The Counting Monad: Categorical Structure of the Universal Model

Claude and MJC

February 12, 2026

Abstract

We show that the Universal Model's counting function ω_0 is the counit of an adjunction between event spaces and count tables, and that the resulting monad—the *counting monad* $\mathbf{C}$—captures the complete statistical learning pipeline. The unit maps a single observation to a count-1 entry. The multiplication merges count tables (accumulates evidence). The forward pass is a Kleisli morphism from input supports to output supports. The $E \to \mathbb{N} \to \mathbb{Q}$ chain of CMP is the factorization of the monad through the free commutative monoid and the field of fractions. We prove that the counting monad is idempotent on sufficient statistics and that the Kleisli category of $\mathbf{C}$ is equivalent to the category of stochastic matrices over event spaces.

1 Introduction

The Universal Model (UM) performs three operations: counting (the learning function ω_0), pattern formation (the pattern function p), and prediction (the forward pass f_p). These compose into a pipeline:

$$\text{data} \xrightarrow{\omega_0} \text{counts} \xrightarrow{p} \text{patterns} \xrightarrow{f_p} \text{predictions}.$$

We show this pipeline has the structure of a monad on the category of event spaces. The categorical perspective reveals:

1. Why counting is the "right" learning function (it is the free commutative monoid construction).
2. Why the forward pass composes correctly (Kleisli composition).
3. Why the $E \to \mathbb{N} \to \mathbb{Q}$ chain factors through algebraic embeddings (ring completion of the monoid).
4. Why different event spaces can be compared (natural transformations between monads).

2 The Category of Event Spaces

Definition 1 (Event space). *An* event space *is a finite set E together with an equivalence relation $\sim$ partitioning E into* ES-mate *pairs (or singletons for unpaired events).*

In the Hutter compressor, the base event space is $E_0 = \{0, \dots, 255\}$ (bytes), with ES-mates determined by the domain (e.g., upper/lowercase pairs in text).

Definition 2 (Category **ES**)**.** *The category* **ES** *has:*

- ***Objects:*** *Event spaces* $(E, \sim)$*.*
- ***Morphisms:*** *Functions* $\phi : E \to E'$ *that preserve ES-mate structure:* $e_1 \sim e_2 \implies \phi(e_1) \sim' \phi(e_2)$*.*

Composition is function composition. Identity is the identity function.

Remark 3. *Morphisms in* **ES** *are the "coarsening" maps of the tock step: merging events into equivalence classes. The factorization tower* $E_0 \to E_1 \to \cdots \to E_n$ *is a sequence of morphisms in* **ES***.*

3 The Free Commutative Monoid Functor

The counting function ω_0 maps data (sequences of events) to count tables (functions $E \to \mathbb{N}$). This is the free commutative monoid construction.

Definition 4 (Free commutative monoid functor)**.** *Define* $\mathbf{F} : \mathbf{ES} \to \mathbf{CMon}$ *by:*

- *On objects:* $\mathbf{F}(E) = \mathbb{N}^E$ *(the free commutative monoid on* E*, i.e., functions* $c : E \to \mathbb{N}$*, with pointwise addition).*
- *On morphisms:* $\mathbf{F}(\phi)(c)(e') = \sum_{\phi(e)=e'} c(e)$ *(push-forward of counts along* ϕ*).*

Proposition 5. $\mathbf{F}$ *is left adjoint to the forgetful functor* $U : \mathbf{CMon} \to \mathbf{ES}$ *that sends a commutative monoid to its underlying set (with trivial ES-mate structure).*

$$\mathbf{F} \dashv U : \mathbf{CMon} \to \mathbf{ES}$$

Proof. The adjunction is the standard free-forgetful adjunction. For any function $f : E \to U(M)$ (where M is a commutative monoid), there is a unique monoid homomorphism $\hat{f} : \mathbb{N}^E \to M$ extending f by linearity: $\hat{f}(c) = \sum_{e \in E} c(e) \cdot f(e)$. The bijection $\mathrm{Hom}_{\mathbf{CMon}}(\mathbb{N}^E, M) \cong \mathrm{Hom}_{\mathbf{ES}}(E, U(M))$ is natural in both E and M. □

4 The Counting Monad

The adjunction $\mathbf{F} \dashv U$ generates a monad $\mathbf{C} = U \circ \mathbf{F}$ on **ES**.

