Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

# The Neuron as a Latent State: Classical Variational Readout in Distributional Neural Units

**Yves Ruffenach**

yves@ruffenach.net

Conservatoire National des Arts et Métiers https://orcid.org/0009-0009-4737-0555

**Research Article**

**Keywords:** Variational neural networks, distributional neurons, latent-state computation, uncertainty estimation, probabilistic readout, Monte Carlo prediction, neural uncertainty

**Posted Date:** June 30th, 2026

**DOI:** https://doi.org/10.21203/rs.3.rs-10177882/v1

**License:** This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

**Additional Declarations:** The authors declare no competing interests.

# The Neuron as a Latent State: Classical Variational Readout in Distributional Neural Units

Yves Ruffenach[1*]

[1*]Conservatoire National des Arts et Métiers, Strasbourg, France.

Corresponding author(s). E-mail(s): yves@ruffenach.net;

**Abstract**

Classical neural networks usually represent hidden units as deterministic scalar activations. This makes computation easy to compose, but it collapses each unit's internal state into a single value, so uncertainty is usually modeled outside the hidden unit through Bayesian weights, ensembles, output distributions, or post-hoc predictive scores. We study an alternative view with EVE, a classical distributional neural mechanism in which hidden computation carries a local variational latent state. This state is parameterized by a posterior mean and variance, sampled through reparameterization, and regularized by a local Kullback–Leibler divergence. We focus not only on whether such a state can be learned, but on how it should be read. To isolate this question, we introduce classical variational readout and use a posterior-freezing protocol: after training EVE, the learned posterior is fixed and only the readout rule is varied. On a controlled multimodal forecasting benchmark, the frozen EVE posterior represents multiple plausible futures more effectively than a deterministic baseline. A non-oracle branch-medoid readout further improves CRPS and pinball loss, removes invalid samples, improves branch balance, and preserves coverage and best-of-sample accuracy without posterior retraining. These results support the interpretation of EVE as an information-bearing latent computational state rather than merely stochastic noise or regularization.

**Keywords:** Variational neural networks, distributional neurons, latent-state computation, uncertainty estimation, probabilistic readout, Monte Carlo prediction, neural uncertainty

# 1 Introduction

Modern neural networks are usually built from point-valued hidden units. Given an input representation, a neuron computes a scalar activation and passes this single value to the next layer. This interface has been extremely effective, but it also collapses the internal state of the unit into one number. As a result, uncertainty, variability, and reliability are usually modeled outside the hidden unit itself, through Bayesian weights, stochastic regularization, ensembles, output distributions, or post-hoc uncertainty scores.

This paper studies a different view: the neuron as a local latent state. We analyze EVE, a classical distributional neural mechanism in which a hidden computation is equipped with a local variational posterior,

$$q_\phi(z \mid h) = \mathcal{N}\left(\mu_\phi(h), \mathrm{diag}(\sigma^2_\phi(h))\right).$$

The latent state is sampled through reparameterization,

$$z = \mu_\phi(h) + \sigma_\phi(h) \odot \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I),$$

and regularized by a local Kullback–Leibler divergence. The resulting unit remains fully classical and differentiable, but its internal computation is no longer point-valued. It becomes a state that can be sampled, diagnosed, and read.

This changes the prediction problem. If a hidden computation carries a posterior rather than a scalar activation, prediction is not only a matter of learning that posterior. It is also a matter of choosing how to read it. The same learned state may be summarized by a Monte Carlo mean, a median, a robust aggregation rule, an uncertainty-conditioned rule, or a structure-aware readout. These choices can produce different predictions even when the posterior is unchanged.

We call this problem *classical variational readout*. To isolate it, we use a posterior-freezing protocol. We train an EVE model once, select a checkpoint on validation data, and then freeze all posterior parameters. We then apply multiple readout rules to the same frozen latent state. If a readout improves over naive Monte Carlo aggregation, the gain cannot be attributed to additional training. It must come from predictive structure already present in the learned posterior.

The resulting diagnostic is the *readout gap*: the performance difference between naive Monte Carlo aggregation and alternative readouts applied to the same frozen posterior. A positive readout gap means that the posterior contains useful predictive information that the naive readout does not fully extract.

We evaluate this hypothesis on a controlled branching-future forecasting benchmark. The task is deliberately multimodal: each observed prefix can lead to several valid future trajectories. This setting exposes a weakness of point-valued prediction, because averaging several valid futures can produce an invalid one. It also provides a direct test of whether EVE learns a structured posterior and whether this posterior can be read more effectively than by naive aggregation.

The experiments show two main results. First, the frozen EVE posterior represents the multimodal future structure better than a deterministic baseline, improving coverage, best-of-sample accuracy, CRPS, and NLL. Second, a non-oracle branch-aware readout improves over the raw Monte Carlo set without retraining the posterior. The branch-medoid readout improves CRPS and pinball loss, removes invalid samples, improves branch balance, and preserves coverage and best-of-sample accuracy.

The contribution of this paper is therefore not simply another stochastic neural predictor. It is the separation of two problems that are usually coupled: learning a latent state and reading that state. This separation makes it possible to test whether a distributional neuron behaves merely as a stochastic regularizer or as an information-bearing computational state.

