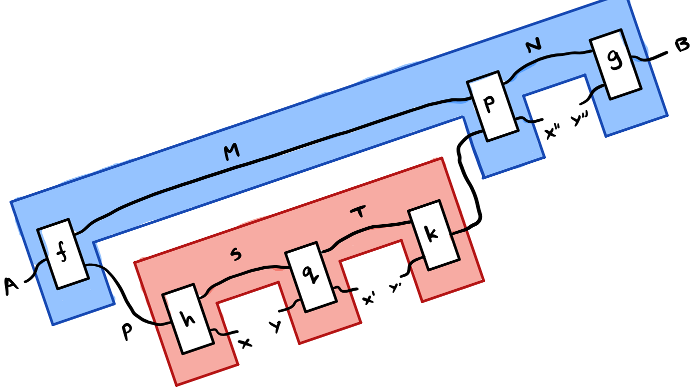
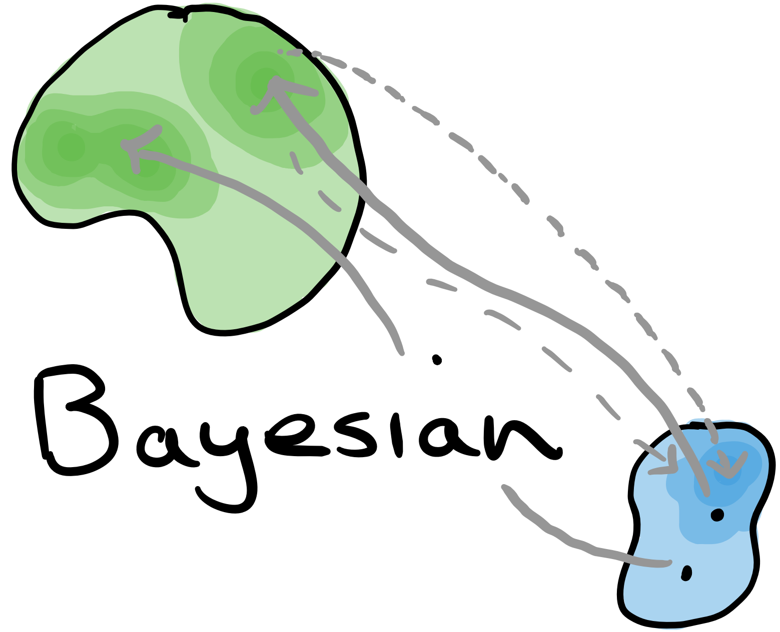




Compositional Bayesian Inference and Stochastic Combs

Dylan Braithwaite

University of Strathclyde

Computing and Information Science

Bayesian Inversion

Bayesian Inversion

Bayes law: Invert a conditional distribution to update a prior belief.

Bayesian Inversion

Bayes law: Invert a conditional distribution to update a prior belief.

Conditional distribution is "inverted"

prior distribution

$$P(X \mid Y=y) = \frac{P(Y=y \mid X)\, P(X)}{P(Y=y)}$$

Bayesian Inversion

Bayes law: Invert a conditional distribution to update a prior belief.

Conditional distribution is "inverted"

prior distribution

$$P(X \mid Y=y) = \frac{P(Y=y \mid X)\, P(X)}{P(Y=y)}$$

```
def example():
   let x := sample(Bernoulli(2/7))
       r := if x then 3 else 10
   in observe 4 from Poisson(r);
      return x
```

Based on an example from 'Probabilistic Programs as Measures', Staton (2020).

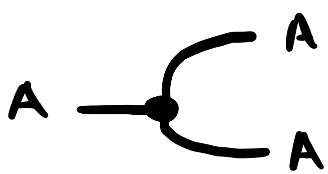
Bayesian Inversion, Sythetically

Markov categories offer a setting for doing synthetic probability theory.

Bayesian Inversion, Sythetically

Markov categories offer a setting for doing synthetic probability theory.

A monoidal category $\mathcal{C}$ with compatible families of copy and delete morphisms:

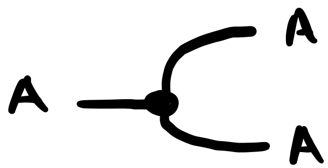


Fritz
(Adv. Math 370, 2020)

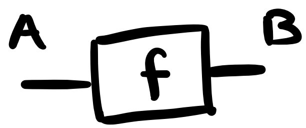
Bayesian Inversion, Sythetically

Markov categories offer a setting for doing synthetic probability theory.

A monoidal category $\mathcal{C}$ with compatible families of copy and delete morphisms:



Fritz
(Adv. Math 370, 2020)

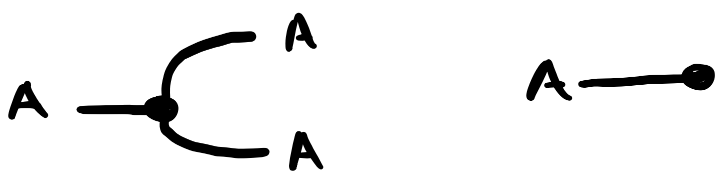
Morphisms 

are interpreted as stochastic functions: $a \longmapsto p_B$

Bayesian Inversion, Sythetically

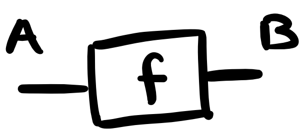
Markov categories offer a setting for doing synthetic probability theory.

A monoidal category $\mathcal{C}$ with compatible famillies of copy and delete morphisms:



Fritz
(Adv. Math 370, 2020)

Morphisms

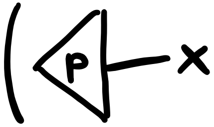


are interpreted as stochastic functions: $a \longmapsto p_B$

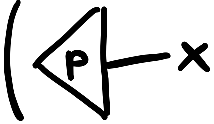
"probability that a is mapped to b"

Bayesian Inversion, Sythetically

Bayesian Inversion, Sythetically

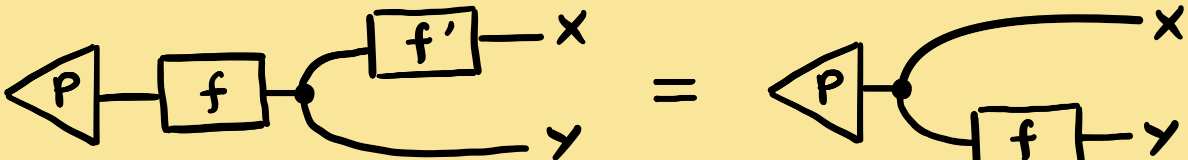
Internally, a distribution on X is a state $p : I \longrightarrow X$, () .

Bayesian Inversion, Sythetically

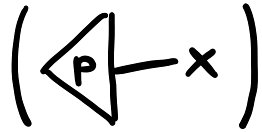
Internally, a distribution on X is a state $p: I \longrightarrow X$, ().

Defn (Cho & Jacobs, 2019)

A Bayesian inverse to $f: X \longrightarrow Y$ at p is a morphism $f': Y \longrightarrow X$ satisfying

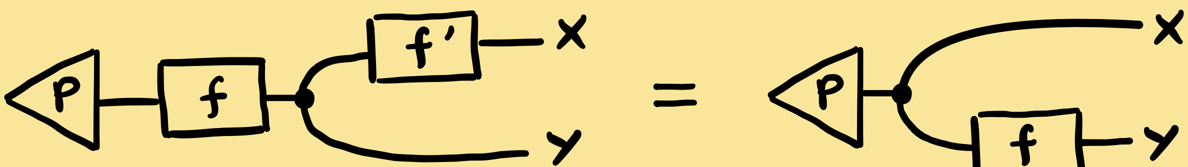


Bayesian Inversion, Sythetically

Internally, a distribution on X is a state $p : I \longrightarrow X$, () .

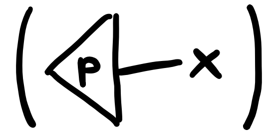
Defn (Cho & Jacobs, 2019)

A Bayesian inverse to $f : X \longrightarrow Y$ at p is a morphism $f' : Y \longrightarrow X$ satisfying



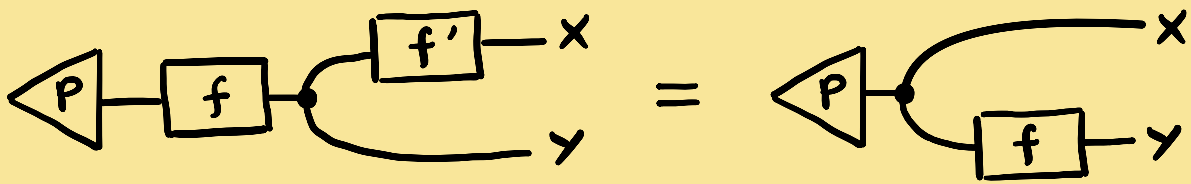
- In general, non-unique

Bayesian Inversion, Sythetically

Internally, a distribution on X is a state $p : I \longrightarrow X$, .

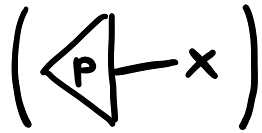
Defn (Cho & Jacobs, 2019)

A Bayesian inverse to $f : X \longrightarrow Y$ at p is a morphism $f' : Y \longrightarrow X$ satisfying



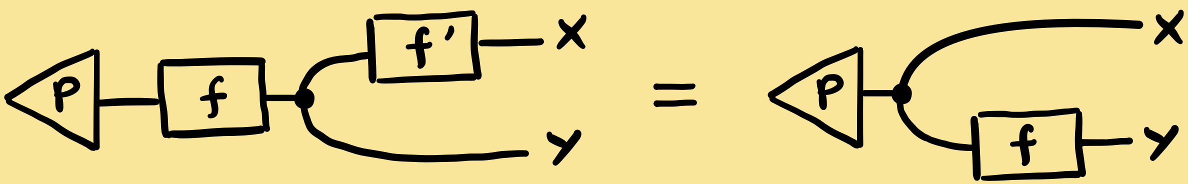
- In general, non-unique
- Depends on p

Bayesian Inversion, Sythetically

Internally, a distribution on X is a state $p : I \longrightarrow X$, () .

Defn (Cho & Jacobs, 2019)

A Bayesian inverse to $f : X \longrightarrow Y$ at p is a morphism $f' : Y \longrightarrow X$ satisfying



- In general, non-unique
- Depends on p
- Not guaranteed to exist (but always does in many examples).

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on ProbStoch($\mathcal{C}$)

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on $\mathrm{ProbStoch}(\mathcal{C})$

  objects paired with distributions

  hom-sets quotiented by a.s.-equality.

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on ProbStoch($\mathcal{C}$)
  - objects paired with distributions
  - hom-sets quotiented by a.s.-equality.
- (Staton, Stein, 2021) Compositional semantics for probabilistic programs with exact conditioning

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on ProbStoch($\mathcal{C}$)
  objects paired with distributions
  hom-sets quotiented by a.s.-equality.
- (Staton, Stein, 2021) Compositional semantics for probabilistic programs with exact conditioning
  - Encodes Bayesian updating well but cannot talk about general Bayesian inverses

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on $\mathrm{ProbStoch}(\mathcal{C})$
  objects paired with distributions
  hom-sets quotiented by a.s.-equality.
- (Staton, Stein, 2021) Compositional semantics for probabilistic programs with exact conditioning
  - Encodes Bayesian updating well but cannot talk about general Bayesian inverses
- (Smithe 2020) Bayesian inversion is almost surely a functor $\mathcal{C}^{op} \longrightarrow \mathrm{BLens}(\mathcal{C})$

Bayesian Inversion, Functorially

- (Fritz, 2020) Bayesian inversion is a dagger functor on $\mathbf{ProbStoch}(\mathcal{C})$
  objects paired with distributions
  hom-sets quotiented by a.s.-equality.
- (Staton, Stein, 2021) Compositional semantics for probabilistic programs with exact conditioning
  - Encodes Bayesian updating well but cannot talk about general Bayesian inverses
- (Smithe 2020) Bayesian inversion is almost surely a functor $\mathcal{C}^{op} \longrightarrow \mathbf{BLens}(\mathcal{C})$
  morphisms are $\mathcal{C}(I,X)$-indexed families of kernels.

