + 1) = \mathbf{G}_{\leq 1}$ is a monad [Pan99], with multiplication and unit given by compositions of the multiplications and units of $\mathbf{G}$ and $(- + 1)$. For the units, this is easy to see as the unit of $\mathbf{G}(- + 1)$ is just the inclusion of the unit of $\mathbf{G}$, and the inclusion $\mathbf{G} \to \mathbf{G}(- + 1)$ is given by the unit of $(- + 1)$. For the multiplications, we can check that $\mu = (\mathrm{id} \otimes \mathsf{d} \otimes \mathrm{id}) \fatsemi (\mu_1 \otimes \mu_2)$. In fact, let $p \in \mathbf{G}(\mathbf{G}(X + 1) + 1)$. Then,

$$((\mathrm{id} \otimes \mathsf{d}_X \otimes \mathrm{id}) \fatsemi (\mu_1 \otimes \mu_2))(p) = \int_{\tau \in \mathbf{G}(X+1)} \tau(-) \cdot p(d\tau),$$

which corresponds with the definition of the multiplication of $\mathbf{G}(- + 1)$. The components defined in Remark 2.20 form a natural transformation. These conditions already imply that there is in fact a distributive law between $\mathbf{G}$ and $(- + 1)$. □

Remark 2.22 (Domain of definition)**.** In $\mathsf{subStoch}$, the composition of a morphism $f\colon X \to Y$ with the discard map yields the probability of "success": the probability that it does not fail,

$$(f \fatsemi \varepsilon)(* \mid x) = \sum_{y \in Y} f(y \mid x) = 1 - f(\perp \mid x).$$

In *restriction categories*, Cockett and Lack [CL02] call $(f \fatsemi \varepsilon)\colon X \to I$ the *domain of definition* of f; we extend this nomenclature to all copy-discard categories.

For instance, in $\mathsf{subStoch}$, each morphism has a fuzzy domain of definition where the degree of membership of the element $x \in X$ is $1 - f(\perp \mid x)$. This fuzzy set is crisp exactly when $(f \fatsemi \varepsilon)$ is itself deterministic (Figure 9): in this case, f factors through the inclusion $\mathbf{D}(Y) + 1 \hookrightarrow \mathbf{D}_{\leq 1}$.

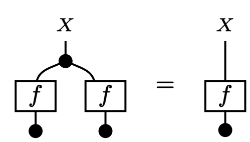


Figure 9: Deterministic domain of definition.

Remark 2.23 (Towards partial Markov categories)**.** These last two categories, $\mathsf{subStoch}$ and $\mathsf{BorelStoch}_{\leq 1}$, are still copy-discard categories with conditionals; but not all of their morphisms are total. This means that they are not Markov categories. We claim that totality is not essential for modelling stochastic processes and that dropping this assumption allows us to pursue a synthetic theory of inferential update and observations. This is the main idea behind *partial Markov categories*.

3. Partial Markov Categories

3.1. **Partial Markov categories.** Cartesian restriction categories extend cartesian categories encoding *partiality*: a map may not be defined on all its inputs and fail when evaluated on inputs outside its domain of definition. We introduce *partial Markov categories* as a similar extension of Markov categories to encode partial stochastic processes, i.e., stochastic processes that have a probability of failure on each one of their inputs. Partiality is obtained by dropping naturality of the discard maps, i.e. by allowing morphisms to be non-total.

This extra generality will lead us later to discrete partial Markov categories: partial Markov categories with the ability of comparing two outputs. These are the analogue of *discrete cartesian restriction categories* [CL02, DLNS21].

Cartesian category	Markov category
Cartesian restriction category	Partial Markov category
Discrete cartesian restriction category	Discrete partial Markov category

Definition 3.1 (Partial Markov category)**.** A *partial Markov category* is a copy-discard category with conditionals.

In a Markov category, all morphisms—and in particular all conditionals—are total. In partial Markov categories, we drop the totality assumption while still obtaining marginals by discarding one of the outputs. Instead of being total, conditionals in partial Markov categories are almost surely total.

Proposition 3.2 (Conditionals are almost surely total)**.** *Conditionals in partial Markov categories are almost surely total with respect to the marginal.*

Proof. Let us show that, given $f\colon X \to Y \otimes Z$, its conditional, $\boldsymbol{c}(f)\colon X \otimes Y \to Z$, is $(f \mathbin{⨟} \pi_1)$-almost surely total, meaning that $(f \mathbin{⨟} \pi_1) \triangleleft (\boldsymbol{c}(f) \mathbin{⨟} \varepsilon) = (f \mathbin{⨟} \pi_1) \triangleleft \varepsilon$. By *(i)* the definition of projection, *(ii)* the definition of conditional, and *(iii)* counitality, we have that

$$(f \mathbin{⨟} \pi_1) \triangleleft (\boldsymbol{c}(f) \mathbin{⨟} \varepsilon) \overset{(i)}{=} ((f \mathbin{⨟} \pi_1) \triangleleft \boldsymbol{c}(f)) \mathbin{⨟} \pi_2 \overset{(ii)}{=} f \mathbin{⨟} \pi_2 \overset{(iii)}{=} (f \mathbin{⨟} \pi_2) \triangleleft \varepsilon.$$

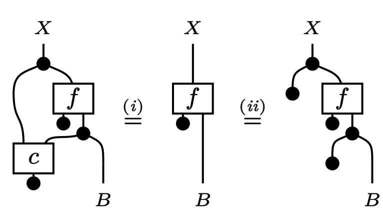


Figure 10: Conditionals are almost surely total.

Equivalently, the string diagram reasoning in Figure 10 shows that the conditional is $(f \mathbin{⨟} \pi_2)$-almost surely total. We employ *(i)* the definition of conditional, and *(ii)* counitality. □

3.2. **Bayesian inversions.** The *Bayesian inversion* of a stochastic channel $g\colon X \to Y$ with respect to a distribution p over X is the stochastic channel $g_{\dagger}(p)\colon Y \to X$ defined by

$$g_{\dagger}(p)(x \mid y) = \frac{g(y \mid x) \cdot p(x)}{\sum_{x_\bullet \in X} g(y \mid x_\bullet) \cdot p(x_\bullet)},$$

for any $y \in Y$ with positive probability, meaning that $\sum_{x_\bullet \in X} g(y \mid x_\bullet) \cdot p(x_\bullet) > 0$.

Bayesian inversions can be defined abstractly in partial Markov categories, as they can be in Markov categories [Fri20, Proposition 11.17]: Bayesian inversions are a particular case of conditionals. We state this result for partial Markov categories (Proposition 3.4) as a straightforward generalisation of that for Markov categories [Fri20, Proposition 11.17].

Definition 3.3 (Bayesian inversion). A *Bayesian inversion* of a morphism $g\colon X \to Y$ with respect to $p\colon I \to X$ is a morphism $g_{\dagger}(p)\colon Y \to X$ satisfying the equation in Figure 11.

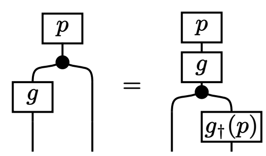


Figure 11: Bayesian inversion.

Proposition 3.4. *In a partial Markov category, all Bayesian inversions exist. The Bayesian inversion of g with respect to p is $(p \,⨾\, g)$-almost surely unique.*

Proof. The Bayesian inversion of g with respect to p is given by the conditional of the morphism $p \,⨾\, \nu_X \,⨾\, (g \otimes \mathrm{id})$. Explicitly,

$$g_{\dagger}(p) = \boldsymbol{c}(p \,⨾\, \nu_X \,⨾\, (g \otimes \mathrm{id})).$$

Note that $(p \,⨾\, \nu_X \,⨾\, (g \otimes \mathrm{id}) \,⨾\, \pi_1) = (p \,⨾\, g)$, making the conditional $(p \,⨾\, g)$-almost surely unique. □

Bayesian inversions can be computed compositionally: results for doing so appear in the work of Fritz [Fri20, Lemma 11.11], for Markov categories, and Jacobs [Jac19, Section 5.1], for Stoch. We recast these in the setting of partial Markov categories.