Our contributions are:

1. We formulate EVE as a classical latent-state mechanism for hidden neural computation.
2. We introduce classical variational readout: the problem of extracting predictions from a fixed learned posterior.
3. We propose a posterior-freezing protocol that separates state learning from state reading.
4. We define the readout gap as an operational diagnostic of whether a frozen posterior contains exploitable predictive structure beyond naive aggregation.
5. We show on a controlled multimodal benchmark that a non-oracle branch-aware readout improves over raw Monte Carlo aggregation without retraining the posterior.

The purpose of this paper is not to present EVE as another Bayesian neural network in the usual parameter-space sense, nor as a stochastic output head. EVE is instead a hidden-state distributional mechanism: its posterior is local to the hidden computation, and its readout is a separate design choice. This supports a view of hidden representations not only as point-valued transformations, but as latent states whose predictive content depends on how they are read.

# 2 Related Work

## 2.1 Uncertainty outside the hidden unit

Bayesian neural networks represent uncertainty by placing probability distributions over model parameters. Bayes by Backprop learns an approximate posterior over weights with a variational objective [1], while variational dropout and the local reparameterization trick improve the scalability of stochastic variational training [2]. These methods are closely related to EVE through variational inference, reparameterized sampling, and KL regularization. However, their uncertainty is primarily parameter-space uncertainty. EVE instead attaches a posterior to the hidden computation itself: the stochastic object is a local latent state conditioned on the current representation, not a distribution over global weights.

Other widely used uncertainty methods preserve the usual point-valued hidden representation. Monte Carlo dropout obtains approximate Bayesian prediction by sampling stochastic forward passes [3], and deep ensembles estimate uncertainty through a population of independently trained models [4]. These approaches are strong predictive baselines, but the stochasticity lives in dropped activations, weights, or model populations. EVE differs by exposing a structured posterior inside the computation, making the hidden state itself sampleable, diagnosable, and readable after training.

## 2.2 Latent variables and stochastic hidden representations

Variational latent-variable models provide the technical foundation for differentiable stochastic computation. The variational autoencoder introduced amortized inference, reparameterized sampling, and stochastic gradient optimization for continuous latent variables [5]. EVE uses the same broad toolkit, but at a different granularity: its latent variable is local to the hidden computation and participates directly in the predictive pathway.

EVE is also related to stochastic hidden representation methods, including variational information bottleneck models that regularize intermediate representations by preserving task-relevant information while discarding nuisance variation [6]. The difference is conceptual and methodological. In this paper, the latent state is not used only as a regularized representation. It is treated as a learned posterior that can be frozen and read in multiple ways. The readout gap then tests whether this posterior contains predictive structure beyond naive averaging.

## 2.3 Output-level distributional uncertainty

A large family of methods models uncertainty directly at the output level. Heteroscedastic regression predicts an input-dependent variance and captures aleatoric uncertainty [7]. Evidential regression predicts parameters of a higher-order uncertainty distribution, often avoiding sampling at inference time [8]. Mixture density networks parameterize conditional mixture distributions and are a classical approach to multimodal regression [9].

EVE is complementary to these methods. It does not only parameterize a target distribution or output variance. It exposes a latent posterior inside the hidden computation, and the same frozen posterior can be mapped to predictions through different readout rules. This location of the distributional object is the key distinction: the present work studies how internal uncertainty can be used as a computational state, not only as an output score.

## 2.4 Probabilistic evaluation and reliability

Because EVE produces stochastic predictions through sampled latent states, evaluation must include both point accuracy and probabilistic quality. Negative log-likelihood evaluates the probability assigned to observed targets, while proper scoring rules such as CRPS assess full predictive distributions [10]. Quantile-based losses, including pinball loss, evaluate predictive quantiles and distributional forecasts [11].

Reliability is also central to the latent-state view. Selective prediction studies the trade-off between risk and coverage, allowing models to abstain or defer on examples estimated to be unreliable [12, 13]. For EVE, this is especially relevant because posterior variance, sample dispersion, branch disagreement, and invalid-sample rates are internal quantities. If such diagnostics rank predictive risk, then the posterior is not only stochastic but operationally informative.

### 2.5 Readout and posterior summaries

The closest conceptual motivation for this paper is the observation that a latent state is useful only through a readout. Standard neural networks hide this issue because scalar activations have an implicit readout: the scalar itself. Distributional units expose the design choice. A posterior can be read by its mean, median, robust aggregation, validation-guided statistics, uncertainty-conditioned selection, or a structured function of samples.

This perspective is related to posterior predictive summaries and decision-theoretic prediction, where a predictive distribution is mapped to an action through a loss-dependent rule. Our goal is different. We do not derive a Bayes-optimal rule for a known task loss. We empirically test whether a fixed hidden posterior contains readout-dependent structure. After training, the posterior is frozen, and the only remaining degree of freedom is the classical rule used to read it.

This positions EVE between variational neural computation, uncertainty estimation, stochastic representation learning, probabilistic forecasting, and reliability analysis. Its distinctive feature is not merely that it produces uncertainty, but that uncertainty is internal to the hidden computation and can be transformed into predictions through an explicit readout protocol.