Bayesian Inversion, Functoriality

- (Fritz, 2020) Bayesian inversion is a dagger functor on $\mathrm{ProbStoch}(\mathcal{C})$

  objects paired with distributions
  hom-sets quotiented by a.s.-equality.

- (Staton, Stein, 2021) Compositional semantics for probabilistic programs with exact conditioning
  - Encodes Bayesian updating well but cannot talk about general Bayesian inverses

- (Smithe 2020) Bayesian inversion is almost surely a functor $\mathcal{C}^{op} \longrightarrow \mathrm{BLens}(\mathcal{C})$

  morphisms are $\mathcal{C}(I,X)$-indexed families of kernels.

- (Braithwaite, Smithe, Hedges, 2023) Fix functoriality using families of support objects.

Repeated Updating

Repeated Updating

Morphisms of $\mathbf{BLens}(\mathcal{C})$ are pairs $\begin{pmatrix} A \\ X \end{pmatrix} \underset{\longleftarrow}{\overset{f}{\longrightarrow}} \begin{pmatrix} B \\ Y \end{pmatrix}$

Repeated Updating

Morphisms of $\mathbf{BLens}(\mathcal{C})$ are pairs $\begin{pmatrix}A\\X\end{pmatrix} \overset{f}{\rightleftarrows} \begin{pmatrix}B\\Y\end{pmatrix}$.

family of inverse kernels.

Repeated Updating

Morphisms of BLens($\mathcal{C}$) are pairs $\begin{pmatrix} A \\ X \end{pmatrix} \underset{}{\overset{f}{\rightleftarrows}} \begin{pmatrix} B \\ Y \end{pmatrix}$

family of inverse kernels.

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in  observe 4 from Poisson(r);
        observe 5 from Poisson(r);
        return x
```

Repeated Updating

Morphisms of BLens($\mathcal{C}$) are pairs $\begin{pmatrix} A \\ X \end{pmatrix} \underset{\leftarrow}{\xrightarrow{f}} \begin{pmatrix} B \\ Y \end{pmatrix}$

family of inverse kernels.

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in  observe 4 from Poisson(r);
        observe 5 from Poisson(r);
        return x
```

Modified example:

multiple observations

Repeated Updating

Morphisms of BLens(𝒞) are pairs $\begin{pmatrix} A \\ X \end{pmatrix} \underset{}{\overset{f}{\rightleftarrows}} \begin{pmatrix} B \\ Y \end{pmatrix}$

family of inverse kernels.

```
def example():
  let x := sample(Bernoulli(2/7))
      r := if x then 3 else 10
  in  observe 4 from Poisson(r);
      observe 5 from Poisson(r);
      return x
```

Modified example: multiple observations

$\Rightarrow$ Use output of update map as new prior

Repeated Updating

Morphisms of BLens(𝒞) are pairs $\begin{pmatrix}A\\X\end{pmatrix} \overset{f}{\rightleftarrows} \begin{pmatrix}B\\Y\end{pmatrix}$

family of inverse kernels.

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in  observe 4 from Poisson(r);
        observe 5 from Poisson(r);
        return x
```

Modified example: multiple observations

$\Rightarrow$ Use output of update map as new prior

Need a composite like $\begin{pmatrix}A\\A\end{pmatrix} \rightleftarrows {}_{5}\begin{pmatrix}B\\B\\B\end{pmatrix}^{4} \leftrightarrows \begin{pmatrix}A\\A\end{pmatrix}$

Repeated Updating

Morphisms of $\mathbf{BLens}(\mathcal{C})$ are pairs
$$\begin{tikzcd} \begin{pmatrix}A\\X\end{pmatrix} \arrow[r, shift left, "f"] & \begin{pmatrix}B\\Y\end{pmatrix} \arrow[l, shift left] \end{tikzcd}$$
family of inverse kernels.

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in  observe 4 from Poisson(r);
        observe 5 from Poisson(r);
        return x
```

Modified example: multiple observations

$\Rightarrow$ Use output of update map as new prior

Need a composite like
$$\begin{tikzcd} \begin{pmatrix}A\\A\end{pmatrix} \arrow[r, shift left] & \begin{pmatrix}B\\B\\B\end{pmatrix} \arrow[l, shift left, "5\,\partial"] \arrow[r, shift right] & \begin{pmatrix}A\\A\end{pmatrix} \arrow[l, shift right, "\partial^{4}"'] \end{tikzcd}$$

Can we represent inverses in a category where this makes sense?

Aside: Representable Markov Categories

(Fritz, Gonda, Perrone, Rischel 2023)

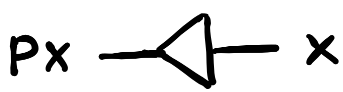
Aside: Representable Markov Categories

A Markov category is representable if every object X, has a distribution object PX:

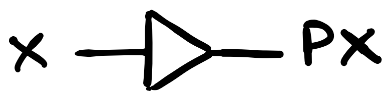
(Fritz, Gonda, Perrone, Rischel 2023)

Aside: Representable Markov Categories

A Markov category is representable if every object X, has a distribution object PX:



sampling morphism



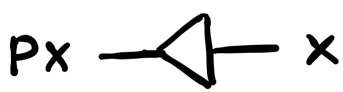
thunking morphism

such that

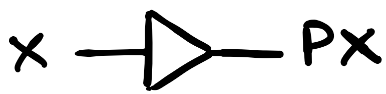
(Fritz, Gonda, Perrone, Rischel 2023)

Aside: Representable Markov Categories

A Markov category is representable if every object X, has a distribution object PX:

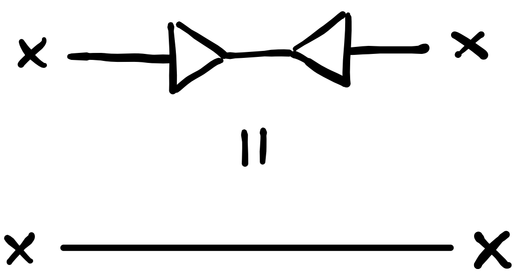


sampling morphism



thunking morphism

such that

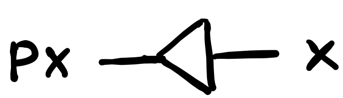


and

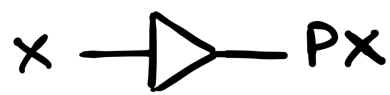
(Fritz, Gonda, Perrone, Rischel 2023)

Aside: Representable Markov Categories

A Markov category is representable if every object X, has a distribution object PX:

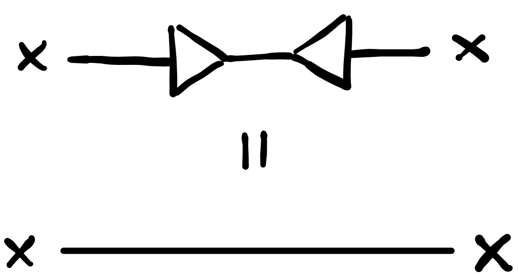


sampling morphism

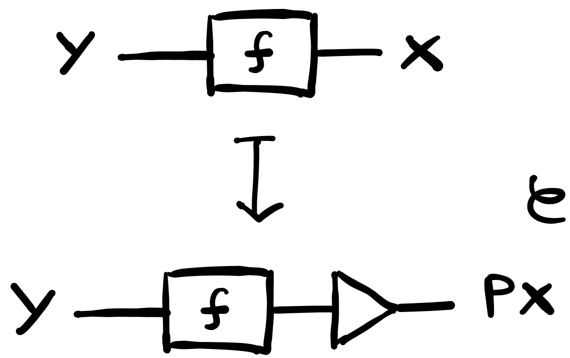


thunking morphism

such that



and



is an isomorphism $\mathcal{C}(Y,X) \cong \mathcal{C}_{det}(Y,PX)$

(Fritz, Gonda, Perrone, Rischel 2023)

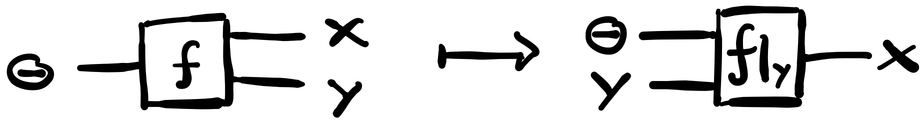
Internally Parameterised Inverses

Internally Parameterised Inverses

Markov category with conditionals:

Internally Parameterised Inverses

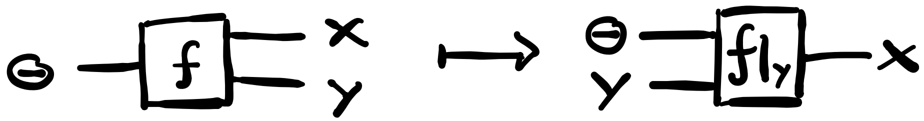
Markov category with conditionals:



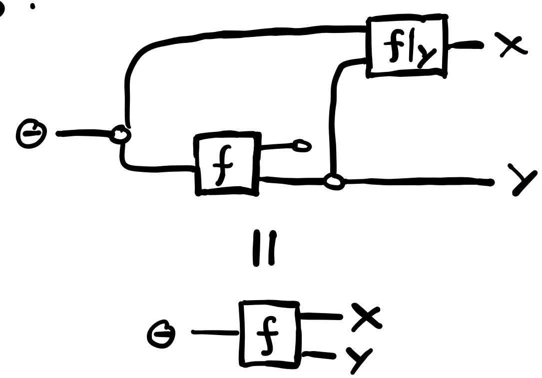
s.t.

Internally Parameterised Inverses

Markov category with conditionals:

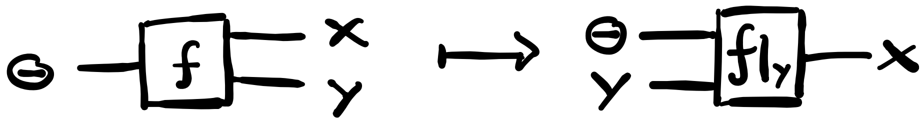


s.t.

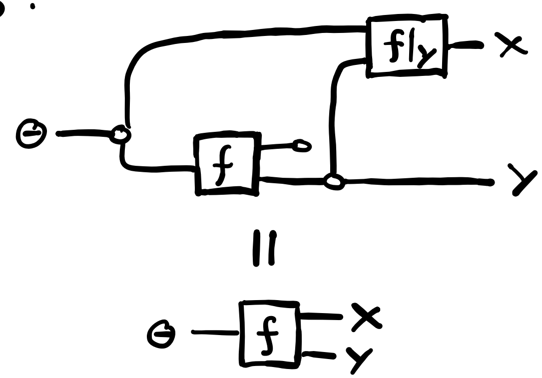


Internally Parameterised Inverses

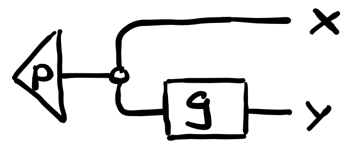
Markov category with conditionals:



s.t.



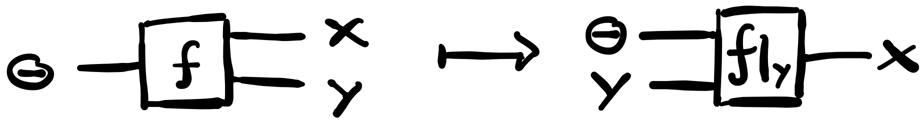
<u>Proposition</u> $g' : Y \to X$ is a conditional on Y of



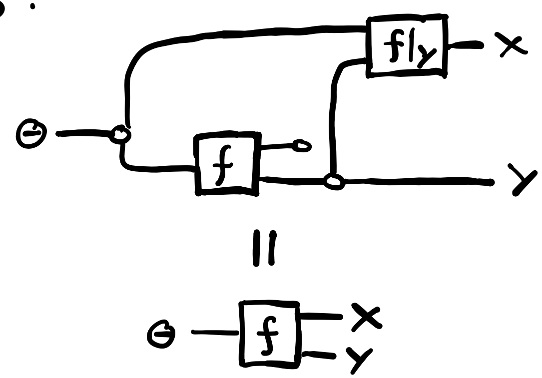
if and only if g' is a Bayesian inverse of g w.r.t. p.