Proposition 3.5 (Bayesian inversion of a composite). *A Bayesian inversion of a composite channel $(f \,⨾\, g)\colon X \to Y$ with respect to a distribution $p\colon I \to X$ can be computed by first inverting f with respect to p and then inverting g with respect to $(p \,⨾\, f)$:*

$$(f \,⨾\, g)_{\dagger}(p) = g_{\dagger}(p \,⨾\, f) \,⨾\, f_{\dagger}(p).$$

Proposition 3.6 (Conditional of a composite). *Let $f\colon X \to Y$ and $g\colon Y \to Z \otimes W$ be two morphisms of a partial Markov category. The conditional of their composition can be computed from a Bayesian inversion, $\boldsymbol{b}(f, g) = (g \,⨾\, \pi_1)_{\dagger}(f)$, and the conditional of the second morphism, $\boldsymbol{c}(g)\colon Y \otimes Z \to W$, as in Figure 12.*

Proof. We reason by string diagrams. We use *(i)* the definition of conditional; *(ii)* the definition of Bayesian inversion; and *(iii)* associativity and commutativity. □

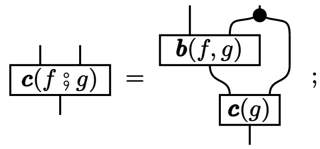


Figure 12: Conditional of a composition.

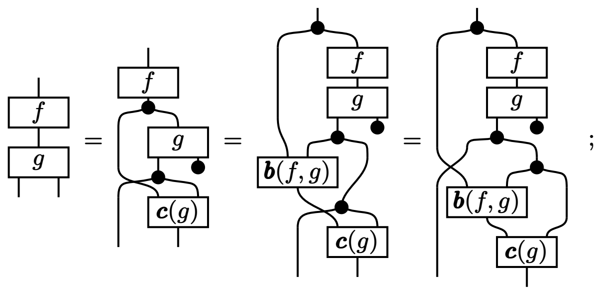


Figure 13: Construction of the conditional of a composition.

Example 3.7 (Bayesian inversions via density functions, c.f. [CJ19, Example 3.9])**.** In BorelStoch, the Bayesian inversion of a channel $f\colon (X, \Sigma_X) \to \mathbf{G}(Y, \Sigma_Y)$ with respect to a prior, $p \in \mathbf{G}(X, \Sigma_X)$, is a channel satisfying

$$\int_{x \in S_X} \int_{y \in S_Y} c(dy|x) \cdot \sigma(dx) = \int_{x \in S_X} \int_{y \in S_Y} c_\dagger(\sigma)(dx|y) \cdot \int_{x_0 \in X} c(dy|x_0) \cdot \sigma(dx_0),$$

for each $S_X \in \Sigma_X$ and $S_Y \in \Sigma_Y$. These inversions need not to exist in the more general category of arbitrary Borel spaces [FP19] and, even for standard Borel spaces, they could be difficult to compute. Note how the inversion, $c_\dagger(p)$, cannot depend on dy but just on y.

However, distributions and channels are often given by density functions,

$$p(S_X) = \int_{x \in S_X} p^{pdf}(x)\, dx; \qquad f(S_Y|x) = \int_{y \in S_Y} f^{pdf}(y, x)\, dy.$$

In these cases, we can construct a Bayesian inversion as follows: we compute its density function and integrate it to obtain the following channel,

$$f_\dagger(p)^{pdf}(x, y) = \frac{f^{pdf}(y, x) \cdot p^{pdf}(x)}{\int_{x_0 \in X} f^{pdf}(y, x_0) \cdot p^{pdf}(x_0)\, dx_0}; \quad f_\dagger(p)(S_X, y) = \int_{x \in S_X} f_\dagger(p)^{pdf}(x, y)\, dx,$$

which does constitute a Bayesian inversion in this case,

$$\int_{x \in S_X} \int_{y \in S_Y} f_\dagger(p)^{pdf}(x, y) \cdot \left(\int_{x_0 \in X} f^{pdf}(y, x_0) \cdot p^{pdf}(x_0)\, dx_0 \right) =$$
$$\int_{x \in S_X} \int_{y \in S_Y} f^{pdf}(y, x) \cdot p^{pdf}(x)\, dx.$$

3.3. **Normalisation.** The normalisation of a partial stochastic channel $f\colon X \to Y$ is, classically, uniquely defined on each $x \in X$ with non-zero probability of sucess, i.e. $1 - f(\bot \mid x) > 0$; while, whenever $f(\bot \mid x) = 1$, any output acts as a normalisation.

$$\mathtt{n}(f)(y \mid x) = \frac{f(y \mid x)}{1 - f(\bot \mid x)}.$$

Normalisations can be defined in any partial Markov category. In Markov categories, this notion trivialises: because all morphisms are required to be total, every morphism is its own normalisation.

Definition 3.8 (Normalisation)**.** Let $f\colon X \to Y$ be a morphism in a partial Markov category. A *normalisation* of f is any morphism $\mathtt{n}(f)\colon X \to Y$ such that $f = (f \mathbin{⨟} \varepsilon) \triangleleft \mathtt{n}(f)$.

Proposition 3.9 (Existence and uniqueness of normalisations)**.** *In a partial Markov category, all normalisations exist. The normalisation of $f\colon X \to Y$ is $(f \mathbin{⨟} \varepsilon_Y)$-almost surely unique.*

Remark 3.10. Normalisation induces an operator taking a morphism $f\colon X \to Y$, to a morphism $\mathtt{n}(f)\colon X \to Y$ defined up to $(f \mathbin{⨟} \varepsilon_Y)$-almost sure equality. Multiple authors depict this operator as a shaded box: the contents of the box represent the morphism that is being normalised; the operator must appear conditionally composed with its discarded version in order for it to be well-defined. The box is not functorial. This shaded box can be explained in terms of a collage of string diagrams, but we do not go into details here.

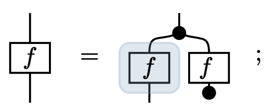


Figure 14: Normalisation of a morphism.

Normalisations are the particular case of conditionals for the domain of definition: the normalisation of f is given by the conditional on the monoidal unit of $(f \mathbin{⨟} \varepsilon)$. Even when normalisations are not unique, they are almost surely unique. Even when normalisations are not idempotent, they are almost surely idempotent.

Proposition 3.11 (Normalisations are almost surely idempotent)**.** *The normalisation of the normalisation of a morphism $f\colon X \to Y$ is $(f \mathbin{⨟} \varepsilon_Y)$-almost surely equal to the original normalisation,*

$$\mathtt{n}(\mathtt{n}(f)) =_{(f ⨟ \varepsilon)} \mathtt{n}(f).$$

Proof. We reason by the string diagrams of Figure 15. □

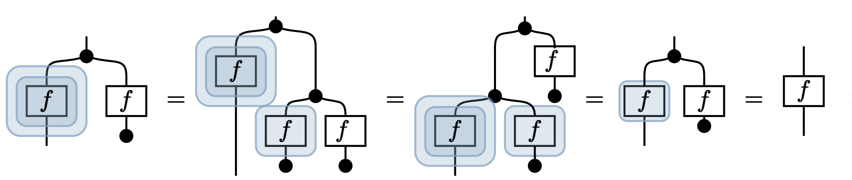


Figure 15: Normalisation is idempotent.