## 3 Methods

This section formalizes EVE as a local latent-state mechanism for hidden computation and defines the posterior-freezing readout protocol used in the experiments. The goal is to separate two questions that are usually coupled in stochastic neural models: how the latent state is learned, and how this state is read after training.

### 3.1 From scalar activations to latent-state units

Consider a standard feedforward neural network layer. Given an incoming representation $h \in \mathbb{R}^d$, a classical hidden unit computes a deterministic pre-activation and returns a scalar activation,

$$a = \rho(w^\top h + b),$$

where $\rho$ is a nonlinearity. This scalar value is then passed to the next layer. The internal state of the unit is therefore collapsed into one deterministic number.

EVE replaces this purely point-valued interface by a local variational latent state. Instead of directly emitting a point activation, an EVE module parameterizes a

posterior distribution

$$q_\phi(z \mid h) = \mathcal{N}\left(\mu_\phi(h), \mathrm{diag}(\sigma_\phi^2(h))\right),$$

where $\mu_\phi(h)$ and $\sigma_\phi(h)$ are produced by differentiable functions of the incoming representation. A latent activation is sampled through the reparameterization trick,

$$z = \mu_\phi(h) + \sigma_\phi(h) \odot \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I).$$

The sampled latent state $z$ is then passed to a downstream decoder or prediction head. In this sense, the hidden computation is no longer only a point-valued transformation. It is a local computational state that can be sampled, summarized, diagnosed, and read out.

## 3.2 EVE predictive model

Let $x$ denote the input and $y$ the target. An EVE model first maps $x$ to a hidden representation $h_\theta(x)$, then constructs a latent posterior

$$q_\phi(z \mid x) = q_\phi(z \mid h_\theta(x)).$$

A prediction is obtained by decoding a latent sample,

$$\hat{y}^{(s)} = g_\psi(z^{(s)}, h_\theta(x)), \qquad z^{(s)} \sim q_\phi(z \mid x),$$

where $s = 1, \ldots, S$ indexes Monte Carlo samples. The set

$$\mathcal{Y}_S(x) = \left\{\hat{y}^{(1)}, \ldots, \hat{y}^{(S)}\right\}$$

is the sampled predictive state exposed by EVE. Different readouts can be applied to this same set without changing the learned posterior.

The default Monte Carlo mean readout is

$$r_{\mathrm{mean}}(\mathcal{Y}_S) = \frac{1}{S}\sum_{s=1}^{S} \hat{y}^{(s)}.$$

However, the mean is only one possible way to extract a decision from the latent state. In multimodal settings, averaging valid futures can produce an invalid or unrepresentative prediction. This motivates studying readout as an explicit design choice rather than treating Monte Carlo averaging as the only natural output of the model.

## 3.3 Training objective

During training, the EVE posterior is learned jointly with the encoder and decoder. The basic objective combines a predictive loss with a local variational regularizer,

$$\mathcal{L}_{\mathrm{EVE}} = \mathcal{L}_{\mathrm{pred}} + \beta D_{\mathrm{KL}}\left(q_\phi(z \mid x) \,\|\, p(z)\right),$$

where $p(z) = \mathcal{N}(0, I)$ is a simple prior and $\beta$ controls the strength of the KL penalty. The predictive term can include point accuracy and probabilistic scoring components depending on the task.

For structured multimodal forecasting, the training objective may also include auxiliary terms that encourage sample diversity, branch coverage, geometric validity, and temporal coherence. These terms are used only to learn a useful posterior. They are not changed during the readout comparison. This distinction is important: the experiments do not compare readouts by retraining EVE, but by applying different readout functions to the same frozen posterior.

## 3.4 Posterior-freezing protocol

The central methodological step is posterior freezing. After training, we select a checkpoint using validation criteria and then freeze all parameters of the EVE model:

$$\theta, \phi, \psi \quad \text{fixed.}$$

For every validation or test input $x$, we sample from the fixed posterior to obtain a predictive set $\mathcal{Y}_S(x)$. All readout rules are then evaluated on this same learned state. Therefore, if two readouts produce different performance, the difference cannot be attributed to additional representation learning or posterior updates. It must come from the way the fixed latent state is read.

This protocol isolates the readout effect. It turns the EVE posterior into an object of analysis: a learned latent state that can be queried through multiple classical readout rules.

## 3.5 Classical variational readouts

A readout is a function

$$r : \mathcal{Y}_S(x) \mapsto \tilde{y}$$

for point prediction, or

$$R : \mathcal{Y}_S(x) \mapsto \widetilde{\mathcal{Y}}(x)$$

for distributional prediction. We consider both types.

**Point readouts.** Point readouts collapse the sampled posterior state into a single prediction. Examples include the Monte Carlo mean, coordinate-wise median, trimmed mean, density-weighted mean, validation-fitted linear combinations of posterior statistics, and quadratic observable-style readouts. These readouts test how much information is lost when a multimodal latent state is reduced to one trajectory.

