



# Aleatoric vs Epistemic Uncertainty

Uncertainty quantification 01 | Noise in the world vs ignorance of the model: definitions, heteroscedastic regression with a Gaussian negative log-likelihood, the law-of-total-variance split computed on an ensemble, what each kind tells you to do, and where the split breaks down

#Uncertainty #Aleatoric #Epistemic #Heteroscedastic #UQ

## Two reasons not to be sure

A weather forecaster in a new city

Uncertainty in a model's prediction

Summer storms are hard to time even with perfect data  
Aleatoric: noise in the phenomenon itself

She has only two weeks of local records  
Epistemic: too little data about this region

More years of records make her sure about the climate, not about Tuesday's shower  
Data removes epistemic, never aleatoric

Asked about a city she has never seen, she should say so  
Far from the training data, epistemic uncertainty must grow

WHERE THE ANALOGY BREAKS

A forecaster knows when she is in an unfamiliar city; a neural network does not, unless its uncertainty method is built to notice distance from the training data.

A weather forecaster explains both kinds of doubt

Before Separating aleatoric and epistemic uncertainty, can you explain...

for any regressor or classifier

|                                                                                | Not yet                  | Partly                   | Yes                      |
|--------------------------------------------------------------------------------|--------------------------|--------------------------|--------------------------|
| <b>MUST KNOW</b> Variance, expectation, Gaussian density<br>Learn: maths 08-09 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <b>MUST KNOW</b> Entropy in nats<br>Learn: maths 11                            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <b>HELPFUL</b> Posterior and prior<br>Learn: machine-learning 15               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <b>HELPFUL</b> Bias-variance trade-off<br>Learn: maths 14                      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <b>ADVANCED</b> BALD for active learning<br>Learn: learning-paradigms 07       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

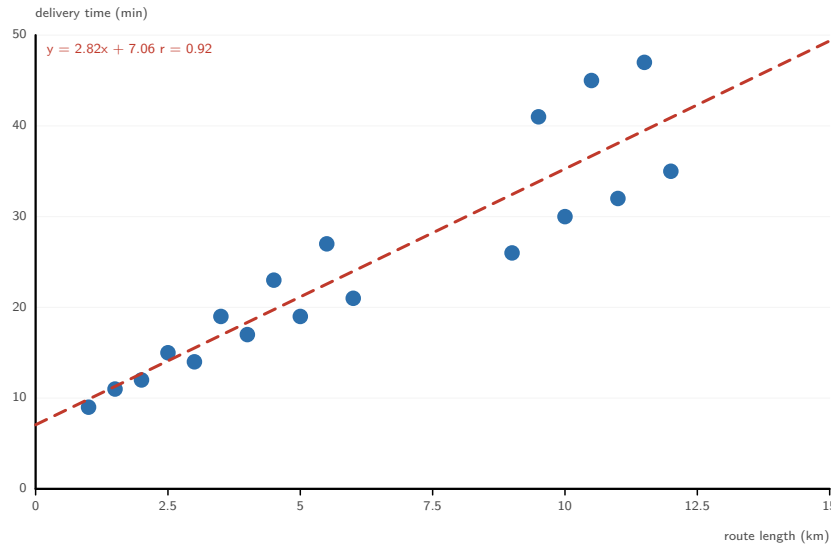
What to know first (one-line recalls below, never re-taught here)

Recall. Entropy  $H(p) = -\sum p \log p$  measures spread of a discrete distribution (maths 11); the posterior  $p(\theta \mid D)$  weights parameter values by how well they explain the data (machine-learning 15). Everything else on this sheet is new.

| Question                              | Aleatoric                                                    | Epistemic                                                  |
|---------------------------------------|--------------------------------------------------------------|------------------------------------------------------------|
| Where does it come from?              | noise, ambiguity, missing features in the data               | limited data, unknown parameters, unfamiliar inputs        |
| Does more data of the same kind help? | no: the floor stays                                          | yes: it shrinks roughly like 1/n near the data             |
| Where is it large?                    | where the target is intrinsically noisy                      | far from training points, in sparse regions                |
| How is it modelled?                   | the likelihood: predict $\sigma^2(x)$ or class probabilities | a distribution over models: ensembles, Bayesian posteriors |
| What do you do about it?              | accept it, add features or better sensors, price the risk    | collect data there, abstain, defer to a human              |
| Other names                           | data uncertainty, irreducible                                | model uncertainty, knowledge uncertainty                   |