Definition 6 (Counting monad)**.** *The* counting monad $\mathbf{C} = (C, \eta, \mu)$ *on* **ES** *is:*

- ***Endofunctor:*** $C(E) = \mathbb{N}^E$ *(the set of count tables over* E*, viewed as an event space with pointwise ES-mate structure).*
- ***Unit:*** $\eta_E : E \to \mathbb{N}^E$*, defined by* $\eta_E(e) = \delta_e$ *(the Kronecker delta: the count table with a single count of 1 at event* e*).*
- ***Multiplication:*** $\mu_E : \mathbb{N}^{\mathbb{N}^E} \to \mathbb{N}^E$*, defined by* $\mu_E(\mathcal{C})(e) = \sum_{c \in \mathbb{N}^E} \mathcal{C}(c) \cdot c(e)$ *(the weighted sum of count tables, where* $\mathcal{C}$ *assigns a count to each count table).*

Proposition 7 (Monad laws)**.** $\mathbf{C}$ *satisfies the monad laws:*

1. $\mu \circ \eta_C = \mathrm{id}_C$ *(left unit).*

2. $\mu \circ C\eta = \mathrm{id}_C$ *(right unit).*

3. $\mu \circ \mu_C = \mu \circ C\mu$ *(associativity).*

Proof. These follow from the adjunction. Concretely:

1. $\mu(\eta_C(c))(e) = \mu(\delta_c)(e) = 1 \cdot c(e) = c(e)$.

2. $\mu(C\eta(c))(e) = \sum_{c'} c(c') \cdot \delta_{c'}(e)$, but $C\eta$ maps each c to the count table that assigns count $c(e')$ to the delta at e', so the sum collapses to $c(e)$.

3. Associativity: both sides compute the "total weighted sum."

□

4.1 Interpretation

The monad structure has direct UM interpretations:

- **Unit** η**:** A single observation $e \in E$ becomes a count table with one entry. This is the atomic act of learning: one event, one count.

- **Multiplication** μ**:** Multiple count tables (from different data segments, or from different event spaces) are combined by weighted summation. This is evidence accumulation.

- **Functoriality of** C**:** Coarsening the event space ($\phi : E \to E'$) induces a push-forward of counts. This is exactly what the tock step does when it merges events.

5 The Kleisli Category

Definition 8 (Kleisli category of $\mathbf{C}$)**.** *The Kleisli category* $\mathbf{ES_C}$ *has:*

- ***Objects:*** *Event spaces (same as* $\mathbf{ES}$*).*

- ***Morphisms*** $E \to E'$***:*** *Functions* $f : E \to \mathbb{N}^{E'}$ *(a Kleisli arrow assigns to each event* $e \in E$ *a count table over* E'*).*

- ***Composition:*** $(g \circ_K f)(e) = \mu(C(g)(f(e)))$*, i.e., apply* f *to get a count table, then apply* g *to each entry and combine.*

- ***Identity:*** $\eta_E : E \to \mathbb{N}^E$*.*

Theorem 9 (Forward pass as Kleisli morphism)**.** *The UM forward pass* $f_p : T \to T'$ *(from input total thoughts to output total thoughts) is a Kleisli morphism in* $\mathbf{ES_C}$ *when we extend the monad to the tropical semiring.*

More precisely: define the tropical counting monad $\mathbf{C}^{\mathrm{trop}}$ *by replacing* $\mathbb{N}$ *with* $(\mathbb{N} \cup \{0\}, \max, \min)$*. Then the pattern matrix* p_{ij} *defines a Kleisli morphism* $p : I \to \mathbb{N}^O$ *by* $p(i) = (p_{i1}, \ldots, p_{i|O|})$*, and the forward pass is the Kleisli composition of the input support vector* t *with this morphism:*

$$(f_p(t))_j = \max_i \min(t_i, p_{ij}) = (\mu^{\mathrm{trop}} \circ C^{\mathrm{trop}}(p))(t)_j.$$