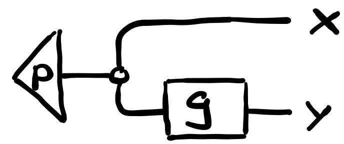
Internally Parameterised Inverses

Markov category with conditionals:



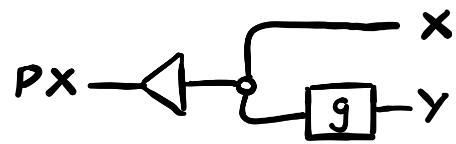
s.t.



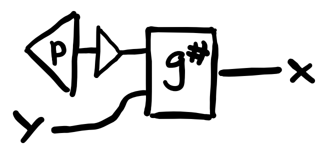
<u>Proposition</u> $g' : Y \to X$ is a conditional on Y of 

if and only if g' is a Bayesian inverse of g w.r.t. p.

<u>Proposition</u> (Parametric Inverses)

Let $g^{\#} : PX \otimes Y \to X$ be a conditional on Y of 

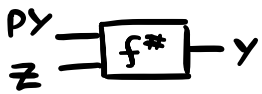
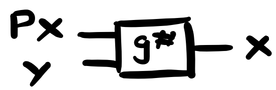
,

Then for all $p : I \to X$ we have that 

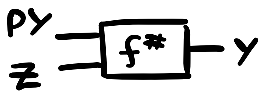
is a Bayesian inverse to g at p.

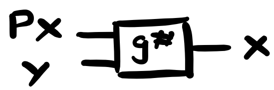
Composing Parametric Inverses

Composing Parametric Inverses

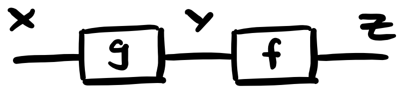
Let  ,  be 'parametric inverses' as defined in the previous proposition.

Composing Parametric Inverses

Let 

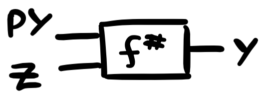
, 

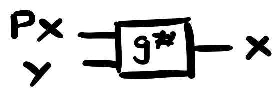
be 'parametric inverses' as defined in the previous proposition.

Q: Is the composite of $f^{\#}$ and $g^{\#}$ an inverse for 

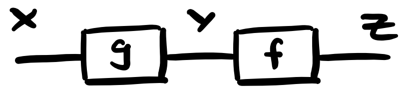
?

Composing Parametric Inverses

Let 

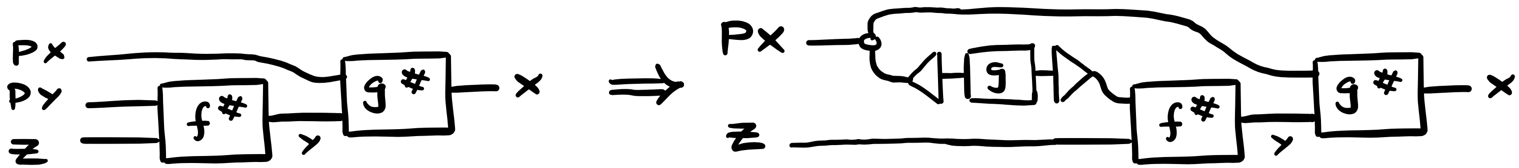
, 

be 'parametric inverses' as defined in the previous proposition.

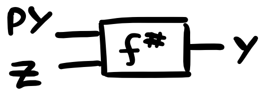
Q: Is the composite of $f^\#$ and $g^\#$ an inverse for 

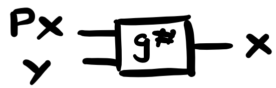
?

A: Yes, laxly.

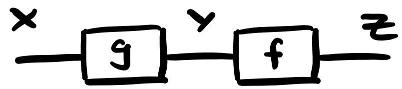


Composing Parametric Inverses

Let 

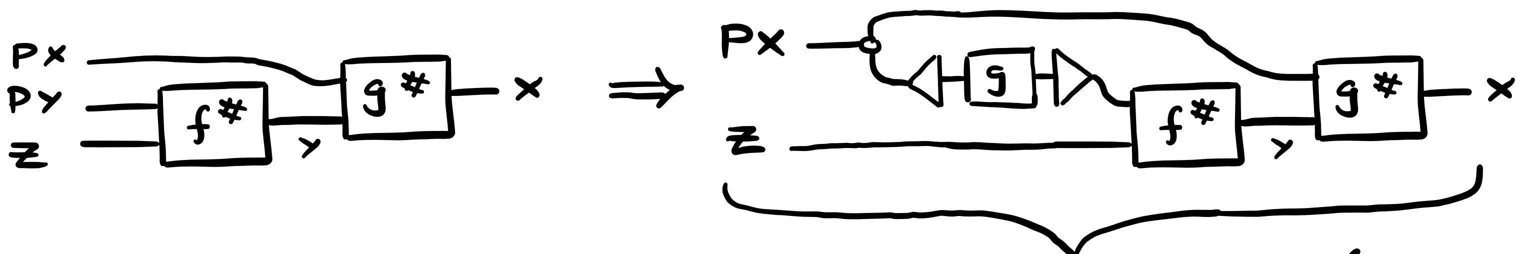
, 

be 'parametric inverses' as defined in the previous proposition.

Q: Is the composite of $f^{\#}$ and $g^{\#}$ an inverse for 

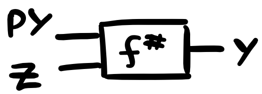
?

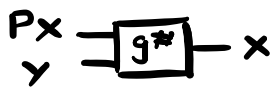
A: Yes, laxly.



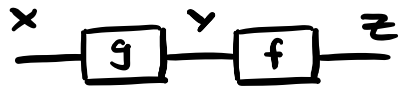
parametric inverse to f;g

Composing Parametric Inverses

Let 

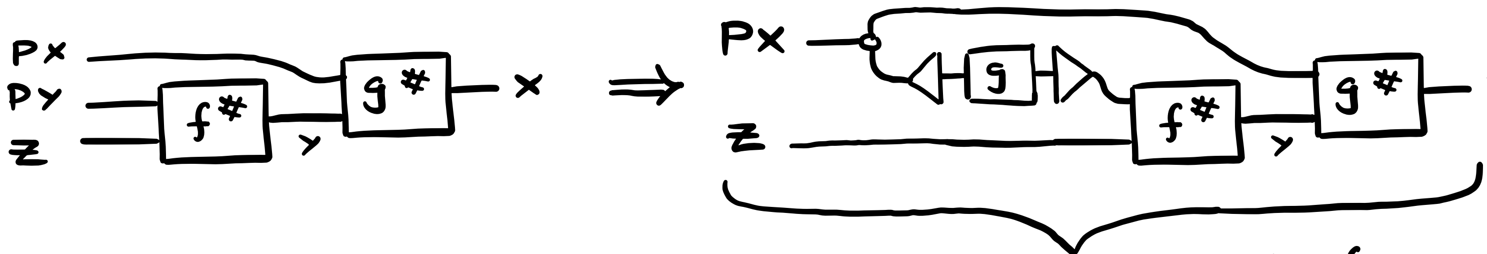
, 

be 'parametric inverses' as defined in the previous proposition.

Q: Is the composite of $f^{\#}$ and $g^{\#}$ an inverse for 

?

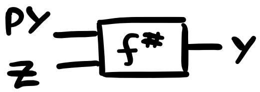
A: Yes, laxly.

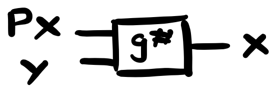


parametric inverse to f;g

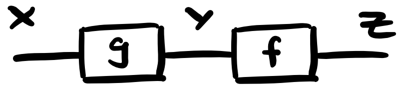
$\text{Inv} : \mathcal{C}^{op} \longrightarrow \text{Para}(\mathcal{C})$ is 'almost surely' a lax functor

Composing Parametric Inverses

Let 

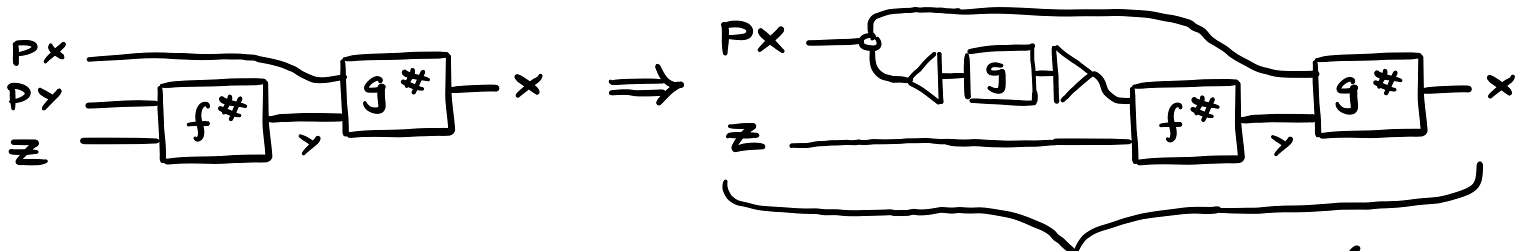
, 

be 'parametric inverses' as defined in the previous proposition.

Q: Is the composite of $f^{\#}$ and $g^{\#}$ an inverse for 

?

A: Yes, laxly.



parametric inverse to $f;g$

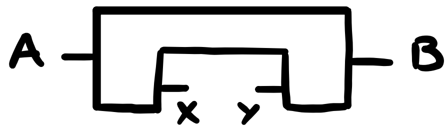
$\text{Inv} : \mathcal{C}^{op} \longrightarrow \text{Para}(\mathcal{C})$ is 'almost surely' a lax functor

Q: Why is this lax while the inversion functor $\mathcal{C} \longrightarrow \text{BLens}(\mathcal{C})$ was strict?

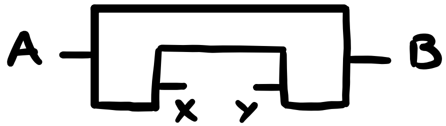
Stochastic Combs

Stochastic Combs

A 1-comb $\begin{pmatrix} A \\ B \end{pmatrix} \to \begin{pmatrix} X \\ Y \end{pmatrix}$ is



Stochastic Combs

A 1-comb $\begin{pmatrix}A\\B\end{pmatrix} \longrightarrow \begin{pmatrix}X\\Y\end{pmatrix}$ is 

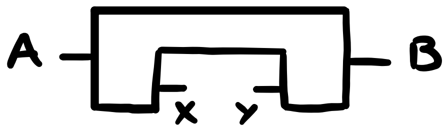
$$\mathrm{Comb}_1\left(\begin{pmatrix}A\\B\end{pmatrix};\begin{pmatrix}X\\Y\end{pmatrix}\right) = \int^{M} \mathcal{C}(A, M\otimes X)\times \mathcal{C}(M\otimes Y, B)$$

Stochastic Combs

AKA: "optics"

"monoidal lenses"

"doubles"

A 1-comb $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \begin{pmatrix} X \\ Y \end{pmatrix}$ is 

$$\mathrm{Comb}_1\left(\begin{pmatrix} A \\ B \end{pmatrix} ; \begin{pmatrix} X \\ Y \end{pmatrix}\right) = \int^M \mathcal{C}(A, M \otimes X) \times \mathcal{C}(M \otimes Y, B)$$

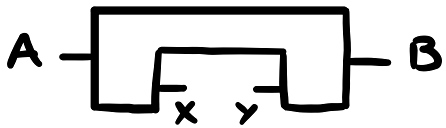
Stochastic Combs

AKA: "optics"

"monoidal lenses"

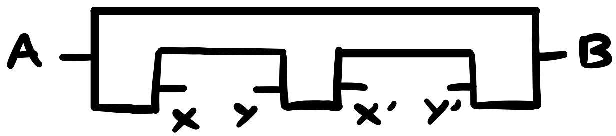
"doubles"

A 1-comb $\begin{pmatrix} A \\ B \end{pmatrix} \to \begin{pmatrix} X \\ Y \end{pmatrix}$ is



$$\mathrm{Comb}_1\left(\begin{pmatrix} A \\ B \end{pmatrix};\begin{pmatrix} X \\ Y \end{pmatrix}\right) = \int^M \mathcal{C}(A, M\otimes X)\times \mathcal{C}(M\otimes Y, B)$$

A 2-comb $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \begin{pmatrix} X \\ Y \end{pmatrix}, \begin{pmatrix} X' \\ Y' \end{pmatrix}$ is