**Distributional readouts.** Distributional readouts preserve a predictive set. They can filter, rebalance, or select samples while keeping the output multimodal. Examples include the raw Monte Carlo set, validity-filtered sets, branch-balanced sets, branch-medoid sets, quality-balanced sets, and validation-selected non-oracle readouts. These readouts are especially relevant when the target distribution is multimodal and no single mean prediction is representative.

### 3.6 Branch-aware readout

The main structure-aware readout used in the experiments is the branch-medoid readout. It is designed for settings where the posterior samples cluster around several plausible futures. Given a sample set $\mathcal{Y}_S(x)$, the readout first groups or assigns samples according to their branch-like trajectory structure. Within each group, it selects a representative medoid sample, namely a sample with small average distance to the other samples in the same group. The resulting set preserves the multimodal structure while removing redundant or geometrically inconsistent samples.

Importantly, this readout is non-oracle: it does not use the test target. It depends only on the frozen posterior samples and on validation-selected readout rules. Therefore, any improvement over the raw Monte Carlo set provides evidence that the posterior contains exploitable structure beyond naive aggregation.

### 3.7 Readout gap

The main diagnostic quantity is the readout gap. For a metric $M$ where lower is better, such as CRPS or pinball loss, the gap of a readout $R$ relative to the raw Monte Carlo set is

$$\Delta_M(R) = M(\text{raw MC}) - M(R).$$

A positive value means that $R$ extracts more useful predictive information from the same frozen posterior than naive Monte Carlo aggregation. For CRPS, this becomes

$$\Delta_{\text{CRPS}}(R) = \text{CRPS}(\text{raw MC}) - \text{CRPS}(R).$$

The relative gap is

$$\Delta^{\%}_{\text{CRPS}}(R) = 100\frac{\text{CRPS}(\text{raw MC}) - \text{CRPS}(R)}{\text{CRPS}(\text{raw MC})}.$$

This quantity is the operational test of the latent-state hypothesis. If a frozen posterior supports a non-oracle readout with a positive readout gap, then the posterior contains predictive structure not fully extracted by the naive mean or raw Monte Carlo readout.

### 3.8 Validation selection and non-oracle constraint

To avoid test-set leakage, deployable readouts are selected using validation data only. The test set is used only once to report the final performance of the selected readout. We distinguish deployable readouts from oracle diagnostics. Oracle readouts may use the test target and are reported only as upper bounds; they are not considered evidence of a deployable readout rule.

This distinction ensures that the reported readout gap is attributable to a legitimate readout of the frozen posterior, not to target-conditioned selection at test time.

### 3.9 Reliability and selective prediction

Because the EVE posterior exposes sample dispersion and structural diagnostics, it can also be used to rank uncertainty. Let $u(x)$ be a posterior-derived uncertainty score, such as sample variance, invalid-sample rate, branch disagreement, or temporal switching. A selective predictor retains only the fraction of examples with the lowest uncertainty scores and measures the risk on this retained subset. If risk decreases as coverage decreases, the posterior diagnostics are informative about prediction reliability.

Selective prediction therefore provides a second test of the latent-state view. It asks not only whether the posterior can generate accurate samples, but whether internal posterior statistics help identify when the model is likely to be reliable.

## 4 Experiments and Results

We now instantiate the posterior-freezing protocol on a controlled multimodal forecasting benchmark. The goal is to test whether the fixed EVE posterior contains branch-structured predictive information that can be extracted by non-oracle readouts. After training, the posterior is frozen and all readout comparisons are performed without further updating the posterior parameters.

### 4.1 Controlled branching-future benchmark

We evaluate EVE on a controlled branching-future forecasting benchmark, denoted CHBF. Each example contains a common observed prefix and four plausible future continuations. This setting is deliberately designed to expose a failure mode of point-valued deterministic predictors: averaging over several valid futures can produce a future trajectory that is itself invalid. The benchmark therefore tests whether a model can represent several plausible futures and whether a readout can extract a valid prediction or predictive set from the learned posterior.

The dataset uses two-dimensional trajectories. Each input consists of $t_{\text{obs}} = 12$ observed time steps and each target future has horizon $t_{\text{future}} = 48$. For each prefix, four branch futures are generated. We use 640 base examples for training, 160 for validation, and 160 for testing; with four branches per base example, this corresponds to 2560 training rows, 640 validation rows, and 640 test rows. The branch noise standard deviation is 0.018, the branch curvature parameter is 0.12, and the branch amplitude range is $[0.95, 1.35]$.

### 4.2 Models and training protocol

We compare a deterministic multilayer perceptron, denoted DET, against an EVE model with a local variational latent state. Both models use hidden width 128 and depth 3. EVE uses latent dimension $k = 8$ and a decoder of depth 2. DET has 48,608 trainable parameters and EVE has 118,648 trainable parameters in the final run.

The EVE model is trained with a curriculum that first encourages distributional coverage, then geometric validity, and finally temporal coherence. The selected posterior is the joint checkpoint, chosen to balance best-of-sample accuracy, branch

**Table 1** CHBF benchmark configuration.

| Quantity | Value |
|---|---|
| Observed prefix length | 12 |
| Future horizon | 48 |
| Trajectory dimension | 2 |
| Number of branches | 4 |
| Base train / validation / test examples | 640 / 160 / 160 |
| Expanded train / validation / test rows | 2560 / 640 / 640 |
| Branch noise standard deviation | 0.018 |
| Branch curvature | 0.12 |
| Branch amplitude range | $[0.95, 1.35]$ |

coverage, validity, and coherence. After selection, the posterior is frozen. All readout results below use this same frozen posterior.