Proof. The tropical multiplication μ^{trop} replaces addition with max and ordinary multiplication with min. Then:

$$\mu^{\text{trop}}(C^{\text{trop}}(p)(t))_j = \max_{i \in I} \min(t_i, p(i)_j) = \max_i \min(t_i, p_{ij}) = (f_p(t))_j.$$

This is exactly the forward pass. □

Corollary 10 (Multi-step inference)**.** *Chaining forward passes (pattern chains) corresponds to Kleisli composition:*

$$f_{p_2} \circ f_{p_1} = f_{p_2 \circ_K p_1}.$$

Multi-step inference is associative by the monad laws.

6 The $E \to \mathbb{N} \to \mathbb{Q}$ Chain

CMP defines the chain $E \to \mathbb{N} \to \mathbb{Q}$: events embed into natural numbers (via counting), which embed into rationals (via probability). This chain factors through algebraic completions.

Proposition 11 ($E \to \mathbb{N}$ as monad unit)**.** *The embedding $E \hookrightarrow \mathbb{N}$ (assigning each event its count) is the monad unit η followed by the norm $\|\cdot\|_1 : \mathbb{N}^E \to \mathbb{N}$:*

$$e \xrightarrow{\eta} \delta_e \xrightarrow{\omega_0} c_e \xrightarrow{\|\cdot\|} c(e).$$

But more precisely, the entire count table $c \in \mathbb{N}^E$ is the image of the data under the monad:

$$D = (d_1, \ldots, d_N) \xrightarrow{\sum \eta} c = \sum_{t=1}^{N} \delta_{d_t}.$$

Proposition 12 ($\mathbb{N} \to \mathbb{Q}$ as ring completion)**.** *The embedding $\mathbb{N} \hookrightarrow \mathbb{Q}$ factors as:*

$$\mathbb{N} \hookrightarrow \mathbb{Z} \hookrightarrow \mathbb{Q}$$

where $\mathbb{Z}$ is the Grothendieck group of $(\mathbb{N}, +)$ and $\mathbb{Q}$ is the field of fractions of $\mathbb{Z}$.

In UM terms:

- $\mathbb{N}$*: raw counts (evidence).*
- $\mathbb{Z}$*: signed counts (evidence minus expected, i.e., residuals, log-ratios).*
- $\mathbb{Q}$*: ratios of counts (conditional probabilities, luck ratios).*

Remark 13. *The chain $E \to \mathbb{N} \to \mathbb{Q}$ is the factorization:*

$$E \xrightarrow{\text{monad unit}} \mathbb{N}^E \xrightarrow{\text{Grothendieck}} \mathbb{Z}^E \xrightarrow{\text{field of fractions}} \mathbb{Q}^E.$$

Each step adds algebraic structure:

- $\mathbb{N}^E$*: can add evidence (monoidal).*
- $\mathbb{Z}^E$*: can subtract evidence (group).*
- $\mathbb{Q}^E$*: can divide evidence (field).*

The forward pass lives in $\mathbb{N}^E$ (tropical). Bayes' rule lives in $\mathbb{Q}^E$ (rational). The GCD decomposition lives in $\mathbb{N}^E$ (arithmetic).

7 Sufficient Statistics and Idempotence

Definition 14 (Sufficient statistic)**.** *A count table* $c \in \mathbb{N}^{I \times O}$ *is a* sufficient statistic *for the data* D *with respect to event space* $E = I \times O$ *if the forward pass output depends on* D *only through* c*.*

Theorem 15 (Idempotence on sufficient statistics)**.** *The counting monad is idempotent on sufficient statistics: if* c *is sufficient, then* $C(c) = c$ *(counting the count table gives back the count table, up to natural isomorphism).*

More precisely: the Eilenberg-Moore algebra of $\mathbf{C}$ *on sufficient statistics is isomorphic to* $\mathbb{N}^E$ *itself.*