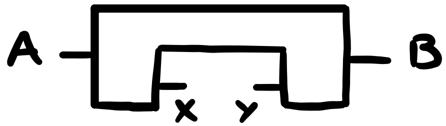
Stochastic Combs

AKA: "optics"

"monoidal lenses"

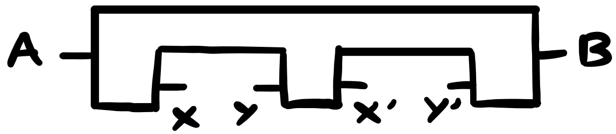
"doubles"

A 1-comb $\begin{pmatrix} A \\ B \end{pmatrix} \to \begin{pmatrix} X \\ Y \end{pmatrix}$ is



$$\mathrm{Comb}_1\left(\begin{pmatrix} A \\ B \end{pmatrix} ; \begin{pmatrix} X \\ Y \end{pmatrix}\right) = \int^{M} \mathcal{C}(A, M \otimes X) \times \mathcal{C}(M \otimes Y, B)$$

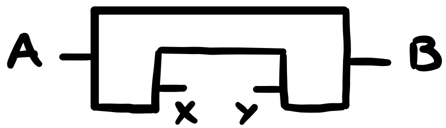
A 2-comb $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \begin{pmatrix} X \\ Y \end{pmatrix}, \begin{pmatrix} X' \\ Y' \end{pmatrix}$ is



Stochastic Combs

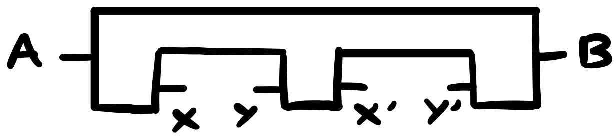
AKA: "optics"
"monoidal lenses"
"doubles"

A 1-comb $\begin{pmatrix} A \\ B \end{pmatrix} \to \begin{pmatrix} X \\ Y \end{pmatrix}$ is



$$\mathrm{Comb}_1\left(\begin{pmatrix} A \\ B \end{pmatrix} ; \begin{pmatrix} X \\ Y \end{pmatrix}\right) = \int^{M} \mathcal{C}(A, M \otimes X) \times \mathcal{C}(M \otimes Y, B)$$

A 2-comb $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \begin{pmatrix} X \\ Y \end{pmatrix}, \begin{pmatrix} X' \\ Y' \end{pmatrix}$ is



$$\mathrm{Comb}_2\left(\begin{pmatrix} A \\ B \end{pmatrix} ; \begin{pmatrix} X \\ Y \end{pmatrix}, \begin{pmatrix} X' \\ Y' \end{pmatrix}\right) = \int^{M,N} \mathcal{C}(A, M \otimes X) \times \mathcal{C}(M \otimes Y, N \otimes X') \times \mathcal{C}(N \otimes Y', B)$$

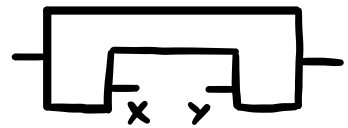
Stochastic Combs

AKA: "optics"

"monoidal lenses"

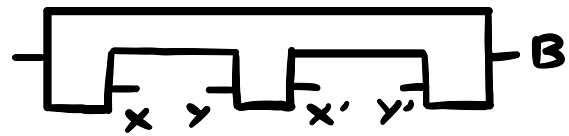
"doubles"

A 1-comb $\binom{A}{B} \to \binom{X}{Y}$ is



$$\mathrm{Comb}_1\left(\binom{A}{B};\binom{X}{Y}\right) = \int^{M} \mathcal{C}(A, M\otimes X)\times \mathcal{C}(M\otimes Y, B)$$

A 2-comb $\binom{A}{B} \to \binom{X}{Y}, \binom{X'}{Y'}$ is

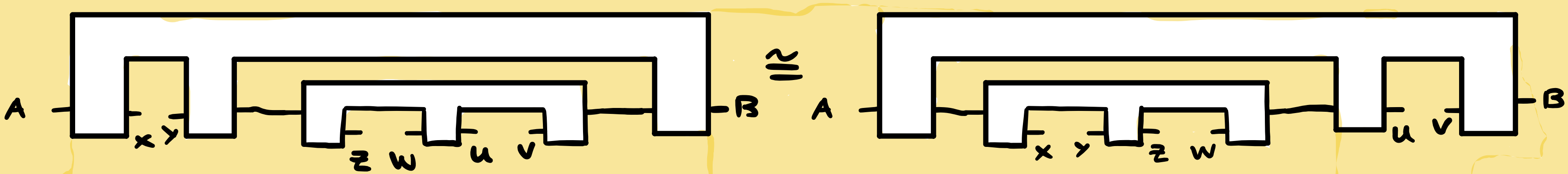


$$\mathrm{Comb}_2\left(\binom{A}{B};\binom{X}{Y},\binom{X'}{Y'}\right) = \int^{M,N} \mathcal{C}(A, M\otimes X)\times \mathcal{C}(M\otimes Y, N\otimes X')\times \mathcal{C}(N\otimes Y', B)$$

Define n-combs inductively by profunctor composition:

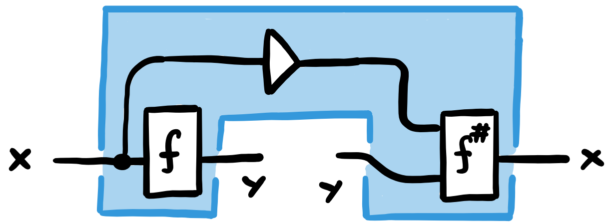
Earnshaw, Hefford & Román (CSL 2023):

Comb composition is a promonoidal product.

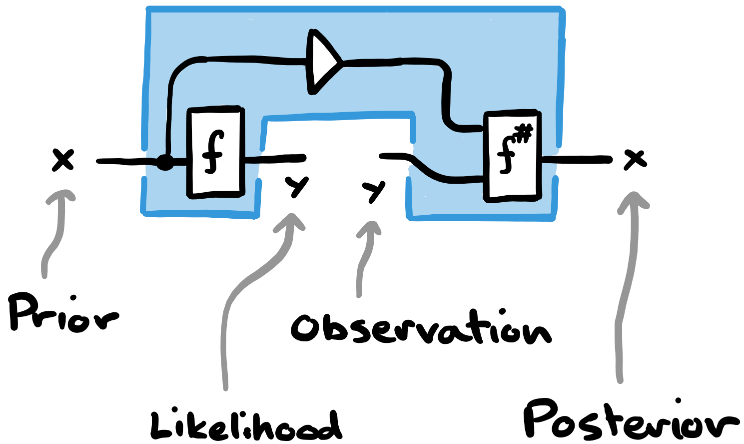


Bayesian Inverses as Combs

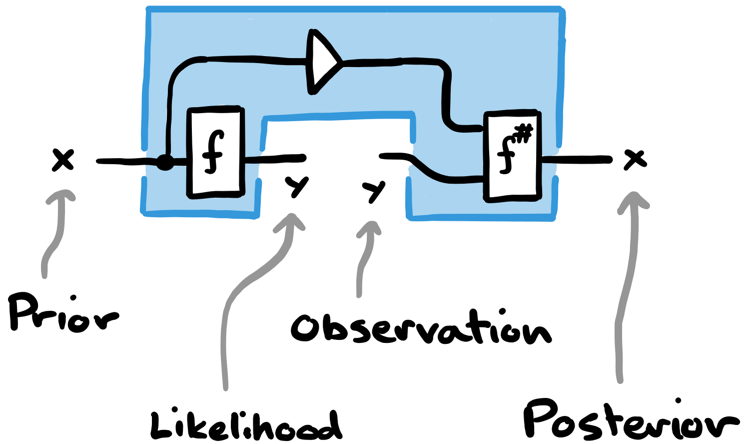
Bayesian Inverses as Combs



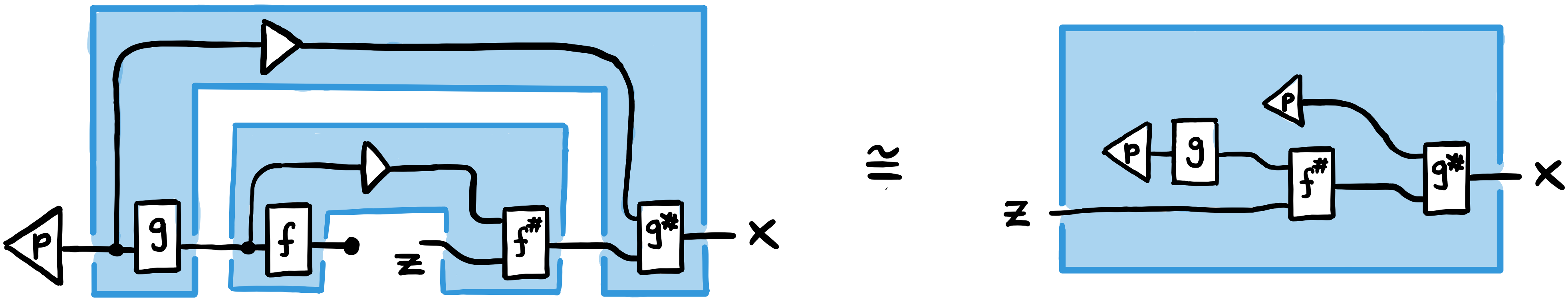
Bayesian Inverses as Combs



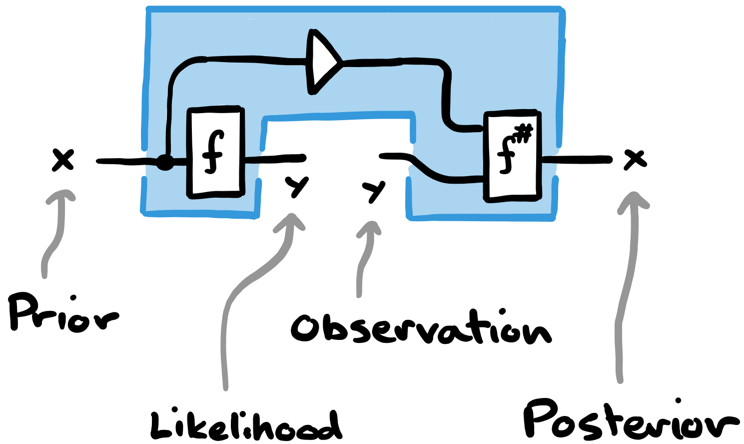
Bayesian Inverses as Combs



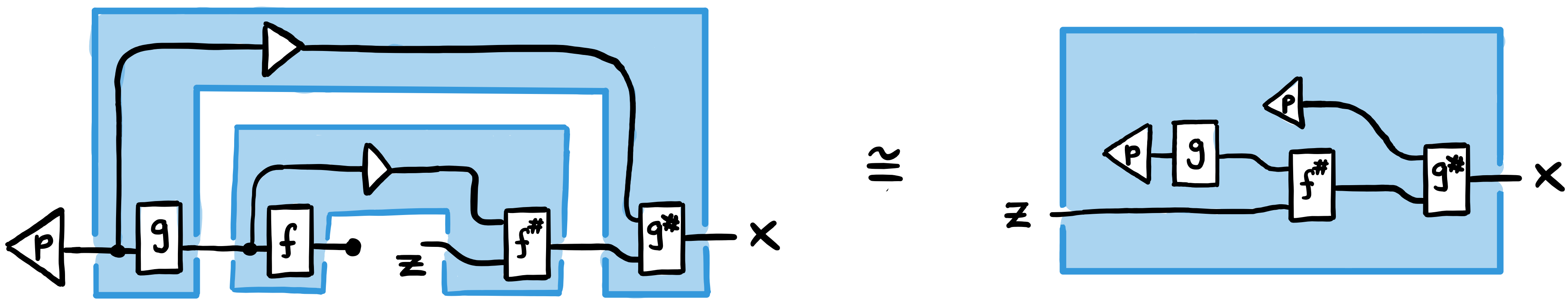
- Composite inverses are nested combs



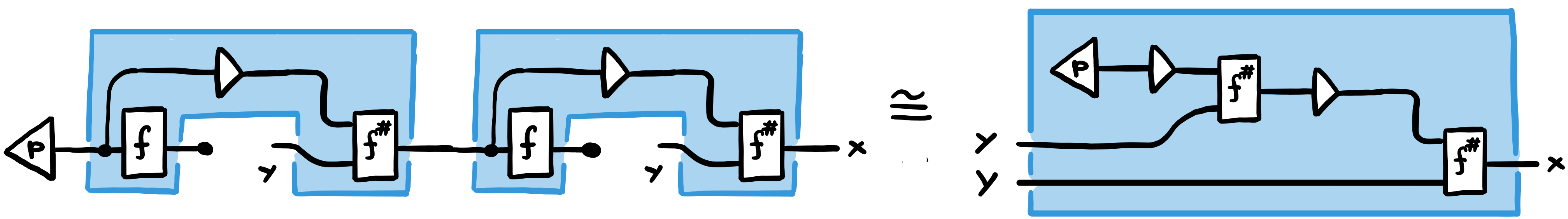
Bayesian Inverses as Combs



- Composite inverses are nested combs



- Repeated updates are end-to-end composition



Example: Building statistical models from combs

Example: Building statistical models from combs

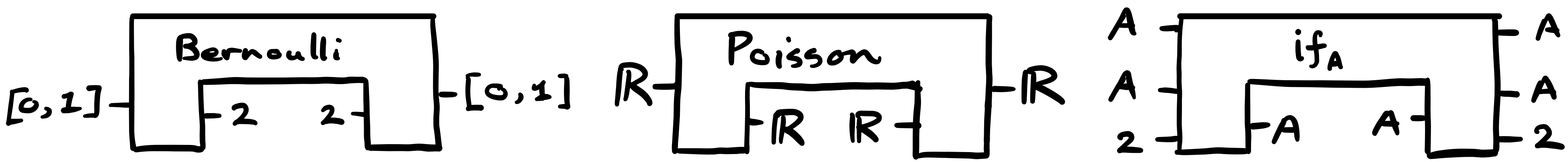
```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in observe 4 from Poisson(r);
       observe 5 from Poisson(r);
       return x
```

Example from earlier.