Normalisation is not functorial (and the shaded box is not a functor box); functoriality, $\mathtt{n}(f \mathbin{⨟} g) = \mathtt{n}(f) \mathbin{⨟} \mathtt{n}(g)$, already fails in SubStoch. What is instead true is that normalising at the end of a computation is the same as normalising at any point, $\mathtt{n}(f \mathbin{⨟} g) = \mathtt{n}(\mathtt{n}(f) \mathbin{⨟} g)$.

Proposition 3.12 (Normalisation precomposes)**.** *Let $f: X \to Y$ and $g: Y \to Z$ be two morphisms in a partial Markov category, then $\mathtt{n}(f \mathbin{⨟} g) = \mathtt{n}(\mathtt{n}(f) \mathbin{⨟} g)$.*

Proof. We reason by the string diagrams in Figure 16. □

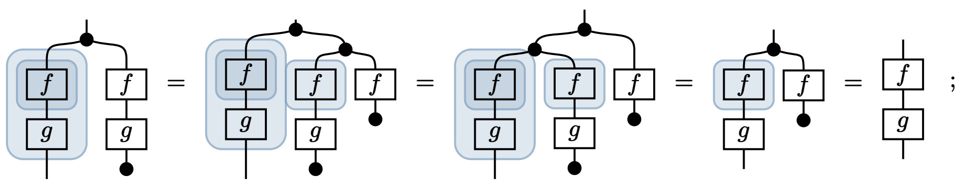


Figure 16: Normalisation precomposes.

Normalisation does not influence conditioning: the conditionals of a normalisation are still conditionals for the original morphism. Theorem 5.4 relies on this result.

Proposition 3.13. *Let $f: X \to Y \otimes Z$ be a morphism in a partial Markov category. Then, the conditionals of the normalisation of f are conditionals of f.*

3.4. **Deterministic domains.** We could be tempted to impose that conditionals be total morphisms, but this is not possible even in SubStoch: the "always fail" map $\perp: A \to 0$ cannot have a total conditional. However, we may impose a weaker condition: that conditionals have a *deterministic domain*. This is a route we have explored in previous work [DR23]; in this manuscript, we will study it without assuming that all conditionals have a deterministic domain.

Definition 3.14 (Deterministic domain)**.** A *deterministic-domain* morphism $f: A \to B$ in a copy-discard category is a morphism such that its domain of definition, $(f \mathbin{⨟} \varepsilon): A \to I$, is deterministic (Figure 9).

Remark 3.15. All deterministic morphisms are deterministic-domain morphisms; discarding their outputs (Remark 2.22) defines their domain in the sense of Cockett and Lack [CL02]. Lemma 3.16 precises this relationship.

Lemma 3.16. *In a partial Markov category, a morphism $f: A \to B$ has a deterministic domain if and only if it is its own normalisation: $f =_{f ⨟ \varepsilon} \mathtt{n}(f)$, or, in other words, $f = (f ⨟ \varepsilon) \triangleleft f$ (Figure 17).*

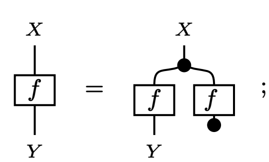


Figure 17: Characterisation of deterministic-domain morphisms.

Proof. Let us show the first implication: whenever the equation in Figure 17 holds, we can discard the outputs on both sides of the equation and conclude that the domain must be deterministic: $f = (f \mathbin{⨾} \varepsilon) \triangleleft f$ implies

$$(f \mathbin{⨾} \varepsilon) = (f \mathbin{⨾} \varepsilon) \triangleleft (f \mathbin{⨾} \varepsilon) = \nu \mathbin{⨾} ((f \mathbin{⨾} \varepsilon) \otimes (f \mathbin{⨾} \varepsilon)).$$

For the opposite implication, we employ the existence of the normalisation of f.

$$f = (f \mathbin{⨾} \varepsilon) \triangleleft \mathtt{n}(f) = (f \mathbin{⨾} \varepsilon) \triangleleft (f \mathbin{⨾} \varepsilon) \triangleleft \mathtt{n}(f) = (f \mathbin{⨾} \varepsilon) \triangleleft f.$$

We conclude that any deterministic-domain morphism is its own normalisation. □

Example 3.17. In the Kleisli category of the distribution monad, subStoch, a morphism $f \colon X \to Y$ has a deterministic domain if and only if it factors through the inclusion distributions and failure into subdistributions, $\mathbf{D}(X) + 1 \hookrightarrow \mathbf{D}_{\leq 1}(X)$.

Lemma 3.18 (Deterministic morphisms have deterministic domains)**.** *Deterministic morphisms in a copy-discard category have a deterministic domain.*

Proof. Using that the morphism is deterministic,

$$(f \mathbin{⨾} \varepsilon) = f \mathbin{⨾} \nu \mathbin{⨾} (\varepsilon \otimes \varepsilon) = \nu \mathbin{⨾} ((f \mathbin{⨾} \varepsilon) \otimes (f \mathbin{⨾} \varepsilon)),$$

which proves that its domain is deterministic. □

Lemma 3.19. *Let $f \colon X \to A \otimes B$ be a morphism in a partial Markov category that can be factored as $f = (m \triangleleft c)$ for any $m \colon X \to A$ and some deterministic-domain $c \colon A \otimes X \to B$; then $f = (f \mathbin{⨾} \pi_1) \triangleleft c$.*

Proof. Let us show the existence of conditionals, $(f \mathbin{⨾} \pi_1) \triangleleft c = f$. We employ *(i)* the decomposition of f, by assumption; *(ii)* associativity; *(iii)* that c has a deterministic domain, characterized in Lemma 3.16; and *(iv)* again the decomposition of f.

$$(f \mathbin{⨾} \pi_1) \triangleleft c \overset{(i)}{=} ((m \triangleleft c) \mathbin{⨾} \pi_1) \triangleleft c \overset{(ii)}{=} m \triangleleft (c \mathbin{⨾} \pi_1) \triangleleft c \overset{(iii)}{=} m \triangleleft c = f.$$ □

Considering again Proposition 2.17, one can conjecture that a similar procedure may exist in Kleisli categories of Maybe monads on other Markov categories. We show that this is indeed the case.

3.5. **Kleisli categories of maybe monads.** We prove in this section that the Kleisli categories of exception monads over Markov categories have conditionals; these conditionals are inherited from the base category. This is a recipe for constructing partial Markov categories from Markov categories with coproducts: any exception monad will induce a partial Markov category.

Definition 3.20 (Distributive Markov category, c.f. [AFK+24, LS24])**.** A *distributive Markov category* is a Markov category with coproducts that have deterministic injections, $i_1 \colon X \to X + Y$ and $i_2 \colon Y \to X + Y$, and such that the distributor, $d \colon X \otimes Z + Y \otimes Z \to (X + Y) \otimes Z$, is a deterministic isomorphism.[4]

In a distributive Markov category, an idempotent $\otimes$-monoid E induces a monoidal monad, $(\bullet + E)$, called its *exception monad.* The Kleisli category of this exception monad is again a copy-discard category; we prove that it is also a partial Markov category. For this proof, we use multiple results that we detail during the rest of the section; we also introduce the notion of a functor that preserves the copy morphism.

Theorem 3.21 (Partial Markov categories from Maybe monads)**.** *The Kleisli category of the maybe monad,* $(\bullet + 1)$*, over a distributive Markov category is a partial Markov category.*

Proof. From the deterministic inclusions of a distributive Markov category, we build a deterministic laxator for the maybe monad; the maybe monad is a copy functor (as we will introduce in Definition 3.22) with this monoidal structure

$$\mathfrak{l}_{X,Y} \colon (X + 1) \otimes (Y + 1) \to X \otimes Y + 1.$$

Applying Proposition 3.23, we obtain a section for the laxator, $\mathfrak{s}_{X,Y} \fatsemi \mathfrak{l}_{X,Y} = \mathrm{id}$.