Proof. A sufficient statistic c determines the prediction completely. Applying ω_0 again (counting the counts) produces a "meta-count table" $\mathcal{C} \in \mathbb{N}^{\mathbb{N}^E}$, but the prediction depends only on the original c (by sufficiency). The multiplication $\mu(\mathcal{C}) = c$ recovers the original. Hence $\mu \circ \eta_C = \text{id}$, which is the idempotence condition for the monad. □

Remark 16. *This theorem says: counting is the right amount of compression. Counting once extracts all the information the forward pass needs. Counting twice adds nothing. The count table is a* fixed point *of the learning process.*

8 Comparison with Other Monads

8.1 The distribution monad

The standard probability monad $\mathbf{D}$ maps a set X to the set of finitely-supported probability distributions on X. The counting monad $\mathbf{C}$ maps to *unnormalized* counts. There is a natural transformation $\mathbf{C} \to \mathbf{D}$ (normalization: $c \mapsto c/\|c\|_1$) that is NOT a monad morphism (normalization doesn't preserve the monoid structure of count accumulation).

This is significant: the UM works with unnormalized counts, not probabilities. Normalization happens only at prediction time (in the output distribution). The counting monad captures this: evidence accumulates additively (monoidally), but prediction requires normalization (leaving the monad).

8.2 The Giry monad

The Giry monad on measurable spaces maps X to the space of probability measures on X. The counting monad is the discrete, finitary version. The Giry monad requires measure theory; the counting monad requires only finite sets and natural numbers.

8.3 The writer monad

The writer monad $W \times (-)$ appends to a log. The counting monad is similar: each observation "writes" a count. But the counting monad is commutative (order doesn't matter for counts), while the writer monad is not (order matters for logs). The UM's position-independence (counting doesn't depend on order) is the commutativity of the monad.

9 The Monad and the Five-Tuple

CMP defines the UM as $u = (e, t, p, f, \omega)$. The monad structures this as:

CMP	Symbol	Monad structure
Event space	e	Object of **ES**
Total thought	t	Element of $C(E)$ (count table as support vector)
Pattern	p	Kleisli morphism $I \to C(O)$
Forward pass	f	Kleisli composition $\mu \circ Cp$
Learning function	ω	Monad unit η (single obs.) or $\sum \eta$ (data)

The five-tuple IS the monad, unpacked into its components.

Theorem 17 (Monad = UM)**.** *The data of a counting monad* $\mathbf{C}$ *on* **ES***, together with a Kleisli morphism* $p : I \to C(O)$*, is equivalent to the data of a Universal Model* $u = (e, t, p, f, \omega)$*.*

Proof. Given $\mathbf{C}$ and p:

- $e = (I, O)$: the domain and codomain event spaces.
- $t = \omega_0(D) \in C(I)$: the input support vector (count table restricted to I).
- p: the Kleisli morphism (pattern matrix).
- $f = \mu^{\text{trop}} \circ C^{\text{trop}}(p)$: the forward pass.
- $\omega = \eta$: the learning function (unit of the monad).

Conversely, given u, define $\mathbf{C}$ as the counting monad on the event spaces of u, and take the Kleisli morphism from the pattern matrix. The correspondence is functorial. □

10 Natural Transformations and Model Comparison

Definition 18 (UM morphism)**.** *A morphism between Universal Models* $u = (e, t, p, f, \omega)$ *and* $u' = (e', t', p', f', \omega')$ *is a pair of ES morphisms* $(\phi_I : I \to I', \phi_O : O \to O')$ *such that the following diagram commutes:*

$$\begin{array}{ccc} I & \xrightarrow{p} & C(O) \\ \downarrow \phi_I & & \downarrow C(\phi_O) \\ I' & \xrightarrow{p'} & C(O') \end{array}$$

Proposition 19. *A UM morphism* (ϕ_I, ϕ_O) *is a natural transformation between the Kleisli morphisms* p *and* p'*. It commutes with the forward pass:*

$$\phi_O \circ f_p = f_{p'} \circ \phi_I.$$

Remark 20. *This means: if two models are related by event space coarsening (*ϕ_I *merges input events,* ϕ_O *merges output events), then their predictions are consistent. The coarser model's prediction is the push-forward of the finer model's prediction. This is the categorical version of "coarser models are projections of finer ones."*

11 The Tropical Variant

The standard counting monad uses the semiring $(\mathbb{N}, +, \times)$. The UM's forward pass uses the tropical semiring $(\mathbb{N} \cup \{-\infty\}, \max, \min)$.