Example: Building statistical models from combs

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in observe 4 from Poisson(r);
       observe 5 from Poisson(r);
       return x
```

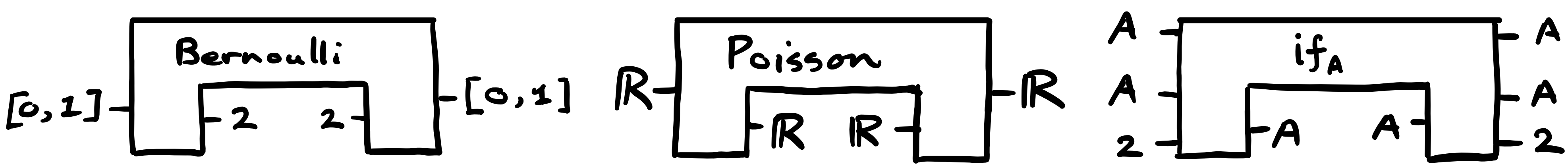
Example from earlier.

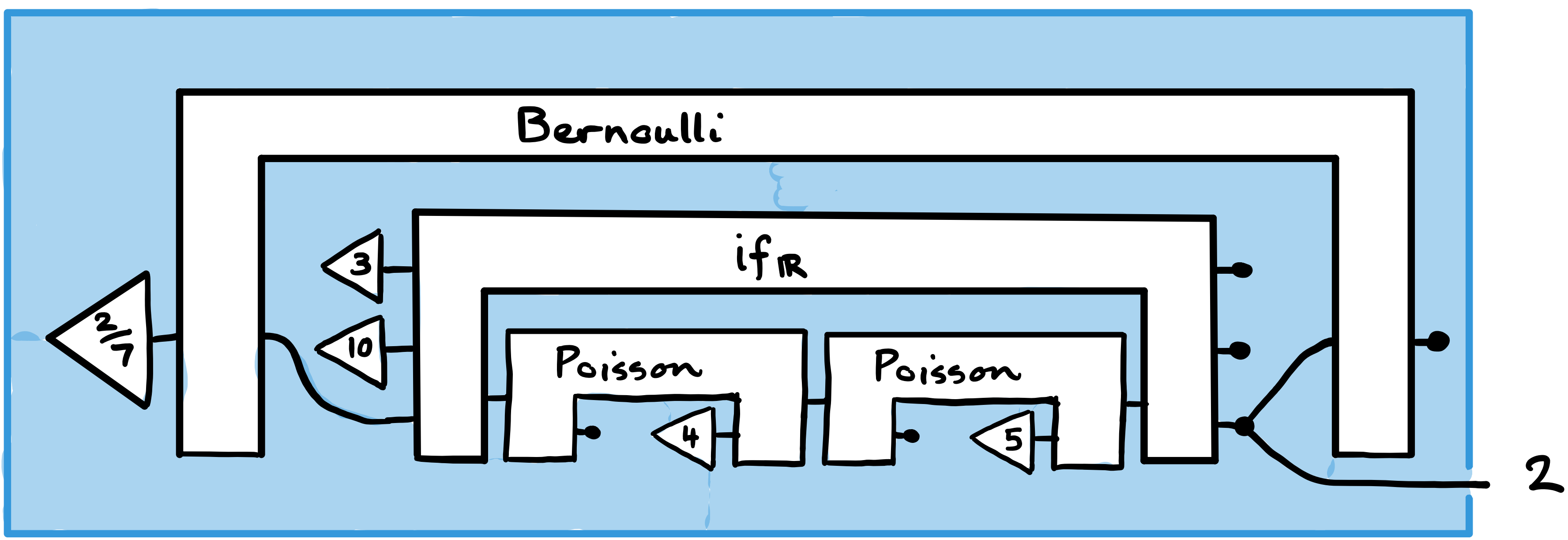


Example: Building statistical models from combs