Given any morphism in the Kleisli category of the maybe monad, $f \colon X \to T(Y \otimes Z)$, we can construct a conditional for $(f \fatsemi \mathfrak{s}_{Y,Z})$: there exists some $c \colon X \otimes T(Y) \to T(Z)$ such that

$$f \fatsemi \mathfrak{s}_{Y,Z} = (f \fatsemi \mathfrak{s}_{Y,Z} \fatsemi \pi_1) \triangleleft c.$$

Postcomposing with the laxator, we obtain the following equation; by , we know that the conditional can be replaced by the Kleisli extension of some $d \colon X \otimes Y \to T(Z)$

$$\begin{aligned} f &= ((f \fatsemi \mathfrak{s}_{Y,Z} \fatsemi \pi_1) \triangleleft c) \fatsemi \mathfrak{l}_{Y,Z} \\ &= ((f \fatsemi \mathfrak{s}_{Y,Z} \fatsemi \pi_1) \triangleleft d^*) \fatsemi \mathfrak{l}_{Y,Z}. \end{aligned}$$

Finally, by Lemma 3.24, this is the same as saying that $f = (f \fatsemi \overline{\pi}^*) \triangleleft^* d$; in other words, we have constructed a conditional in the Kleisli category. □

Definition 3.22 (Copy functor)**.** A *copy functor* is a lax monoidal functor F between copy-discard categories, with structure maps $\mathfrak{l}_{X,Y} \colon F(X) \otimes F(Y) \to F(X \otimes Y)$ and $\mathfrak{l}_0 \colon I \to F(I)$, that preserves the copy morphism, $F(\nu_X) = \nu_{FX} \fatsemi \mathfrak{l}_{X,X}$.

Proposition 3.23. *Any copy functor with a deterministic laxator,* $(F, \mathfrak{l})$*, is also an oplax copy functor,* $(F, \mathfrak{l}, \mathfrak{s})$*, with the oplaxator constructed from copying and discarding,*

$$\mathfrak{s} = \nu_{F(X \otimes Y)} \fatsemi (F(\pi_1) \otimes F(\pi_2)).$$

Moreover, the oplaxator splits the laxator, $\mathfrak{s} \fatsemi \mathfrak{l} = \mathrm{id}$.

[4] Note that we ask Markov categories to have conditionals: in the literature, distributive Markov categories can be found as "distributive Markov categories with conditionals" [AFK+24, LS24].

Proof. We write this proof equationally. Alternatively, one can employ *functor boxes* for lax monoidal functors [Mel06]. Let us first show that it splits the laxator. We reason by *(i)* naturality of the laxator, *(ii)* preservation of copy, *(iii)* functoriality, and *(iv)* the comonoid axioms.

$$
\begin{aligned}
&\nu_{F(X\otimes Y)} ⨾ (F(\pi_1)\otimes F(\pi_2)) ⨾ \wr_{X,Y} &&\overset{(i)}{=}\\
&\nu_{F(X\otimes Y)} ⨾ \wr_{X\otimes Y, X\otimes Y} ⨾ F(\pi_1\otimes\pi_2) &&\overset{(ii)}{=}\\
&F\nu_{X\otimes Y} ⨾ F(\pi_1\otimes\pi_2) &&\overset{(iii)}{=}\\
&F(\nu_{X\otimes Y} ⨾ (\pi_1\otimes\pi_2)) &&\overset{(iv)}{=}\\
&F(\mathrm{id}).
\end{aligned}
$$

Let us show that the oplaxator preserves copy. We reason by *(i)* preservation of copy, *(ii)* determinism of the laxator, *(iii)* associativity of the comonoid, *(iv)* preservation of copy (twice), and *(v)* unitality of the comonoid.

$$
\begin{aligned}
&F\nu_X ⨾ \nu_{F(X\otimes X)} ⨾ (F(\pi_1)\otimes F(\pi_2)) &&\overset{(i)}{=}\\
&\nu_{FX} ⨾ \wr_{X,X} ⨾ \nu_{F(X\otimes X)} ⨾ (F(\pi_1)\otimes F(\pi_2)) &&\overset{(ii)}{=}\\
&\nu_{FX} ⨾ \nu_{F(X)\otimes F(X)} ⨾ (\wr_{X,X}\otimes \wr_{X,X}) ⨾ (F(\pi_1)\otimes F(\pi_2)) &&\overset{(iii)}{=}\\
&\nu_{FX} ⨾ (\nu_{F(X)}\otimes\nu_{F(X)}) ⨾ (\wr_{X,X}\otimes \wr_{X,X}) ⨾ (F(\pi_1)\otimes F(\pi_2)) &&\overset{(iv)}{=}\\
&\nu_{FX} ⨾ (F(\nu_X)\otimes F(\nu_X)) ⨾ (F(\pi_1)\otimes F(\pi_2)) &&\overset{(v)}{=}\\
&\nu_{FX}.
\end{aligned}
$$

In categories of total maps, we moreover have a unit oplaxator, $\mathfrak{s}_0 \colon F(I)\to I$. □

The next lemma will rewrite the conditionals of the Kleisli category of a lax monoidal monad in terms of the conditionals of the base category and Kleisli extension. Given any strong monad, T, its strong Kleisli extension [MU22] is the operation lifting a morphism $f\colon X\otimes Y\to T(Z)$ to a morphism $f^*\colon X\otimes T(Y)\to T(Z)$.

Lemma 3.24. *Let $f\colon X\to T(Y)$ and $g\colon X\otimes Y\to T(Z)$ be morphisms in the Kleisli category of a copy monad over a total copy-discard category. Their conditional composition in the Kleisli category, $(\triangleleft^*)$, can be rewritten as*

$$(f \triangleleft^* g) = (f\triangleleft g^*) ⨾ \wr.$$

Lemma 3.25. *Let $g\colon X\otimes(Y+1)\to Z+1$ be a morphism in the Kleisli category of the maybe monad. There exists a morphism $h\colon X\otimes Y\to Z+1$ whose Kleisli extension replaces g in any conditional composition, by any $f\colon X\to Y+1$, followed by the laxator,*

$$(f\triangleleft g) ⨾ \wr = (f\triangleleft h^*) ⨾ \wr.$$

Proof. We may split the morphism into $g_1\colon X\otimes Y\to Z+1$ and $g_2\colon X\to Z+1$. □

4. Discrete Partial Markov Categories

We shall now introduce *discrete partial Markov categories*, a refinement of partial Markov categories that allows for the encoding of *constraints*.

Discrete cartesian restriction categories [CGH12] are a refinement of cartesian restriction categories that allows for the encoding of *constraints*: a map may fail if some conditions are not satisfied. Our observation is that a similar refinement can be applied to partial Markov categories to obtain discrete partial Markov categories. They provide a setting in which it is possible to (i) constrain the behaviour of a morphism, via Bayesian updates; and (ii) reason with these constrained stochastic maps.

The encoding of constraints requires the existence of *comparator maps* that interact nicely with the copy and discard (Figure 18, see also Remark 4.4). A *comparator* declares that some constraint—usually coming from an observation—must be satisfied in a probabilistic process.

Definition 4.1 (Comparator)**.** A copy-discard category $\mathbb{C}$ has *comparators* if every object X has a morphism, $\mu_X : X \otimes X \to X$, that is uniform, commutative, associative, satisfies the Frobenius axiom, and the special axiom. In other words, it is a partial Frobenius monoid [DLNS21], as in Figure 18.

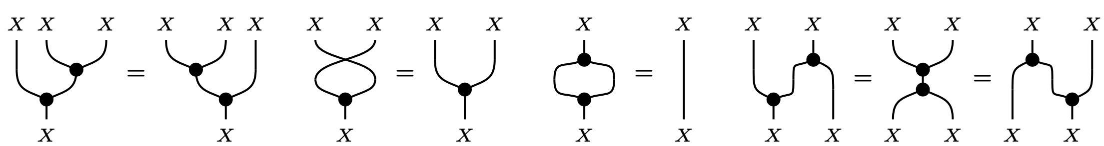


Figure 18: Axioms of a partial Frobenius monoid.