The two kinds side by side

## Seeing both in one dataset



Toy delivery data (minutes vs km). Spread grows with distance (heteroscedastic aleatoric noise); no trips between 6 and 9 km and none beyond 12 km, so any model there is guessing (epistemic)

| Quantity          | Formula                                                                                                 | Reads                                                                                            |
|-------------------|---------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| Model output      | $f_{\theta}(x) = (\mu_{\theta}(x), \sigma_{\theta}^2(x))$                                               | two heads: mean and variance (variance through softplus or exp to stay > 0)                      |
| Gaussian NLL loss | $\ell = \frac{1}{2} \log \sigma_{\theta}^2(x) + \frac{(y - \mu_{\theta}(x))^2}{2 \sigma_{\theta}^2(x)}$ | big errors are forgiven where $\sigma^2$ is large, but log $\sigma^2$ charges for claiming noise |
| Optimal variance  | $\sigma^{*2}(x) = \mathbb{E}[(y - \mu(x))^2   x]$                                                       | minimising $\ell$ over $\sigma^2$ gives the local mean squared error                             |

where  $x, y$  = input and observed target (feature units; target units);  $\theta$  = network weights (-);  $\mu_{\theta}(x)$  = predicted mean of  $y$  (target units);  $\sigma_{\theta}^2(x)$  = predicted aleatoric variance of  $y$  at  $x$  (target units squared);  $\ell$  = loss for one example (constant  $\frac{1}{2} \log 2\pi$  dropped) (nats)

Heteroscedastic regression: let the network predict its own noise

Mean-variance network in PyTorch (GaussianNLLLoss expects the variance, not the std)

```
import torch, torch.nn as nn, torch.nn.functional as F
class MeanVar(nn.Module):
    def __init__(self, d_in, h=64):
        super().__init__()
        self.body = nn.Sequential(nn.Linear(d_in, h), nn.ReLU(),
                                   nn.Linear(h, h), nn.ReLU())
        self.mu, self.raw_var = nn.Linear(h, 1), nn.Linear(h, 1)
    def forward(self, x):
        z = self.body(x)
        return self.mu(z), F.softplus(self.raw_var(z)) + 1e-6
model, loss_fn = MeanVar(8), nn.GaussianNLLLoss()
mu, var = model(x_batch)
loss = loss_fn(mu, y_batch, var) # aleatoric sigma^2(x) = var
```

### Training pitfall

With the plain NLL the network can explain hard points away with a large  $\sigma^2$  and stop fitting the mean there. Common fixes: warm up on MSE for the mean first, then switch to NLL; or weight each example's NLL by  $\sigma^{-2}(2\beta)$  with  $\beta \approx 0.5$  (the  $\beta$ -NLL of Seitzer et al., 2022). A single mean-variance network still gives **no** epistemic uncertainty: far from the data it extrapolates  $\sigma^2(x)$  as confidently as  $\mu(x)$ .

## Splitting total uncertainty

### PROBLEM

Four mean-variance networks predict a 7.5 km route (inside the 6-9 km gap). Means  $\mu = 30, 34, 31, 33$  min; variances  $\sigma^2 = 4.0, 5.0, 4.4, 4.6$  min<sup>2</sup>. Split the uncertainty.

#### 1. Average mean

$\Rightarrow \mu? = (30 + 34 + 31 + 33) / 4 = 32$  min  
why: the ensemble's point prediction

#### 2. Aleatoric part

$\Rightarrow (4.0 + 5.0 + 4.4 + 4.6) / 4 = 4.5$  min<sup>2</sup>  
why: average noise each member expects

#### 3. Epistemic part

$\Rightarrow ((-2)^2 + 2^2 + (-1)^2 + 1^2) / 4 = 10 / 4 = 2.5$  min<sup>2</sup>  
why: members disagree on the mean in the data gap

#### 4. Total

$\Rightarrow 4.5 + 2.5 = 7.0$  min<sup>2</sup>,  $\sigma = \sqrt{7.0} = 2.65$  min  
why: variances add, standard deviations do not

#### 5. Share

$\Rightarrow 2.5 / 7.0 = 36\%$  epistemic  
why: a third of the doubt would shrink with trips of 6-9 km