```
def example():
    let x := sample(Bernoulli(2/7))
        r := if x then 3 else 10
    in  observe 4 from Poisson(r);
        observe 5 from Poisson(r);
        return x
```

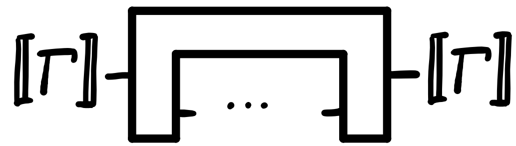
Example from earlier.





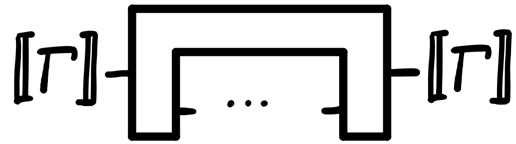
Combs as Models for Probabilistic Programs

Combs as Models for Probabilistic Programs

Expressions in context Γ are modelled as 

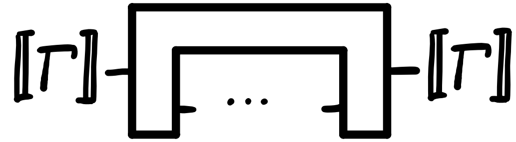
Combs as Models for Probabilistic Programs

Expressions in context Γ are modelled as



Three kinds of expression:

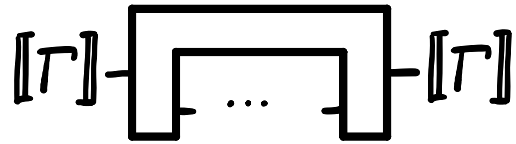
Combs as Models for Probabilistic Programs

Expressions in context Γ are modelled as 

Three kinds of expression:

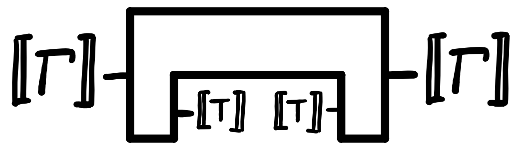
- Typed terms $\quad \Gamma \vdash t : T$ 

Combs as Models for Probabilistic Programs

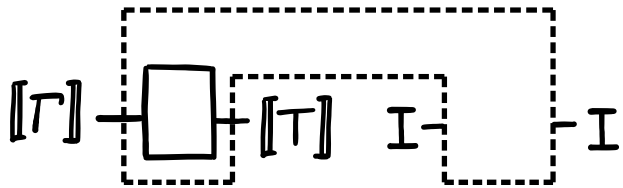
Expressions in context Γ are modelled as 

Three kinds of expression:

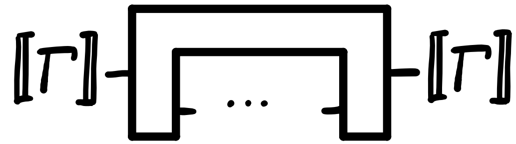
- Typed terms $\Gamma \vdash t : T$



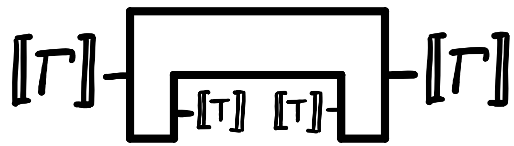
- Values $\Gamma \vdash_v t : T$

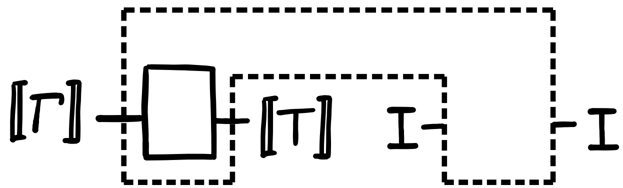


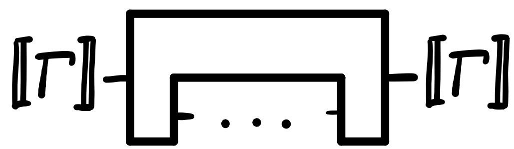
Combs as Models for Probabilistic Programs

Expressions in context Γ are modelled as 

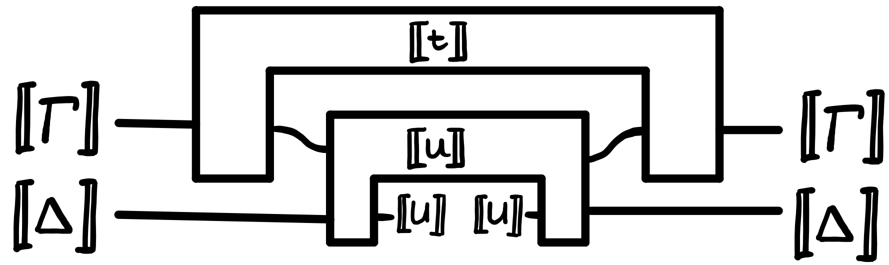
Three kinds of expression:

- Typed terms $\quad \Gamma \vdash t : T$ 

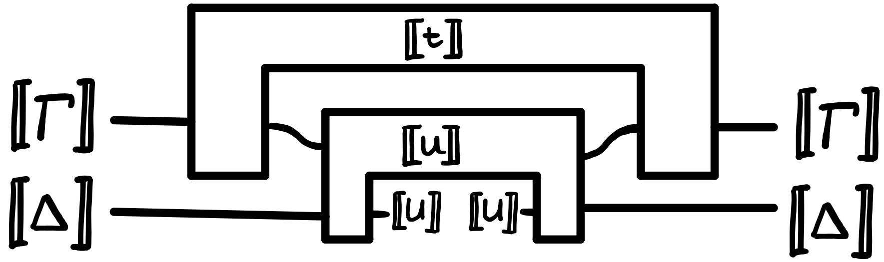
- Values $\quad \Gamma \vdash_v t : T$ 

- Statements $\quad \Gamma \vdash s \ \mathrm{stmt}$ 

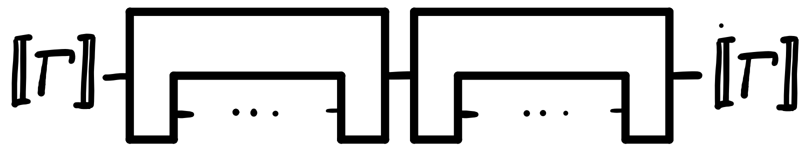
$$\frac{\Gamma \vdash t : T \qquad \Delta, T \vdash u : U}{\Delta, \Gamma \vdash u[t] : U}$$



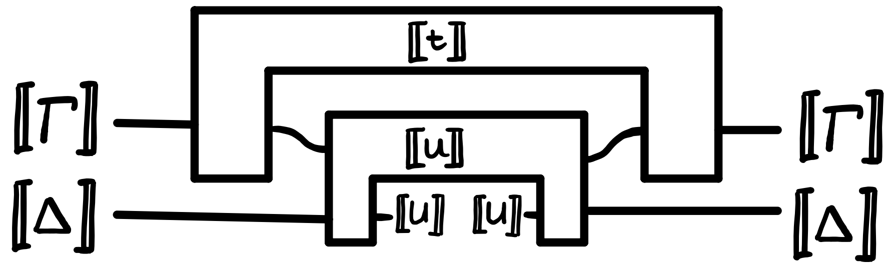
$$\frac{\Gamma \vdash t : T \qquad \Delta, T \vdash u : U}{\Delta, \Gamma \vdash u[t] : U}$$



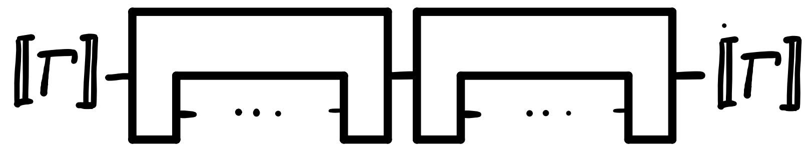
$$\frac{\Gamma \vdash s_1 \ \mathsf{stmt} \qquad \Gamma \vdash s_2 \ \mathsf{stmt}}{\Gamma \vdash s_1 ; s_2 \ \mathsf{stmt}}$$



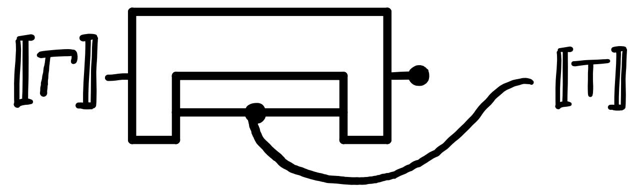
$$\frac{\Gamma \vdash t : T \qquad \Delta, T \vdash u : U}{\Delta, \Gamma \vdash u[t] : U}$$



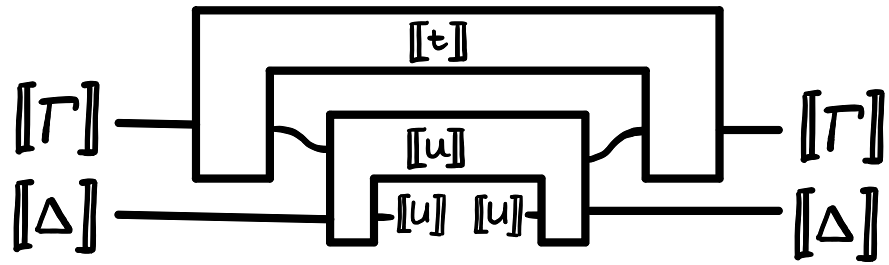
$$\frac{\Gamma \vdash s_1 \ \mathsf{stmt} \qquad \Gamma \vdash s_2 \ \mathsf{stmt}}{\Gamma \vdash s_1 ; s_2 \ \mathsf{stmt}}$$



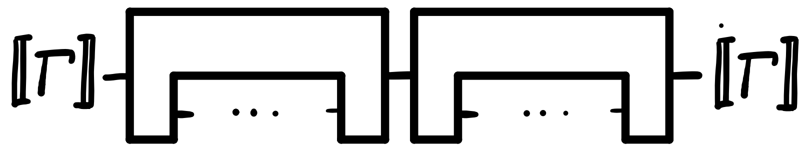
$$\frac{\Gamma \vdash t : T}{\Gamma \vdash_v \mathsf{sample}(t) : T}$$



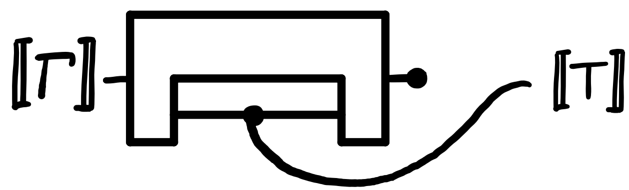
$$\frac{\Gamma \vdash t : T \qquad \Delta, T \vdash u : U}{\Delta, \Gamma \vdash u[t] : U}$$



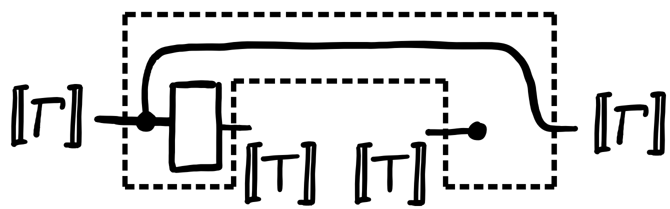
$$\frac{\Gamma \vdash s_1 \ \mathsf{stmt} \qquad \Gamma \vdash s_2 \ \mathsf{stmt}}{\Gamma \vdash s_1 ; s_2 \ \mathsf{stmt}}$$



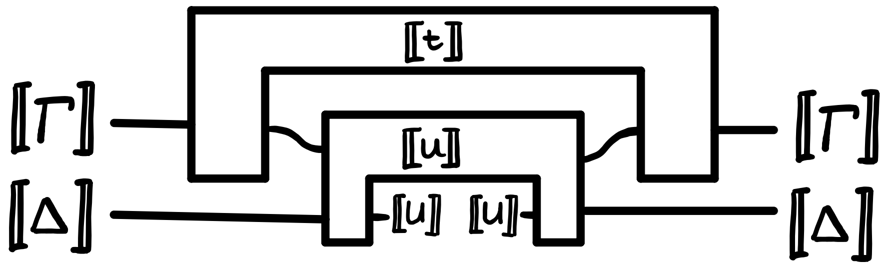
$$\frac{\Gamma \vdash t : T}{\Gamma \vdash_v \mathsf{sample}(t) : T}$$



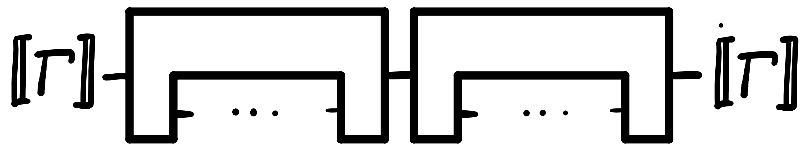
$$\frac{\Gamma \vdash_v t : T}{\Gamma \vdash \mathsf{certain}(t) : T}$$



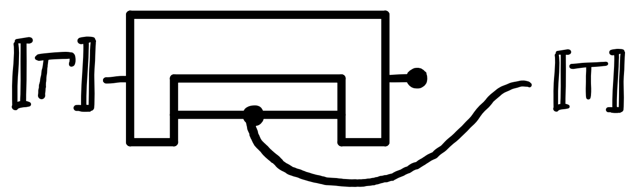
$$\frac{\Gamma \vdash t : T \qquad \Delta, T \vdash u : U}{\Delta, \Gamma \vdash u[t] : U}$$



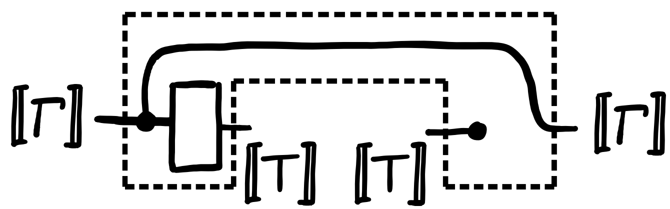
$$\frac{\Gamma \vdash s_1 \ \mathrm{stmt} \qquad \Gamma \vdash s_2 \ \mathrm{stmt}}{\Gamma \vdash s_1 ; s_2 \ \mathrm{stmt}}$$



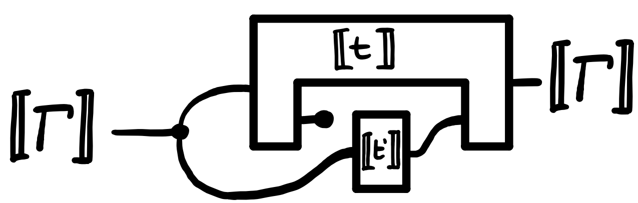
$$\frac{\Gamma \vdash t : T}{\Gamma \vdash_v \mathrm{sample}(t) : T}$$



$$\frac{\Gamma \vdash_v t : T}{\Gamma \vdash \mathrm{certain}(t) : T}$$



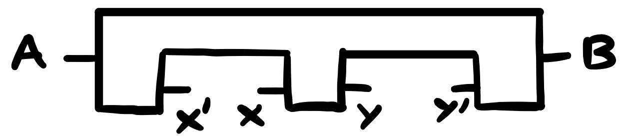