For probabilistic evaluation, the deterministic baseline uses a calibrated Gaussian output scale. EVE is evaluated from Monte Carlo samples of the frozen posterior. In the final run, the calibrated output scale is 0.5165 for DET and 0.0758 for EVE.

## 4.3 Readout protocols

We evaluate two families of readouts.

First, point-collapse readouts convert the predictive sample set into a single future trajectory. These include the Monte Carlo mean, Monte Carlo median, trimmed mean, density-weighted mean, a learned linear statistics readout, and a quadratic observable-style point readout. These baselines test what is lost when a multimodal posterior is collapsed into one trajectory.

Second, distributional readouts preserve a predictive set. The raw Monte Carlo set is the naive posterior readout. We compare it to branch-aware and validity-aware set readouts, including a branch-medoid set, branch-balanced set, quality-balanced set, ridge-guided branch set, validity-filtered set, time-consistent filtered set, and an oracle best-of-sample upper bound. The oracle uses the test target and is reported only as a non-deployable upper bound. Its set-level branch coverage and branch-balance diagnostics are therefore not directly comparable to deployable readouts. The publication champion is selected on validation among deployable non-oracle readouts and then evaluated on the test set.

## 4.4 Metrics

We report mean squared error (MSE), best-of-sample MSE, negative log-likelihood (NLL), continuous ranked probability score (CRPS), pinball loss, branch coverage, branch probability $\ell_1$ error, temporal coherence, and invalid trajectory rate. Lower is better for MSE, best-of-sample MSE, NLL, CRPS, pinball loss, branch $\ell_1$, and invalid rate. Higher is better for coverage and coherence.

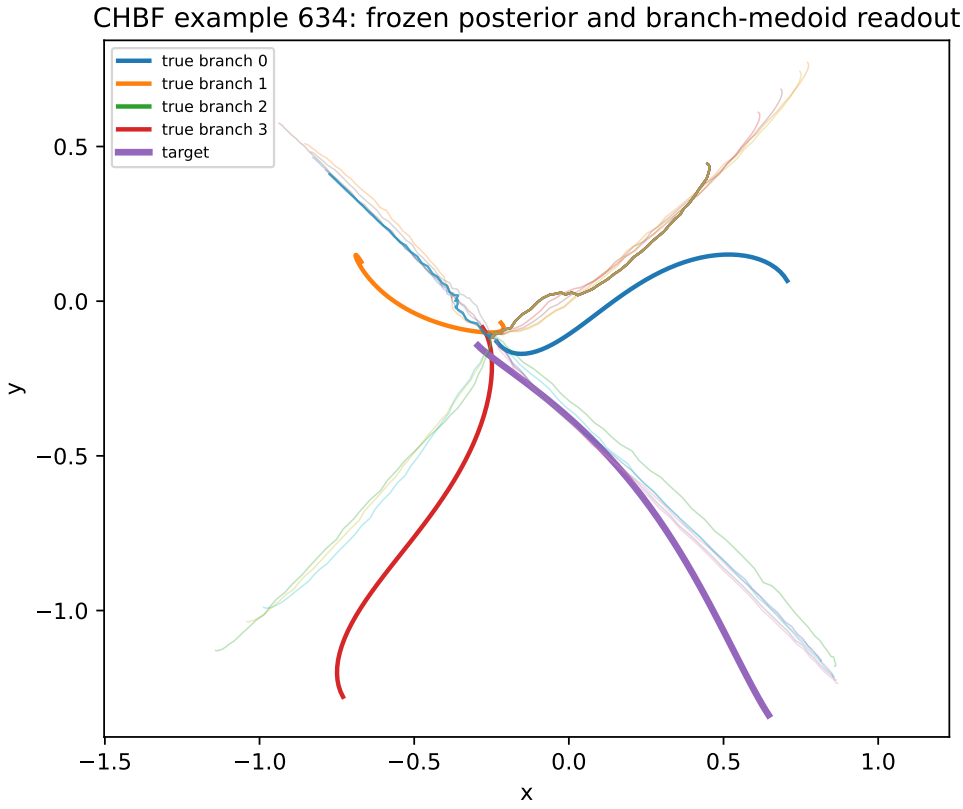



**Fig. 1** Qualitative CHBF example. The frozen EVE posterior contains multiple plausible future trajectories, while the branch-medoid readout extracts a structured subset without retraining the posterior.

The central metric for the paper is the readout gap. For a readout $r$, we define the CRPS readout gap relative to the raw Monte Carlo set as

$$\Delta_{\mathrm{CRPS}}(r) = \mathrm{CRPS}(\text{raw MC}) - \mathrm{CRPS}(r).$$

A positive gap means that the readout extracts predictive structure from the same frozen posterior that is not fully captured by naive Monte Carlo aggregation.