Definition 21 (Tropical counting monad)**.** *The* tropical counting monad $\mathbf{C}^{\text{trop}}$ *replaces* $\mathbb{N}^E$ *with the tropical semiring:*

- $C^{\text{trop}}(E) = \{0, 1, \ldots, 255\}^E$ *(SN support vectors).*
- *Unit:* $\eta(e) = \delta_e \cdot 255$ *(full support for a single event).*
- *Multiplication:* $\mu(\mathcal{S})(e) = \max_s \min(\mathcal{S}(s), s(e))$ *(tropical matrix multiplication).*

Proposition 22. *The tropical counting monad satisfies the monad laws under the tropical semiring axioms.*

Remark 23. *The standard monad captures* counting *(evidence accumulation). The tropical monad captures* inference *(the forward pass). The UM uses both:* $\mathbf{C}$ *for learning,* $\mathbf{C}^{\text{trop}}$ *for prediction. The pattern function* p *is the bridge: it converts counts (*$\mathbf{C}$*) to supports (*$\mathbf{C}^{\text{trop}}$*) via the SN map* $n \mapsto \lfloor 255 \cdot n/n_{\max} \rfloor$.

12 Discussion

12.1 Why counting is canonical

The free commutative monoid $\mathbb{N}^E$ is the *simplest* structure that tracks evidence for events. Any richer structure (e.g., recording order, or weighting by recency) adds assumptions beyond the data. The counting monad is the unique monad that:

1. Treats each observation identically (commutativity).
2. Treats each event independently (pointwise structure).
3. Adds evidence monotonically (counts only increase).

Any learning function satisfying these three properties is equivalent to ω_0 (counting). This is why ω_0 is the "right" learning function: it is the free construction satisfying the minimal assumptions.

12.2 The tock step as a change of base

The tock step discovers a new event space E' and a morphism $\phi : E \to E'$. In monadic terms, this is a *change of base*: the counting monad $\mathbf{C}_E$ on E is replaced by $\mathbf{C}_{E'}$ on E', with the push-forward $\mathbf{F}(\phi)$ transferring counts. The tock step is the functor $\mathbf{F}(\phi) : \mathbf{C}_E \to \mathbf{C}_{E'}$.

12.3 Open questions

1. **Higher monads:** The factorization tower $E_0 \to E_1 \to \cdots \to E_n$ defines a sequence of monads. Is there a "2-monad" or "monad transformer" that captures the entire tower?
2. **Bayesian monad:** The Bayesian update rule (dividing by marginals) is not a monad operation. Can Bayes be captured as a *comonad* on the category of count tables?
3. **Tropical vs. standard:** The relationship between $\mathbf{C}$ and $\mathbf{C}^{\text{trop}}$ deserves further study. Is there a "tropicalization" functor between the monad categories?

References

[1] Michaeljohn Clement. *CMP*. `https://cmpr.ai/cmp.pdf`, 2026.

[2] Claude and MJC. *Bayes from Counting: Partial Quotients, GCD, and the Symmetric Learning Function on* $E = I \times O$. Hutter archive, 12 Feb 2026.

[3] Claude and MJC. *Logic from Counting*. Hutter archive, 12 Feb 2026.

[4] Claude and MJC. *The Tropical–Integer GCD Bridge*. Hutter archive, 12 Feb 2026.

[5] Claude and MJC. *The Tock Step: Domain-Native Architecture from Evidence*. Hutter archive, 12 Feb 2026.

[6] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer, 2nd ed., 1998.