$$\frac{\Gamma \vdash t : T \qquad \Gamma \vdash_v t' : T}{\Delta, \Gamma \vdash (\mathrm{observe}\ t' \ \mathrm{from}\ t) \ \mathrm{stmt}}$$



Other thoughts

Other thoughts

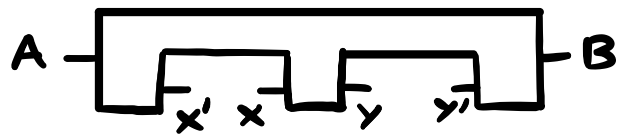
- Higher order terms :



behaves like a 'curried comb' $\begin{pmatrix}A\\B\end{pmatrix} \longrightarrow \left[\begin{pmatrix}x\\y\end{pmatrix} \multimap \begin{pmatrix}x'\\y'\end{pmatrix}\right]$.

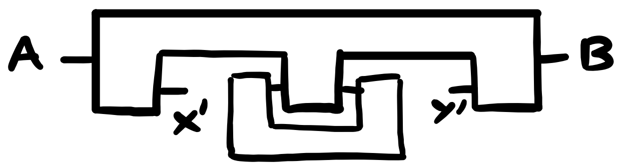
Other thoughts

- Higher order terms :



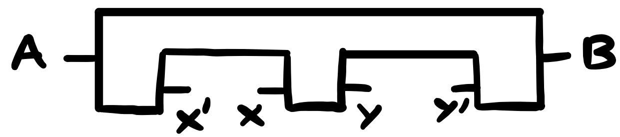
behaves like a 'curried comb' $\begin{pmatrix}A\\B\end{pmatrix} \longrightarrow \left[\begin{pmatrix}X\\Y\end{pmatrix} \multimap \begin{pmatrix}X'\\Y'\end{pmatrix}\right]$.

Eval :



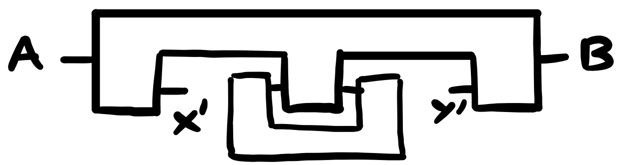
Other thoughts

- Higher order terms :

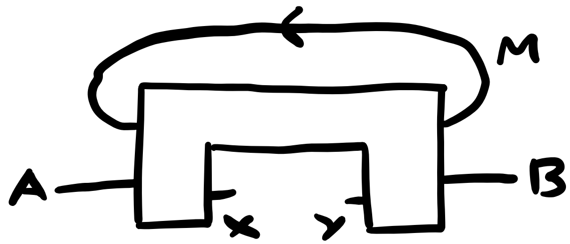


behaves like a 'curried comb' $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \left[\begin{pmatrix} X \\ Y \end{pmatrix} \multimap \begin{pmatrix} X' \\ Y' \end{pmatrix} \right]$.

Eval :

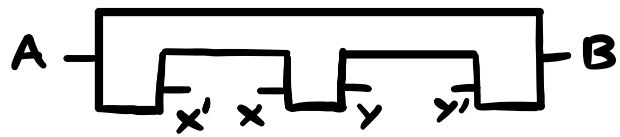


- Iteration ?



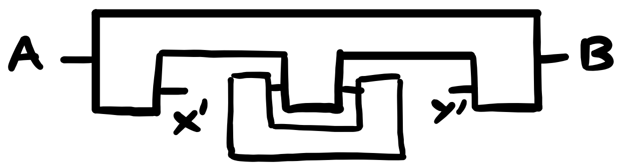
Other thoughts

- Higher order terms :

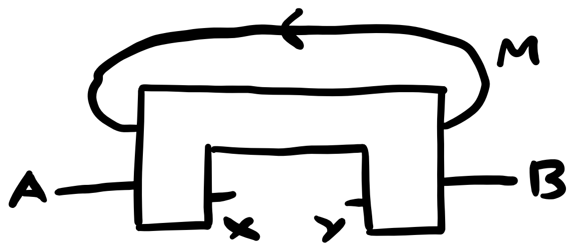


behaves like a 'curried comb' $\begin{pmatrix}A\\B\end{pmatrix} \longrightarrow \left[\begin{pmatrix}X\\Y\end{pmatrix} \multimap \begin{pmatrix}X'\\Y'\end{pmatrix}\right].$

Eval :

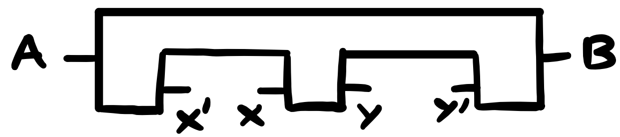


- Iteration ?

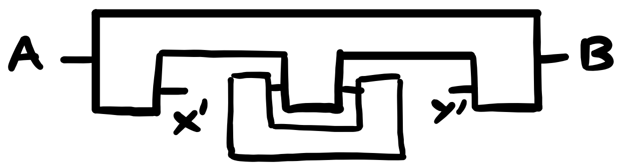


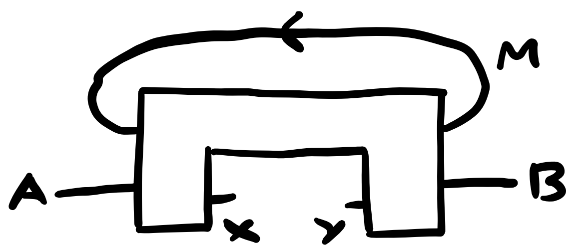
- Holes as models of I/O ?

Other thoughts

- Higher order terms : 

  behaves like a 'curried comb' $\begin{pmatrix} A \\ B \end{pmatrix} \longrightarrow \left[\begin{pmatrix} X \\ Y \end{pmatrix} \multimap \begin{pmatrix} X' \\ Y' \end{pmatrix} \right]$.

  Eval : 

- Iteration ? 

- Holes as models of I/O ?

  Can something akin to session types give a more first-class way to manipulate holes ?

Beyond representable Markov categories?

Beyond representable Markov categories?

Recall: Externally parameterized inverses in BLens

Beyond representable Markov categories?

Recall: Externally parameterized inverses in BLens

$$\mathrm{BLens}\left(\begin{matrix} A \\ X \end{matrix}, \begin{matrix} B \\ Y \end{matrix}\right) \cong \mathrm{Optic}_{\times, \pitchfork}\left(\begin{matrix} \text{よ}^{A} \\ \text{よ}_{x} \end{matrix}, \begin{matrix} \text{よ}^{B} \\ \text{よ}_{x} \end{matrix}\right)$$

Beyond representable Markov categories?

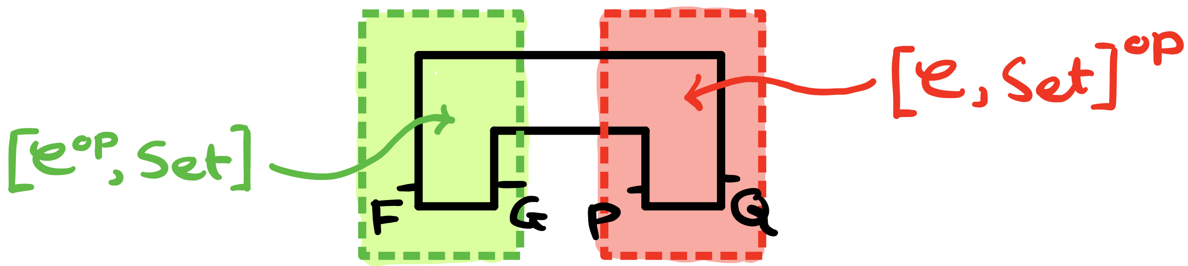
Recall : Externally parameterized inverses in BLens

$$\mathrm{BLens}\begin{pmatrix} A & B \\ X\ , & Y \end{pmatrix} \cong \mathrm{Optic}_{\times,\pitchfork}\begin{pmatrix} よ^{A} & よ^{B} \\ よ_{X}\ , & よ_{X} \end{pmatrix}$$

Mixed optics :

- Left category $[\mathcal{C}^{op}, \mathrm{Set}]$
- Right category $[\mathcal{C}, \mathrm{Set}]^{op}$

} connected by monoidal actions of $[\mathcal{C}^{op}, \mathrm{Set}]$



Beyond representable Markov categories?

Recall: Externally parameterized inverses in BLens

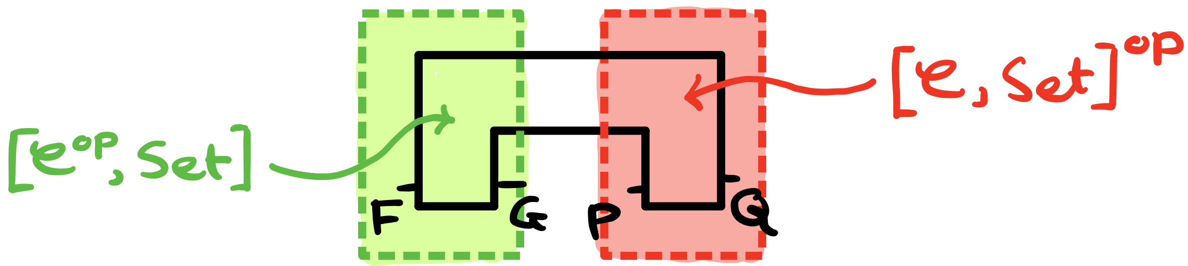
$$\mathbf{BLens}\left(\begin{pmatrix}A & B\\ X, & Y\end{pmatrix}\right) \cong \mathbf{Optic}_{\times,\pitchfork}\left(\begin{pmatrix}\mathcal{K}^{A} & \mathcal{K}^{B}\\ \mathcal{K}_{X}, & \mathcal{K}_{X}\end{pmatrix}\right)$$

Mixed optics:

- Left category $[\mathcal{C}^{op}, \mathbf{Set}]$
- Right category $[\mathcal{C}, \mathbf{Set}]^{op}$

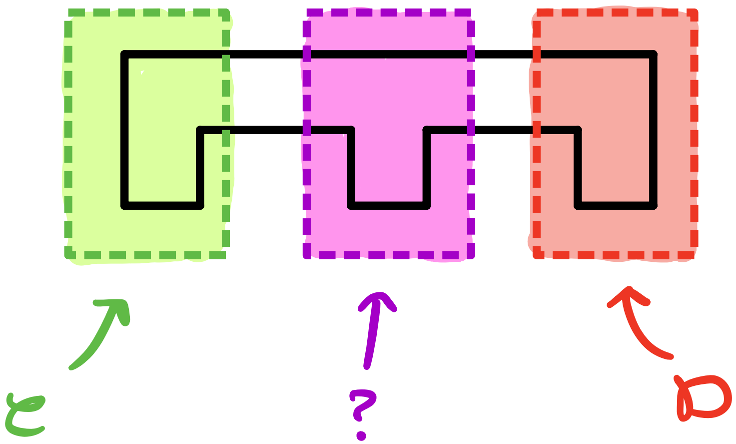
connected by monoidal actions of $[\mathcal{C}^{op}, \mathbf{Set}]$