## 4.5 DET versus frozen EVE posterior

Table 2 compares DET with the raw frozen EVE posterior. DET achieves slightly lower mean point MSE, but this metric rewards the deterministic average trajectory and does not reflect multimodal validity in CHBF. EVE strongly improves the distributional and structural metrics: coverage increases from 0.25 to 1.00, best-of-sample MSE decreases from 0.2499 to 0.0051, CRPS decreases from 0.4081 to 0.2352, and NLL decreases from

**Table 2** DET baseline versus frozen EVE posterior on CHBF test data.

| Model | Mean MSE | Best-S MSE | Coverage | Branch $\ell_1$ | Coherence | Invalid | NLL | CRPS |
|---|---|---|---|---|---|---|---|---|
| DET | 0.2499 | 0.2499 | 0.25 | 1.5000 | 0.0000 | 1.0000 | 69.76 | 0.4081 |
| EVE raw MC set | 0.2575 | 0.0051 | 1.00 | 0.1202 | 0.6580 | 0.0062 | -111.58 | 0.2352 |

**Table 3** Point-collapse readouts on the frozen EVE posterior.

| Readout | Point MSE | Invalid rate |
|---|---|---|
| MC mean | 0.2575 | 1.0000 |
| Trimmed mean | 0.2586 | 1.0000 |
| MC median | 0.4242 | 0.0953 |
| Density-weighted mean | 0.3211 | 0.6422 |

69.76 to -111.58. This indicates that the frozen EVE posterior contains the plausible future branches, even though its naive mean readout is not the best way to use the posterior.

## 4.6 Point-collapse readouts

Point-collapse readouts confirm that the main experimental object should be the posterior set rather than a single averaged trajectory. Table 3 reports representative point readouts. The Monte Carlo mean obtains MSE 0.2575 but has invalid rate 1.0000. The trimmed mean behaves similarly, with MSE 0.2586 and invalid rate 1.0000. The median improves validity but worsens point accuracy, with MSE 0.4242 and invalid rate 0.0953. Density weighting increases the valid rate but also worsens MSE, with MSE 0.3211 and invalid rate 0.6422. Thus, collapsing a multimodal posterior into a single trajectory does not extract the useful structure of the posterior in a reliable way.

## 4.7 Distributional readout gap

Table 4 reports the main result. The raw Monte Carlo set is the naive frozen-posterior readout. The best deployable readout selected on validation is the branch-medoid set. On test, this readout improves CRPS from 0.2352 to 0.2230, improves pinball loss from 0.1056 to 0.1009, preserves best-of-sample MSE at 0.0051, preserves coverage at 1.00, reduces branch probability $\ell_1$ error from 0.1202 to 0.0000, and reduces invalid rate from 0.0062 to 0.0000. The CRPS readout gap is 0.0122, corresponding to a 5.20% relative improvement over raw Monte Carlo aggregation.

This improvement is obtained without retraining the EVE posterior. The gain therefore comes from the readout rule applied to the same learned latent state. The result supports the hypothesis that the EVE posterior contains exploitable predictive structure beyond what is extracted by naive Monte Carlo aggregation. We emphasize that the best deployable readout in this experiment is structure-aware rather than

**Table 4** Distributional readout gap on the frozen EVE posterior. The oracle is shown only as a non-deployable upper bound and is not directly comparable on set-level branch diagnostics.

| Readout | CRPS | Pinball | Best-S MSE | Mean MSE | Coverage | Branch $\ell_1$ | Coherence | Invalid |
|---|---|---|---|---|---|---|---|---|
| Branch-medoid set | **0.2230** | **0.1009** | 0.0051 | **0.2519** | 1.00 | **0.0000** | 0.6590 | **0.0000** |
| Ridge-guided branch set | 0.2248 | 0.1018 | 0.0051 | 0.2532 | 1.00 | 0.0000 | 0.6772 | 0.0000 |
| Branch-balanced set | 0.2264 | 0.1023 | 0.0053 | 0.2537 | 1.00 | 0.0000 | 0.7795 | 0.0000 |
| Quality-balanced set | 0.2272 | 0.1026 | 0.0054 | 0.2538 | 1.00 | 0.0000 | 0.7786 | 0.0001 |
| Raw MC set | 0.2352 | 0.1056 | 0.0051 | 0.2575 | 1.00 | 0.1202 | 0.6580 | 0.0062 |
| Oracle upper bound | 0.0726 | 0.0334 | 0.0051 | 0.0290 | – | – | – | – |

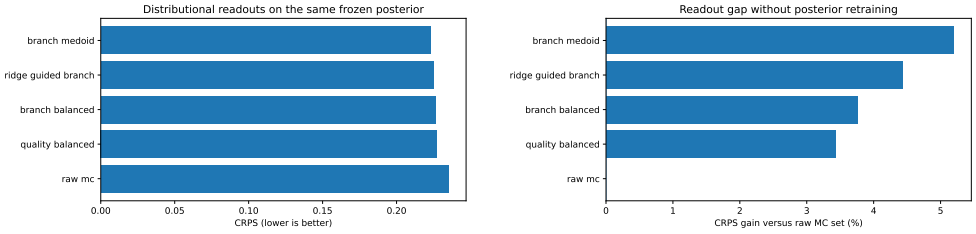



**Fig. 2** Distributional readout gap on the same frozen EVE posterior. Left: CRPS for deployable readouts. Right: relative CRPS improvement over the raw Monte Carlo set. The branch-medoid readout is selected on validation and improves over naive Monte Carlo aggregation on test.