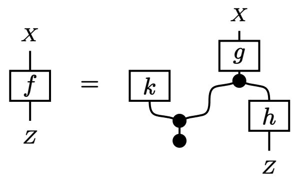
Definition 4.2 (Discrete partial Markov category)**.** A *discrete partial Markov category* is a partial Markov category with comparators. In other words, it is copy-discard category with conditionals and comparators.

Example 4.3 (SubStoch is a discrete partial Markov category)**.** The Kleisli category of the finitary subdistribution monad, subStoch, is a discrete partial Markov category. The comparator $\mu_X : X \otimes X \to X$ is given by

$$\mu_X(x \mid x_1, x_2) = \begin{cases} 1, & x = x_1 = x_2; \\ 0, & \text{otherwise.} \end{cases}$$

The comparator, copy, and discard morphisms of subStoch are lifted from the category of partial functions via the inclusion $\iota : \mathsf{Par} \hookrightarrow \mathsf{subStoch}$ given by post-composition with the unit of the finitary distribution monad. The comparator in Par satisfies the axioms in Figure 18 as copying a resource and then checking that the two copies coincide should be the identity process, and for checking equality of two resources and then copying them should be the same as copying one resource if it coincides with the other one. By functoriality of the inclusion $\iota : \mathsf{Par} \hookrightarrow \mathsf{subStoch}$, the comparator satisfies the same axioms in subStoch.

Remark 4.4. Thanks to the special Frobenius axioms (Figure 18), string diagrams in subStoch keep the same intuitive reading as in Stoch: the value of a morphism is obtained by multiplying the values of all its components and summing on the wires that are not inputs nor outputs. For example, the value of the morphism is the following formula.



$$f(z \mid x) = \sum_{y \in Y} g(y \mid x) \cdot h(z \mid y) \cdot k(y).$$

Example 4.5 (Naive comparators in continuous probability)**.** The category $\mathsf{BorelStoch}_{\leq 1}$ can be endowed with a naive comparator, $\mu_X : (X, \Sigma_X) \otimes (X, \Sigma_X) \to (X, \Sigma_X)$, that satisfies the special Frobenius axioms:

$$\mu_X(A, x, y) = \begin{cases} 1, & \text{if } x = y \in A; \\ 0, & \text{otherwise.} \end{cases}$$

This definition gives a measurable function $\mu_X(A \mid -, -) : X \times X \to [0, 1]$ if and only if the diagonal $\Delta_X = \{(x, x) : x \in X\}$ belongs to the product σ-algebra, $\Sigma_X \times \Sigma_X$, which holds for standard Borel spaces. However, this naive comparator does not behave as we would like it to: the set $\{x_0\}$ has measure 0, so comparing with x_0 yields the subdistribution with measure 0, which cannot be renormalised. We address this problem in the following sections.

4.1. **Bayes' Theorem.** Bayes' theorem prescribes how to update one's belief in light of new evidence. Classically, one observes evidence $y \in Y$ from a prior distribution σ on X through a channel $c : X \to Y$. The updated distribution is given by evaluating the Bayesian inversion of the channel c on the new observation y.

Theorem 4.6 (Bayes' Theorem)**.** *In a discrete partial Markov category, observing a deterministic $y : I \to Y$ from a prior distribution $p : I \to X$ through a channel $f : X \to Y$ is the same, up to scalar, as evaluating the Bayesian inversion on the observation, $f_\dagger(p)(y)$.*

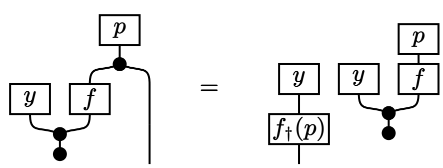


Figure 19: Bayes' theorem.

Proof. The equalities follow from: (*i*) the definition of Bayesian inversion (Definition 3.3), (*ii*) the partial Frobenius axioms (Figure 18), and (*iii*) the fact that y is deterministic. □

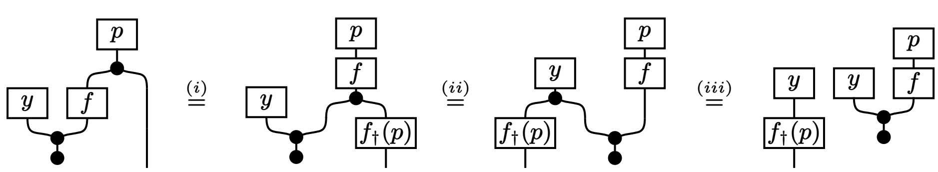


Figure 20: Proof of the Bayes' theorem.

4.2. **Pearl's and Jeffrey's updates.** The process for updating a belief on new evidence may depend on the type of evidence given. Pearl's [Pea88, Pea90] and Jeffrey's [Jef90, Sha81, Hal17] updates are two possibilities for performing an update of a belief in light of a new piece of evidence that is not necessarily a single deterministic observation [Jac19]. Updating a prior belief according to Pearl's rule increases *validity*, i.e. the probability of the new evidence being true according to our belief [CJWW15]. On the other hand, updating with Jeffrey's rule reduces "how far" the new evidence is from our prediction, i.e. it decreases *Kullback-Leibler divergence* [Jac19, Jac21].

The difference between these two update rules comes from the fact that they are based on different types of evidence. Pearl's evidence comes as a probabilistic predicate, i.e. a morphism $q \colon Y \to I$ in a discrete partial Markov category. Pearl's update coincides with the update prescribed by Bayes' theorem.

Definition 4.7 (Pearl's update)**.** Let $p \colon I \to X$ be a prior distribution and $q \colon Y \to I$ be a predicate in a discrete partial Markov category $\mathbb{C}$, which is observed through a channel $f \colon X \to Y$. Pearl's updated prior is defined to be $p \triangleleft (f \mathbin{⨟} q)$ or, equivalently, $(f \mathbin{⨟} q)_{\dagger}(p)$, the Bayesian inversion of $f \mathbin{⨟} q$ with respect to p.

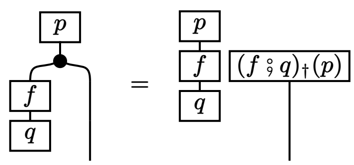
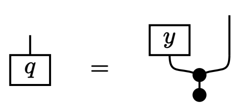


Figure 21: Pearl's update.

Jeffrey's evidence, on the other hand, is given by a distribution on Y.

Definition 4.8 (Jeffrey's update)**.** Let $t \colon I \to Y$ be a state in $\mathbb{C}$. *Jeffrey's updated prior* is $t \mathbin{⨟} f_{\dagger}(p)$, the composition of the evidence with a Bayesian inversion of f with respect to p.

When Pearl's evidence predicate q is deterministic—that is, its probability mass is concentrated in just one point $y \in Y$—then it becomes a comparison. In this case, there



is no difference between the two update rules. This result was shown by Jacobs [Jac19, Proposition 5.3] in the case of the Kleisli category of finitary distribution monad: let us prove it in any discrete partial Markov category.

Proposition 4.9. *Pearl's and Jeffrey's updates coincide for deterministic observations. Let $y \colon I \to Y$ be deterministic, then Pearl's update on the predicate $q \colon Y \to I$, as defined in Section 4.2, is Jeffrey's update on y.*

Proof. The result follows from exactly the same string diagrammatic reasoning as Bayes' theorem (Theorem 4.6). ∎

5. Exact Observations

Bayesian inference, in discrete partial Markov categories, relies on the existence of comparators. However, many partial Markov categories may not have this structure or, even worse, its comparator structure might not behave as expected, providing a poor semantics of Bayesian inference (see Example 4.5). The solution to this problem comes from noticing that, in most inference problems, we only care about *exact observations*.