$$\begin{pmatrix}F\\P\end{pmatrix} \longrightarrow \begin{pmatrix}G\\Q\end{pmatrix} \sim \int^{R\in\hat{\mathcal{C}}} \hat{\mathcal{C}}(F, R\times G) \times \check{\mathcal{C}}^{op}(Q^{R(I)}, P)$$



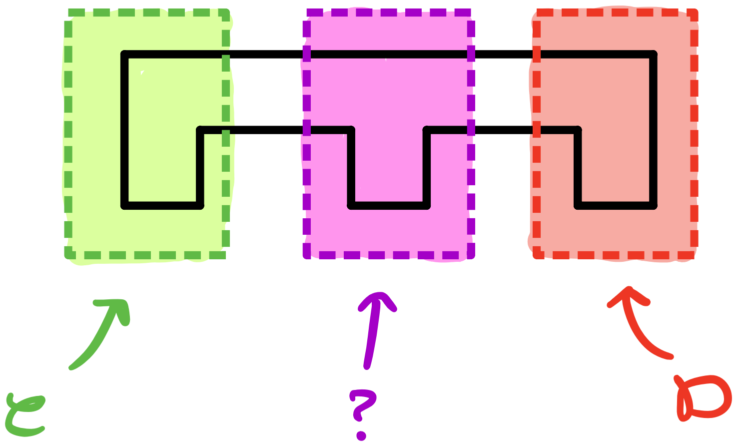
Beyond representable Markov categories?

Beyond representable Markov categories?



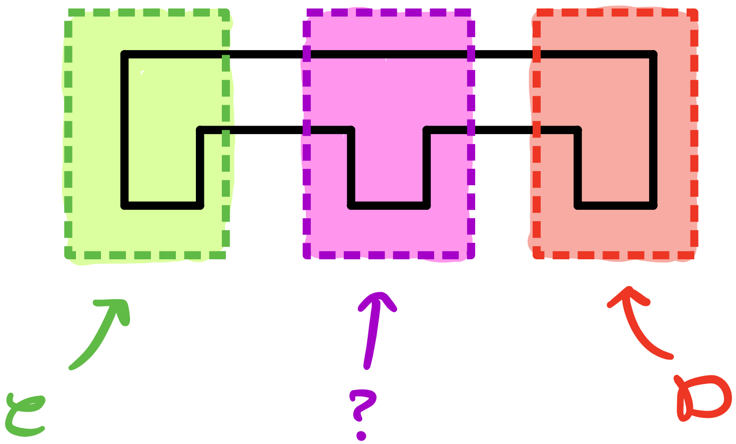
- Need a notion of heteromorphism $\mathcal{D} \nrightarrow \mathcal{C}$

Beyond representable Markov categories?



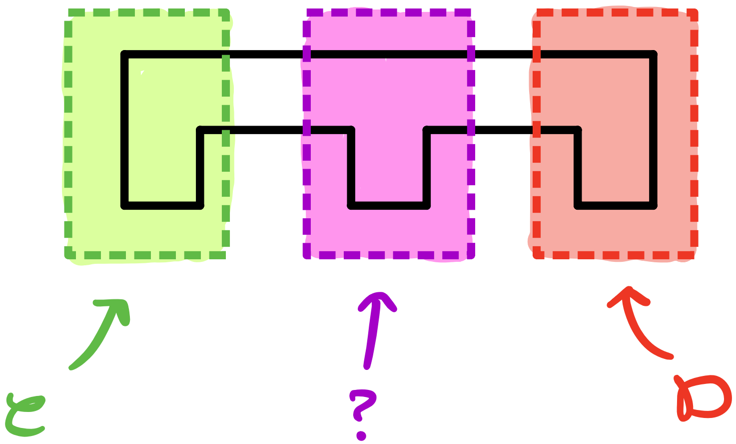
- Need a notion of heteromorphism $\mathcal{D} \nrightarrow \mathcal{C}$
- Compatible with the monoidal actions

Beyond representable Markov categories?



- Need a notion of heteromorphism $\mathcal{D} \nrightarrow \mathcal{C}$
- Compatible with the monoidal actions
- With suitable operations for 'filling holes' in diagrams.

Beyond representable Markov categories?



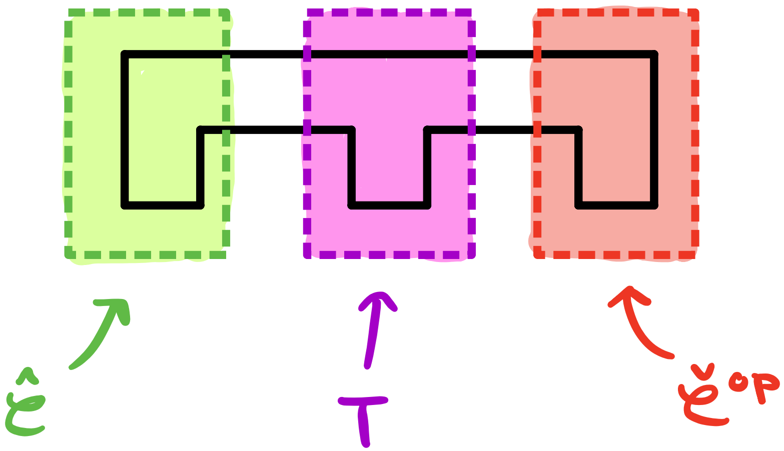
- Need a notion of heteromorphism $\mathcal{D} \nrightarrow \mathcal{C}$
- Compatible with the monoidal actions
- With suitable operations for 'filling holes' in diagrams.

<u>Claim</u>: If $T : \mathcal{D} \nrightarrow \mathcal{C}$ is an equivalence in Tamb (the bicategory of Tambara modules) then

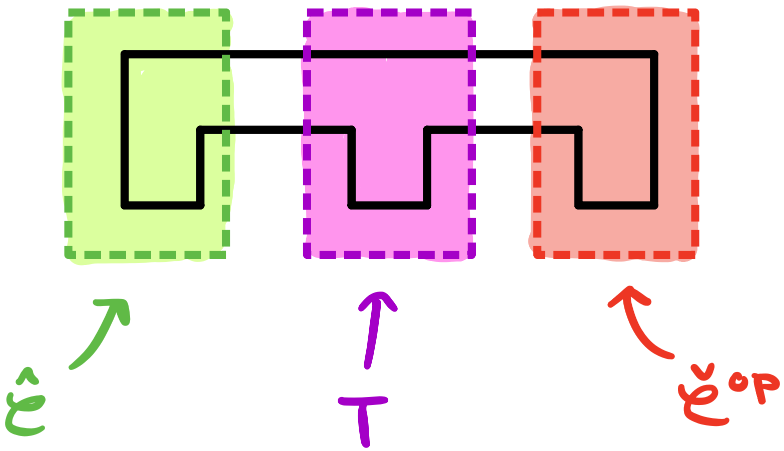
$$\begin{pmatrix} A \\ X \end{pmatrix}, \begin{pmatrix} B \\ Y \end{pmatrix}, \begin{pmatrix} C \\ Z \end{pmatrix} \longmapsto \int^{M,N} \mathcal{C}(A, M \bullet B) \times T(M \bullet Y, N \bullet C) \times \mathcal{D}(N \bullet Z, X)$$

defines a promonoidal product.

Beyond representable Markov categories?

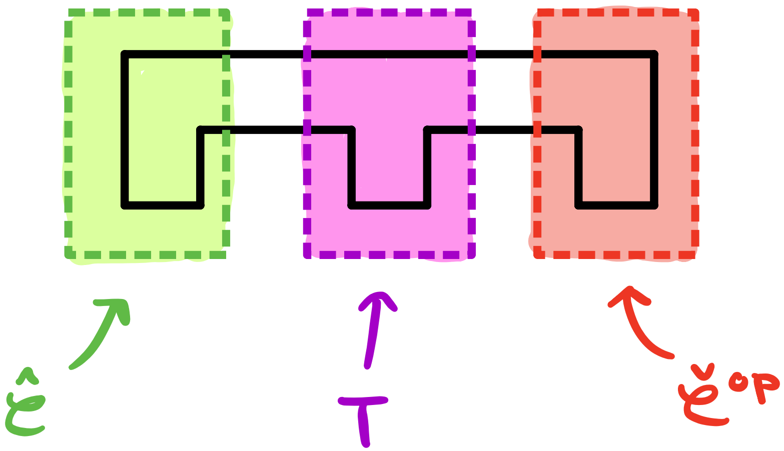


Beyond representable Markov categories?



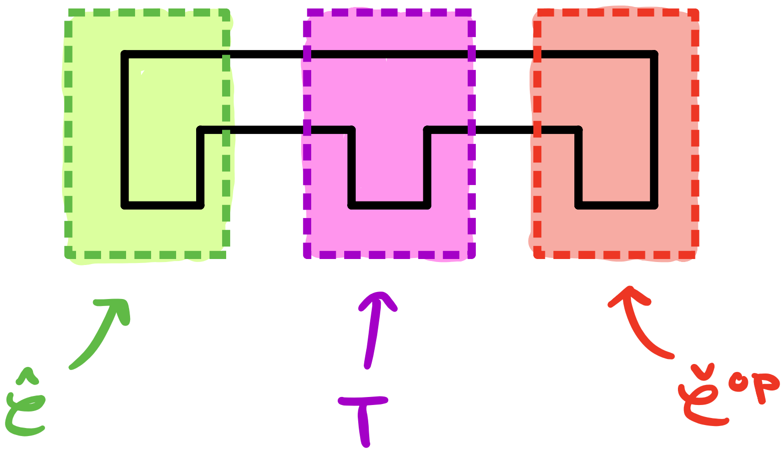
- Need a notion of heteromorphism $\check{\mathcal{C}}^{op} \nrightarrow \hat{\mathcal{C}}$

Beyond representable Markov categories?



- Need a notion of heteromorphism $\check{\mathcal{C}}^{op} \nrightarrow \hat{\mathcal{C}}$
- An equivalence in Tamb

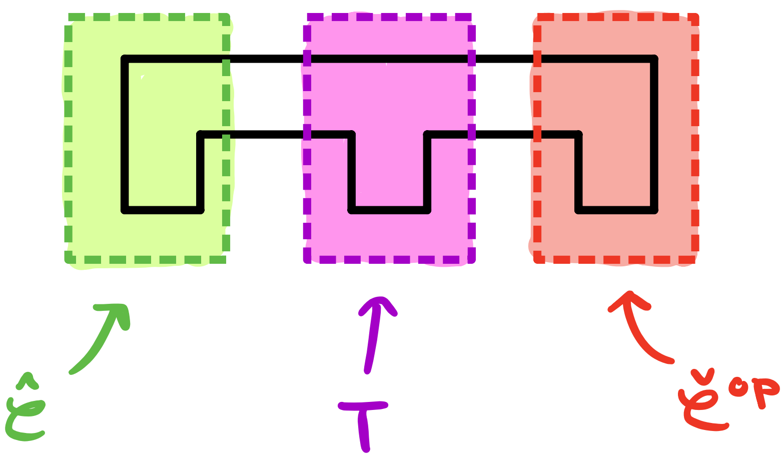
Beyond representable Markov categories?



$$T : \check{\mathcal{C}} \times \hat{\mathcal{C}} \longrightarrow \mathrm{Set}$$

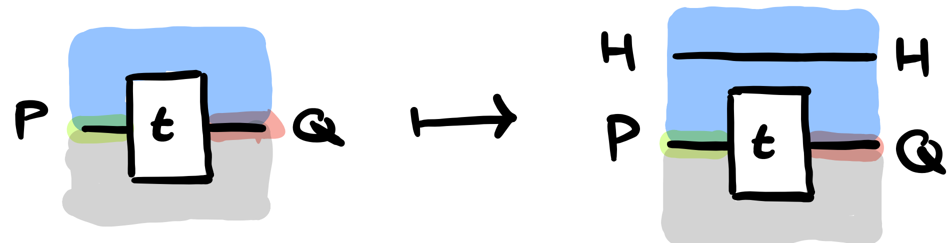
- Need a notion of heteromorphism $\check{\mathcal{C}}^{op} \nrightarrow \hat{\mathcal{C}}$
- An equivalence in Tamb

Beyond representable Markov categories?



- Need a notion of heteromorphism $\check{\mathcal{C}}^{op} \nrightarrow \hat{\mathcal{C}}$
- An equivalence in Tamb

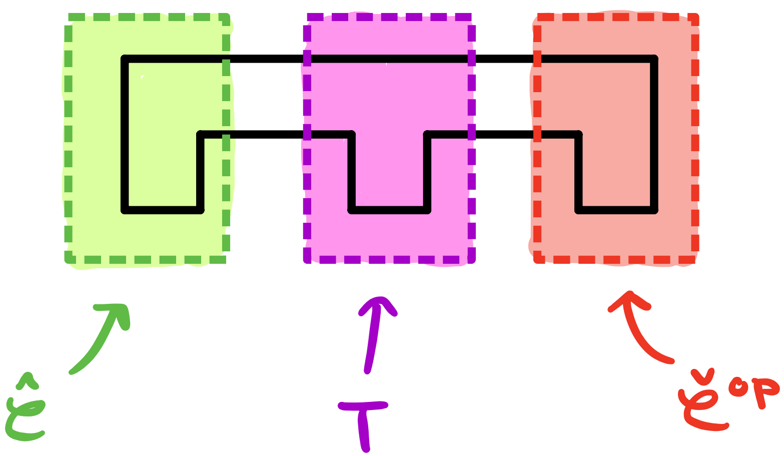
$$T : \check{\mathcal{C}} \times \hat{\mathcal{C}} \longrightarrow \mathrm{Set}$$



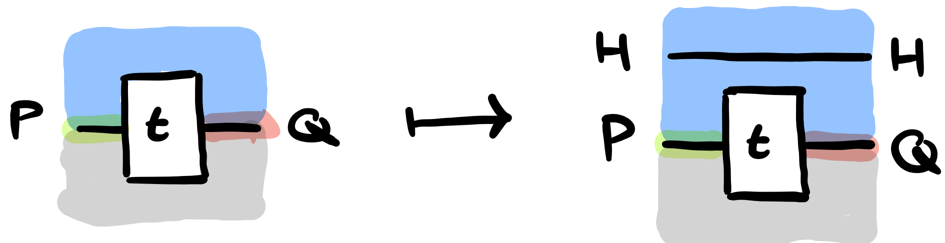
with a strength:

$$T(P,Q) \longrightarrow T(P^{H(I)}, H \times Q)$$

Beyond representable Markov categories?



- Need a notion of heteromorphism $\check{\mathcal{E}}^{op} \nrightarrow \hat{\mathcal{E}}$
- An equivalence in Tamb



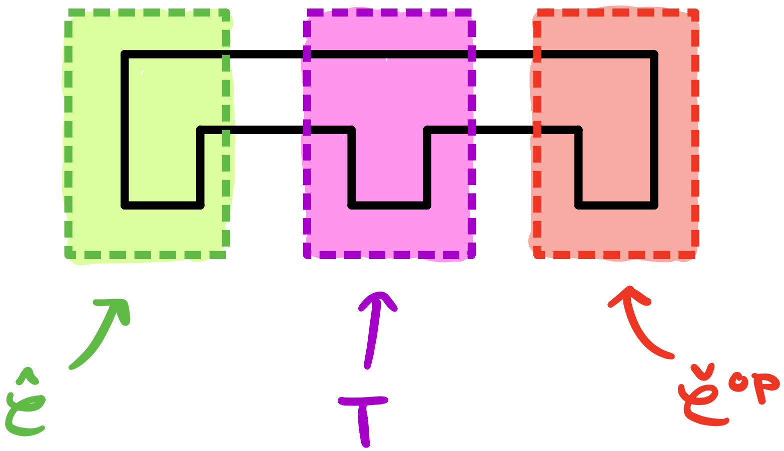
$$T : \check{\mathcal{E}} \times \hat{\mathcal{E}} \longrightarrow \mathrm{Set}$$

with a strength:

$$T(P, Q) \longrightarrow T(P^{H(I)}, H \times Q)$$

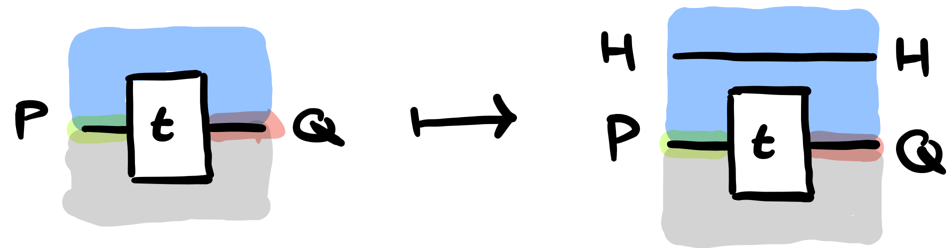
- same variance on P and Q, but different variance on actions

Beyond representable Markov categories?



- Need a notion of heteromorphism $\check{\mathcal{C}}^{op} \nrightarrow \hat{\mathcal{C}}$
- An equivalence in Tamb

$$T : \check{\mathcal{C}} \times \hat{\mathcal{C}} \longrightarrow \mathrm{Set}$$



with a strength:

$$T(P, Q) \longrightarrow T(P^{H(I)}, H \times Q)$$

- same variance on P and Q, but different variance on actions
- Conjecture: No such structure possible

Thank you!

slides at : dylanbraithwaite.github.io