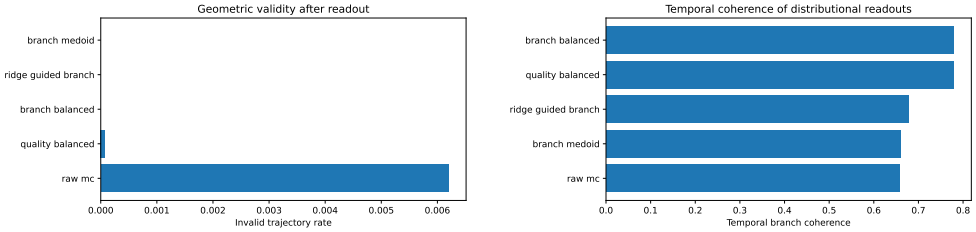



**Fig. 3** Structural diagnostics for frozen-posterior readouts. Branch-aware readouts reduce invalid samples and can increase temporal coherence while preserving multimodal coverage.

learned: learned validation readouts are included in the ablation suite but do not dominate the branch-medoid rule in this run.

## 4.8 Bootstrap confidence intervals

We estimate confidence intervals by bootstrap resampling over the test examples. The raw Monte Carlo set obtains CRPS 0.2352 with 95% confidence interval $[0.2323, 0.2383]$. The branch-medoid set obtains CRPS 0.2230 with 95% confidence interval $[0.2210, 0.2254]$. The intervals do not overlap, supporting the robustness of the readout gap. The ridge-guided branch set also improves over raw Monte Carlo, with CRPS 0.2248 and 95% confidence interval $[0.2228, 0.2272]$.

**Table 5** Bootstrap confidence intervals for the main readouts.

| Readout | Metric | Mean | 95% low | 95% high |
|---|---|---|---|---|
| Raw MC set | CRPS | 0.2352 | 0.2323 | 0.2383 |
| Branch-medoid set | CRPS | **0.2230** | **0.2210** | **0.2254** |
| Ridge-guided branch set | CRPS | 0.2248 | 0.2228 | 0.2272 |
| Raw MC set | Best-S MSE | 0.0051 | 0.0047 | 0.0056 |
| Branch-medoid set | Best-S MSE | 0.0051 | 0.0047 | 0.0056 |

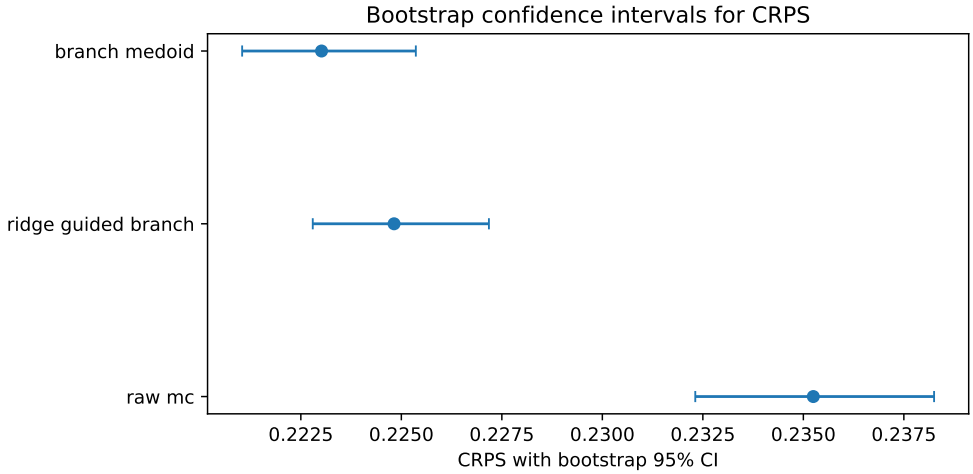



**Fig. 4** Bootstrap confidence intervals for CRPS. The branch-medoid readout improves over the raw Monte Carlo set with non-overlapping 95% confidence intervals.

## 4.9 Selective prediction

We also evaluate whether internal posterior statistics can rank predictive risk. On the raw Monte Carlo set, selecting the 50% lowest-risk examples according to invalid sample rate reduces best-of-sample risk from 0.0051 at full coverage to 0.0038. At 70% coverage, the risk is 0.0037, and at 90% coverage it is 0.0041. This indicates that posterior-derived diagnostics are informative for selective prediction. Other uncertainty scores, such as one-minus-majority, switch rate, sample standard deviation, and sample interquartile range, are less effective in this run but still provide a structured way to evaluate reliability from the frozen posterior.