5.1. **Exact observations.** We define the category $\mathsf{exact}(\mathbb{C})$ of *exact observations* over a copy-discard category, $\mathbb{C}$, by syntactically adding observations of deterministic evidence. A different but similarly behaved construction has appeared in the work of Stein and Staton [SS21, Ste21]. We show that, when $\mathbb{C}$ is a Markov category, processes with exact observations form a partial Markov category; conditionals and normalisations for them can be computed with the conditionals from the original Markov category. A normal form theorem will enable this computation.

Definition 5.1 (Exact observations)**.** The category of processes with exact observations, $\mathsf{exact}(\mathbb{C})$, over a copy-discard category $\mathbb{C}$, is obtained by freely adding a generator $x^\circ \colon X \to I$ for every deterministic morphism without inputs, $X \colon I \to Y$, and quotienting by compatibility with the tensor, $(x \otimes y)^\circ = x^\circ \otimes y^\circ$ and $\mathrm{id}_I^\circ = \mathrm{id}_I$, and the observation axiom $x^\circ \triangleleft \mathrm{id} = x^\circ \fatsemi x$ (in Figure 22).

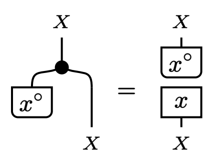


Figure 22: Axiom for the category of exact observations.

We interpret the generator $x^\circ \colon X \to I$ as the observation of the corresponding deterministic evidence $x \colon I \to X$. Under this interpretation, the following equivalent formlation of the axiom explains that observing x and computing some f is the same as observing x and passing it to the input of that computation,

$$x^\circ \triangleleft f = x^\circ \fatsemi x \fatsemi f.$$

Proposition 5.2. *The category of exact observations,* $\mathsf{exact}(\mathbb{C})$*, embeds faithfully in* $\mathsf{partial}(\mathbb{C})$*, the free discrete copy-discard category over* $\mathbb{C}$*.*

Proof sketch. We define an identity-on-objects functor $J \colon \mathsf{exact}(\mathbb{C}) \to \mathsf{partial}(\mathbb{C})$. Every morphism f in $\mathsf{exact}(\mathbb{C})$ that comes from a morphism in $\mathbb{C}$ is left unchanged: $J(f) = f$. In other words, the functor J commutes with the inclusions of $\mathbb{C}$ into $\mathsf{exact}(\mathbb{C})$ and $\mathsf{partial}(\mathbb{C})$.

For every deterministic morphism $x \colon 1 \to X$, the image of its corresponding observation is defined as $J(x^\circ) = (x \otimes \mathrm{id}) \fatsemi \mu_X \fatsemi \varepsilon_X$ (i.e., the right hand side of Figure 22). The fact that J is well defined—and its faithfulness—follow from string diagrammatic reasoning [DR23]. □

Exact observations on a Markov category give a syntax for stochastic processes with some observations of deterministic evidence. In principle, it is not clear how to compute the semantics of these exact observations and, in particular, how to compute conditionals of them. We show that we can give semantics to exact observations in the original Markov category by computing their normalisations.

A consequence of this result is that conditionals of processes with exact observations can be computed by conditionals in the original Markov category.

Theorem 5.3. *Any morphism in the category of processes with exact observations,* $\mathsf{exact}(\mathbb{C})$, *can be written in normal form as* $f = (h \fatsemi z^\circ) \triangleleft g$, *where* g *and* h *are total, and where* z *is total and deterministic. Moreover,* g *is, almost surely, the normalisation of* f.

Proof sketch. The proof proceeds by structural induction on f, we detail the non-trivial cases. Given two processes with exact observations in normal form, $f_1 = (h_1 \fatsemi z_1^\circ) \triangleleft g_1$ and $f_2 = (h_2 \fatsemi z_2^\circ) \triangleleft g_2$, its tensor is again in normal form,

$$f_1 \otimes f_2 = ((h_1 \fatsemi z_1^\circ) \triangleleft g_1) \otimes ((h_2 \fatsemi z_2^\circ) \triangleleft g_2) = ((h_1 \otimes h_2) \fatsemi (z_1 \otimes z_2)^\circ) \triangleleft (g_1 \otimes g_2).$$

Given two composable processes with exact observations in normal form, $f_1 = (h_1 \fatsemi z_1^\circ) \triangleleft g_1$ and $f_2 = (h_2 \fatsemi z_2^\circ) \triangleleft g_2$, its composition can be written in normal form as follows, using Bayesian inversions,

$$f_1 \fatsemi f_2 = (h_1 \otimes (g_1 \fatsemi h_2) \fatsemi (z_1 \otimes z_2)^\circ) \triangleleft ((\mathrm{id} \otimes z_2) \fatsemi h_{2\dagger}(g_1) \fatsemi g_2);$$

we prove this by string diagrammatic reasoning (Figure 23).

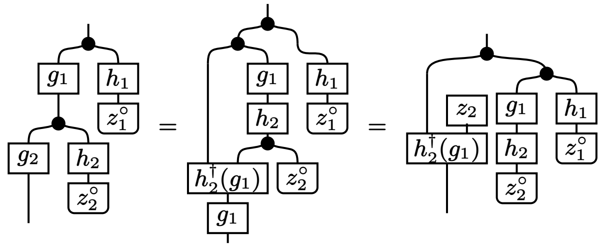


Figure 23: Composition of processes with exact observations in normal form.

Finally, let us show that $f = (h \fatsemi z^\circ) \triangleleft g$—with g a total morphism—implies that g is, almost surely, the normalisation of f. We use *(i)* the assumption, *(ii)* totality of g, and *(iii)* the assumption.

$$f \overset{(i)}{=} (h \fatsemi z^\circ) \triangleleft g \overset{(ii)}{=} (h \fatsemi z^\circ) \triangleleft (g \fatsemi \varepsilon) \triangleleft g \overset{(iii)}{=} (f \fatsemi \varepsilon) \triangleleft g.$$

□

Theorem 5.4. *The category of exact observations over a Markov category,* $\mathsf{exact}(\mathbb{C})$, *is a partial Markov category. Its conditionals can be computed from the conditionals of the base Markov category.*

Proof. By Theorem 5.3, we can compute a normalisation $\mathtt{n}(f)$ of a process with exact observations f in $\mathbb{C}$ by taking conditionals in $\mathbb{C}$. By Proposition 3.13, a conditional of $\mathtt{n}(f)$ is a conditional of f and it can be computed by only taking conditionals in $\mathbb{C}$. □

5.2. **Example — Inferring the mean of a normal distribution.** Assume a normal distribution, $\mathrm{Norm}(m, 1)$, with standard deviation of 1 and with its mean sampled uniformly from the interval $m \sim \mathrm{unif}(0, 1)$. We sample from this normal distribution, $v \sim \mathrm{Norm}(m, 1)$, and observe a value of, say, $v = 2.1$. What is updated posterior distribution on m?

The reasoning from the operational description of the problem to a computational description is in . It rewrites a diagram containing exact observations into a diagram in the normal form of Theorem 5.3.

We now use the explicit density functions for the uniform and normal distributions to construct two morphisms, $\mathrm{unif}\colon I \to \mathbb{R}$ and $\mathrm{norm}\colon \mathbb{R} \to \mathbb{R}$, of the category of standard Borel spaces, BorelStoch.

$$\mathrm{unif}^{pdf}_{[0,1]}(x) = \delta_{[0,1]}(x); \qquad \mathrm{norm}^{pdf}(x, m) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(x-m)^2\right).$$

Because these are constructed from density functions, the Bayesian inversion of the normal distribution has a known density function (Example 3.7).

$$\mathrm{norm}_{\dagger}(\mathrm{unif}_{[0,1]})^{pdf}(m|x) = \frac{\mathrm{unif}^{pdf}_{[0,1]}(x) \cdot \mathrm{norm}^{pdf}(x, m)}{\int_{x_0 \in X} \mathrm{unif}^{pdf}_{[0,1]}(x_0) \cdot \mathrm{norm}^{pdf}(x_0, m)\, dx_0}.$$

Now, string diagrammatic reasoning in the category of processes with exact observations does evaluate the observation on the Bayesian inversion (Figure 24).

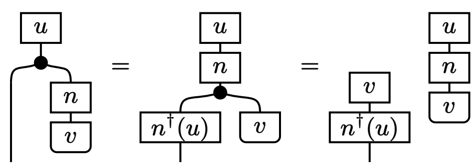
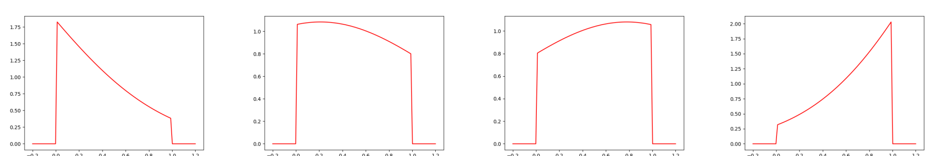


Figure 24: Bayesian inversion via exact observations.

Let us plot the density function of the posterior distribution, $\mathrm{norm}_{\dagger}(\mathrm{unif}_{[0,1]})^{pdf}(m|x)$, for multiple values of v. We do this by numerically evaluating the integral expression we obtain from the string diagrams.



(a) (b) (c) (d)

Figure 25: Graphs for the posterior density functions after observing (a) $v = -1.1$, (b) $v = 0.21$, (c) $v = 0.78$, and (d) $v = 2.4$.

In this way, partial Markov categories of exact observations may seen as providing further algebraic justification for the compositional approach to exact conditioning [SS21, Section IV]. Finally, note that computing exact observations is also not restricted to Gaussian probability theory (Theorem 5.3); we can handle computations with integrals symbolically, even if we choose to compute them numerically for Figure 25.

Acknowledgements

This publication is based upon work from the COST Action EuroProofNet, supported by COST (European Cooperation in Science and Technology, www.cost.eu). We thank multiple helpful discussions and feedback from Bart Jacobs, Dario Stein, Sam Staton, Paolo Perrone, David Spivak, and Márk Széles.

References

[AFK+24] Nathanael L. Ackerman, Cameron E. Freer, Younesse Kaddar, Jacek Karwowski, Sean K. Moss, Daniel M. Roy, Sam Staton, and Hongseok Yang. Probabilistic Programming Interfaces for Random Graphs: Markov Categories, Graphons, and Nominal Sets. *Proceedings of the ACM on Programming Languages*, 8(POPL):1819–1849, 2024. doi:10.1145/3632903.

[Ahm14] Arif Ahmed. *Evidence, Decision and Causality.* Cambridge University Press, 2014.

[Bén67] Jean Bénabou. Introduction to bicategories. In *Reports of the Midwest Category Seminar*, pages 1–77, Berlin, Heidelberg, 1967. Springer Berlin Heidelberg. doi:10.1007/BFb0074299.

[Car87] Aurelio Carboni. Bicategories of partial maps. *Cahiers de topologie et géométrie différentielle catégoriques*, 28(2):111–126, 1987.

[CG99] Andrea Corradini and Fabio Gadducci. An Algebraic Presentation of Term Graphs, via GS-Monoidal Categories. *Applied Categorical Structures*, 7(4):299–331, 1999. doi:10.1023/A:1008647417502.

[CGH12] Robin Cockett, Xiuzhan Guo, and Pieter Hofstra. Range Categories II: Towards regularity. *Theory and Applications of Categories*, 26(18):453–500, 2012.

[CJ19] Kenta Cho and Bart Jacobs. Disintegration and Bayesian Inversion via String Diagrams. *Mathematical Structures in Computer Science*, pages 1–34, March 2019. arXiv:1709.00322, doi:10.1017/S0960129518000488.

[CJWW15] Kenta Cho, Bart Jacobs, Bas Westerbaan, and Abraham Westerbaan. An introduction to effectus theory. *CoRR*, abs/1512.05813, 2015. URL: http://arxiv.org/abs/1512.05813, arXiv:1512.05813.

[CL02] Robin Cockett and Stephen Lack. Restriction categories I: categories of partial maps. *Theoretical Computer Science*, 270(1-2):223–259, 2002. doi:10.1016/S0304-3975(00)00382-0.

[CL03] Robin Cockett and Stephen Lack. Restriction categories ii: partial map classification. *Theoretical Computer Science*, 294(1-2):61–102, 2003.

[CL07] Robin Cockett and Stephen Lack. Restriction categories III: colimits, partial limits and extensivity. *Mathematical Structures in Computer Science*, 17(4):775–817, 2007.

[CO89] Pierre-Louis Curien and Adam Obtułowicz. Partiality, cartesian closedness and toposes. *Information and Computation*, 80(1):50–95, 1989. doi:10.1016/0890-5401(89)90023-0.

[CS12] Bob Coecke and Robert W Spekkens. Picturing classical and quantum bayesian inference. *Synthese*, 186:651–696, 2012.

[DdFR22] Elena Di Lavore, Giovanni de Felice, and Mario Román. Monoidal streams for dataflow programming. In *Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '22, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3531130.3533365.

[DK19] Fredrik Dahlqvist and Dexter Kozen. Semantics of higher-order probabilistic programs with conditioning. *Proc. ACM Program. Lang.*, 4(POPL), dec 2019. doi:10.1145/3371125.

[DLNS21] Ivan Di Liberti, Fosco Loregian, Chad Nester, and Pawel Sobocinski. Functorial semantics for partial theories. *Proc. ACM Program. Lang.*, 5(POPL):1–28, 2021. doi:10.1145/3434338.

[DPH87] Robert Di Paola and Alex Heller. Dominical Categories: Recursion Theory Without Elements. *The Journal of symbolic logic*, 52(3):594–635, 1987.

[DR23] Elena Di Lavore and Mario Román. Evidential decision theory via partial markov categories. In *38th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2023, Boston, MA, USA, June 26-29, 2023*, pages 1–14. IEEE, 2023. doi:10.1109/LICS56636.2023.10175776.

[EPT17] Thomas Ehrhard, Michele Pagani, and Christine Tasson. Measurable cones and stable, measurable functions: a model for probabilistic higher-order programming. *Proceedings of the ACM on Programming Languages*, 2(POPL):1–28, 2017.

[FGL+23] Tobias Fritz, Tomáš Gonda, Antonio Lorenzin, Paolo Perrone, and Dario Stein. Absolute continuity, supports and idempotent splitting in categorical probability. *CoRR*, abs/2308.00651, 2023. URL: https://doi.org/10.48550/arXiv.2308.00651, arXiv:2308.00651, doi:10.48550/ARXIV.2308.00651.

[FGP21] Tobias Fritz, Tomáš Gonda, and Paolo Perrone. De Finetti's theorem in categorical probability. *Journal of Stochastic Analysis*, 2(4), 2021. doi:10.31390.josa.2.4.06.

[FL22] Tobias Fritz and Wendong Liang. Free GS-monoidal categories and free Markov categories. *CoRR*, abs/2204.02284, 2022. arXiv:2204.02284, doi:10.48550/arXiv.2204.02284.

[Fon13] Brendan Fong. Causal Theories: A Categorical Perspective on Bayesian Networks. *Master's Thesis, University of Oxford*, 2013. doi:10.48550/arXiv.1301.6201.

[FP19] Tobias Fritz and Paolo Perrone. A probability monad as the colimit of spaces of finite samples. *Theory and Applications of Categories*, 34(7):170–220, 2019. URL: https://arxiv.org/abs/1712.05363.