## 4.10 Summary

The CHBF experiment provides evidence for the latent-state interpretation of EVE. The raw EVE posterior represents the multimodal future structure much better than DET, as shown by coverage, best-of-sample MSE, CRPS, and NLL. More importantly,

the same frozen posterior supports different readouts with different predictive quality. A non-oracle branch-aware readout improves over naive Monte Carlo aggregation on CRPS, pinball loss, invalid rate, and branch balance without changing the posterior. This positive readout gap is the main experimental evidence that the EVE posterior is not merely stochastic noise or regularization, but an information-bearing computational state whose useful content depends on the readout applied to it.

The experiment also clarifies the limitation of the current result. Learned validation readouts and point-collapse readouts do not dominate the simple branch-medoid rule. The present evidence therefore supports the narrower claim that a structured non-oracle readout can extract useful information from a frozen EVE posterior, rather than the stronger claim that all learned readouts improve over Monte Carlo aggregation.

# 5 Discussion

The experiments support the central hypothesis of this paper: the EVE posterior can behave as an information-bearing latent state rather than merely as stochastic noise or regularization. The key evidence is the readout gap. Different readouts applied to the same frozen posterior produce different predictive quality, even though the posterior parameters are unchanged.

This distinction is important. If the improvement came from retraining, it would be unclear whether the gain came from a better posterior or a better readout. Posterior freezing removes this ambiguity. The model first learns a latent state; the readout then determines how this state is converted into a prediction or predictive set. Standard scalar neurons hide this distinction because the activation is already its own readout. Distributional neurons expose it.

The strongest deployable readout in this experiment is the branch-medoid set. This indicates that the useful information in the posterior is not captured only by global sample moments. Point-collapse rules such as the mean, median, or trimmed mean can destroy multimodal structure by reducing several plausible futures to one trajectory. In contrast, the branch-medoid readout preserves branch structure while selecting representative and geometrically valid samples.

The result should be interpreted narrowly but clearly. The CHBF benchmark is controlled and synthetic, so it does not establish broad empirical superiority across real-world datasets. It does show that a frozen EVE posterior can contain predictive structure that naive Monte Carlo aggregation leaves unused. The strongest readout is also structure-aware rather than learned, suggesting that generic learned summaries of posterior statistics may be insufficient. Future readouts should preserve posterior geometry, enforce validity, and remain robust under distribution shift.

The broader implication is that uncertainty inside a network can be treated as a computational state, not only as an output score. The readout gap provides a practical test for this view: if no readout improves over naive aggregation, the posterior may function mostly as regularization. If a non-oracle readout improves while the posterior remains fixed, then the posterior contains exploitable structure.

# 6 Conclusion

This paper studied EVE, a classical variational mechanism that equips hidden neural computation with a local latent state. Instead of exposing only a deterministic scalar activation, EVE exposes a posterior that can be sampled, diagnosed, and read.

The central claim is that prediction with such a unit has two parts: learning the latent state and reading it. We isolated the second part through a posterior-freezing protocol. EVE is trained once, its posterior is fixed, and multiple readout rules are applied to the same learned state.

On the CHBF benchmark, the frozen EVE posterior represents multimodal future structure better than a deterministic baseline. More importantly, a non-oracle branch-aware readout improves over raw Monte Carlo aggregation without retraining the posterior. This positive readout gap shows that useful predictive structure exists inside the learned latent state beyond its naive mean.

The conclusion is simple: when a hidden unit carries a posterior, its predictive content depends on how that posterior is read. EVE is therefore not only a stochastic neural mechanism, but a way to study hidden computation as an information-bearing latent state.

# Declarations

**Competing interests.** The author declares no competing interests.

**Funding.** No external funding was received for this work.

**Data availability.** The data and code used to generate the reported results are available from the author upon reasonable request.

**Author contributions.** Y.R. conceived the study, implemented the experiments, analyzed the results, and wrote the manuscript.

# References

[1] Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural networks. In: Proceedings of the 32nd International Conference on Machine Learning, pp. 1613–1622 (2015)

[2] Kingma, D.P., Salimans, T., Welling, M.: Variational dropout and the local reparameterization trick. In: Advances in Neural Information Processing Systems, vol. 28 (2015)

[3] Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Proceedings of the 33rd International Conference on Machine Learning, pp. 1050–1059 (2016)

[4] Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[5] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (2014)

[6] Alemi, A.A., Fischer, I., Dillon, J.V., Murphy, K.: Deep variational information bottleneck. In: International Conference on Learning Representations (2017)

[7] Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[8] Amini, A., Schwarting, W., Soleimany, A., Rus, D.: Deep evidential regression. In: Advances in Neural Information Processing Systems, vol. 33, pp. 14927–14937 (2020)

[9] Bishop, C.M.: Mixture density networks. Technical Report NCRG/94/004, Neural Computing Research Group, Aston University (1994)

[10] Gneiting, T., Raftery, A.E.: Strictly proper scoring rules, prediction, and estimation. Journal of the American Statistical Association **102**(477), 359–378 (2007)

[11] Koenker, R., Bassett, G.: Regression quantiles. Econometrica **46**(1), 33–50 (1978)

[12] Geifman, Y., El-Yaniv, R.: Selective classification for deep neural networks. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[13] Geifman, Y., El-Yaniv, R.: SelectiveNet: A deep neural network with an integrated reject option. In: Proceedings of the 36th International Conference on Machine Learning, pp. 2151–2159 (2019)