[FPR21] Tobias Fritz, Paolo Perrone, and Sharwin Rezagholi. Probability, valuations, hyperspace: Three monads on top and the support as a morphism. *Mathematical Structures in Computer Science*, 31(8):850–897, 2021. doi:10.1017/S0960129521000414.

[FR20] Tobias Fritz and Eigil Fjeldgren Rischel. Infinite products and zero-one laws in categorical probability. *Compositionality*, 2:3, 2020.

[Fri20] Tobias Fritz. A synthetic approach to Markov kernels, conditional independence and theorems on sufficient statistics. *Advances in Mathematics*, 370:107239, 2020. URL: http://arxiv.org/abs/1908.07021, arXiv:1908.07021.

[Gir82] Michèle Giry. A categorical approach to probability theory. In *Categorical aspects of topology and analysis*, pages 68–85. Springer, 1982.

[Gre13] Hilary Greaves. Epistemic decision theory. *Mind*, 122(488):915–952, 2013.

[GW06] Hilary Greaves and David Wallace. Justifying conditionalization: Conditionalization maximizes expected epistemic utility. *Mind*, 115(459):607–632, 2006.

[Hal17] Joseph Y Halpern. *Reasoning about uncertainty*. MIT press, 2017.

[Has97] Masahito Hasegawa. *Models of sharing graphs: a categorical semantics of let and letrec*. PhD thesis, University of Edinburgh, UK, 1997. URL: http://hdl.handle.net/1842/15001.

[HJW$^+$92] Paul Hudak, Simon L. Peyton Jones, Philip Wadler, Brian Boutel, Jon Fairbairn, Joseph H. Fasel, María M. Guzmán, Kevin Hammond, John Hughes, Thomas Johnsson, Richard B. Kieburtz, Rishiyur S. Nikhil, Will Partain, and John Peterson. Report on the Programming Language Haskell, A Non-strict, Purely Functional Language. *ACM SIGPLAN Notices*, 27(5):1, 1992. doi:10.1145/130697.130699.

[HKSY17] Chris Heunen, Ohad Kammar, Sam Staton, and Hongseok Yang. A convenient category for higher-order probability theory. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12. IEEE, 2017.

[HV19] Chris Heunen and Jamie Vicary. *Categories for Quantum Theory: An Introduction*. Oxford University Press, 2019. doi:10.1093/oso/9780198739623.001.0001.

[Jac18] Bart Jacobs. From probability monads to commutative effectuses. *Journal of logical and algebraic methods in programming*, 94:200–237, 2018.

[Jac19] Bart Jacobs. The mathematics of changing one's mind, via jeffrey's or via pearl's update rule. *J. Artif. Intell. Res.*, 65:783–806, 2019. doi:10.1613/jair.1.11349.

[Jac21] Bart Jacobs. Learning from what's right and learning from what's wrong. *arXiv preprint arXiv:2112.14045*, 2021.

[Jef90] Richard C Jeffrey. *The logic of decision*. University of Chicago press, 1990.

[Jef97] Alan Jeffrey. Premonoidal categories and a graphical view of programs. *Preprint at ResearchGate*, 1997. URL: https://www.researchgate.net/profile/Alan-Jeffrey/publication/228639836_Premonoidal_categories_and_a_graphical_view_of_programs/links/00b495182cd648a874000000/Premonoidal-categories-and-a-graphical-view-of-programs.pdf.

[JKZ21] Bart Jacobs, Aleks Kissinger, and Fabio Zanasi. Causal inference via string diagram surgery: A diagrammatic approach to interventions and counterfactuals. *Mathematical Structures in Computer Science*, 31(5):553–574, 2021.

[Joy99] James M Joyce. *The foundations of causal decision theory*. Cambridge University Press, 1999.

[JS91] André Joyal and Ross Street. The geometry of tensor calculus, i. *Advances in Mathematics*, 88(1):55–112, 1991. URL: https://www.sciencedirect.com/science/article/pii/000187089190003P, doi:10.1016/0001-8708(91)90003-P.

[JZ20] Bart Jacobs and Fabio Zanasi. The logical essentials of Bayesian reasoning. *Foundations of Probabilistic Programming*, pages 295–331, 2020.

[Lew81] David Lewis. Causal decision theory. *Australasian journal of philosophy*, 59(1):5–30, 1981.

[LJR24] Elena Di Lavore, Bart Jacobs, and Mario Román. A simple formal language for probabilistic decision problems, 2024. URL: https://arxiv.org/abs/2410.10643, arXiv:2410.10643.

[LS24] Jack Liell-Cock and Sam Staton. Compositional imprecise probability. *CoRR*, abs/2405.09391, 2024. arXiv:2405.09391, doi:10.48550/arXiv.2405.09391.

[Mel06] Paul-André Melliès. Functorial boxes in string diagrams. In Zoltán Ésik, editor, *Computer Science Logic, 20th International Workshop, CSL 2006, 15th Annual Conference of the EACSL, Szeged, Hungary, September 25-29, 2006, Proceedings*, volume 4207 of *Lecture Notes in Computer Science*, pages 1–30. Springer, 2006. doi:10.1007/11874683_1.

[MU22] Dylan McDermott and Tarmo Uustalu. What makes a strong monad? In Jeremy Gibbons and Max S. New, editors, *Proceedings Ninth Workshop on Mathematically Structured Functional Programming, MSFP@ETAPS 2022, Munich, Germany, 2nd April 2022*, volume 360 of *EPTCS*, pages 113–133, 2022. doi:10.4204/EPTCS.360.6.

[Pan99] Prakash Panangaden. The Category of Markov Kernels. *Electronic Notes in Theoretical Computer Science*, 22:171–187, January 1999. doi:10.1016/S1571-0661(05)80602-4.

[Pan09] Prakash Panangaden. *Labelled Markov Processes*. Imperial College Press, 2009. URL: http://www.worldscibooks.com/compsci/p595.html.

[Pat01] Ross Paterson. A new notation for arrows. In Benjamin C. Pierce, editor, *Proceedings of the Sixth ACM SIGPLAN International Conference on Functional Programming (ICFP '01), Firenze (Florence), Italy, September 3-5, 2001*, pages 229–240. ACM, 2001. doi:10.1145/507635.507664.

[Pea88] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

[Pea90] Judea Pearl. Jeffrey's rule, passage of experience, and neo-bayesianism. In *Knowledge representation and defeasible reasoning*, pages 245–265. Springer, 1990.

[RR88] Edmund Robinson and Giuseppe Rosolini. Categories of partial maps. *Information and computation*, 79(2):95–130, 1988.

[Sha81] Glenn Shafer. Jeffrey's rule of conditioning. *Philosophy of Science*, 48(3):337–362, 1981.

[SS21] Dario Stein and Sam Staton. Compositional semantics for probabilistic programs with exact conditioning. In *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13. IEEE, 2021.

[Ste21] Dario Stein. Structural foundations for probabilistic programming languages. *University of Oxford*, 2021.

[SV13] Mike Stay and Jamie Vicary. Bicategorical semantics for nondeterministic computation. *Electronic Notes in Theoretical Computer Science*, 298:367–382, 2013.

[SWY+16] Sam Staton, Frank Wood, Hongseok Yang, Chris Heunen, and Ohad Kammar. Semantics for probabilistic programming: higher-order functions, continuous distributions, and soft constraints. In *2016 31st annual ACM/IEEE Symposium on Logic in Computer Science (LiCS)*, pages 1–10. IEEE, 2016.

[VKS19] Matthijs Vákár, Ohad Kammar, and Sam Staton. A domain theory for statistical probabilistic programming. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–29, 2019.

[YS17] Eliezer Yudkowsky and Nate Soares. Functional Decision Theory: a New Theory of Instrumental Rationality. *ArXiv preprint arXiv:1710.05060*, 2017.