### GIVEN

- $M = 4$  members;  $\mu_m$  in minutes;  $\sigma^2_m$  in min<sup>2</sup>
- Law of total variance over  $M$  models:  
 $\text{Var}_{\text{total}} = (1/M) \sum \sigma^2_m$  (aleatoric)  
+  $(1/M) \sum (\mu_m - \mu?)^2$  (epistemic)
- Members can be ensemble members, MC-dropout passes or posterior samples (sheets 02-03)

### ANSWER

**$32 \pm 2.65$  min (1  $\sigma$ ); 64 % aleatoric, 36 % epistemic: log more 6-9 km trips to cut the epistemic third.**

Check: Repeat at 3 km inside dense data: means 14, 14.5, 14, 13.5 give epistemic 0.125 min<sup>2</sup>; with an aleatoric part of about 4 min<sup>2</sup> that is 3 % of the total. Epistemic should be small where data is dense.

Four ensemble members predict one delivery: which doubt dominates?

| Term      | Formula                                               | Reads                                                  |
|-----------|-------------------------------------------------------|--------------------------------------------------------|
| Total     | $H[\bar{p}(y   x)], \bar{p} = \frac{1}{M} \sum_m p_m$ | entropy of the averaged prediction                     |
| Aleatoric | $\frac{1}{M} \sum_m H[p_m(y   x)]$                    | average entropy of each model: confident models give 0 |
| Epistemic | Total – Aleatoric = $I(y; \theta   x, D)$             | $\geq 0$ ; zero when all models agree                  |

where  $p_m(y | x)$  = class probabilities of model m (0-1);  $M$  = number of sampled models (count);  $H[p]$  = entropy - $\sum p \log p$  (nats);  $I$  = mutual information between the label and the model parameters (nats)  
 Classification version: entropy instead of variance (the epistemic term is the mutual information that BALD ranks by, learning-paradigms 07)

☉ Using the split, and its limits

| Conditions                                       | R1 | R2 | R3 | R4 |
|--------------------------------------------------|----|----|----|----|
| Aleatoric high                                   | Y  | Y  | N  | N  |
| Epistemic high                                   | Y  | N  | Y  | N  |
| Actions                                          |    |    |    |    |
| Predict normally                                 |    |    |    | X  |
| Widen the interval or price the risk             | X  | X  |    |    |
| Collect labels near this input (active learning) | X  |    | X  |    |
| Abstain or defer to a human                      | X  |    | X  |    |
| Add features or better sensors                   |    | X  |    |    |

What to do once you know which uncertainty is high

|                                                                                                                                                |    |                                                                          |
|------------------------------------------------------------------------------------------------------------------------------------------------|----|--------------------------------------------------------------------------|
| <b>Aleatoric</b><br>property of the data-generating process<br>Tell: Would 10× more data of the same kind remove it?                           | vs | <b>Epistemic</b><br>property of the model and its training set           |
| <b>Epistemic uncertainty</b><br>spread across plausible models<br>Tell: Shift is a cause; high epistemic uncertainty is one symptom (sheet 06) | vs | <b>Distribution shift</b><br>inputs differ from training data            |
| <b>Predicted <math>\sigma^2(x)</math></b><br>one network's noise estimate<br>Tell: Does it grow away from the data?                            | vs | <b>Predictive variance</b><br>noise plus disagreement across models      |
| <b>Uncertainty estimate</b><br>a number per prediction<br>Tell: Has it been checked on held-out data (sheet 05)?                               | vs | <b>Calibrated uncertainty</b><br>numbers that match observed frequencies |

Notions that are easy to confuse

ⓘ Where the clean split breaks down

The split is **model-relative** (Hüllermeier and Waegeman, 2021 review): a label that looks noisy with three features may be predictable with ten, so 'aleatoric' noise becomes reducible once a feature is added. The entropy decomposition also mixes the two when M is small, and neither term captures **misspecification** (a model family that cannot fit the truth). Treat the split as a guide for action, not as a law of nature.

📄 Kendall and Gal (2017) in computer vision

What uncertainties do we need in Bayesian deep learning for computer vision? (NeurIPS 2017) trained segmentation and depth networks with both a predicted variance (aleatoric) and MC dropout (epistemic). Aleatoric uncertainty was high on object boundaries and far-away pixels; epistemic uncertainty was high on objects rare in the training set. The paper popularised the mean-variance + sampling recipe used on this sheet.

MYTH OR FACT?

“Softmax probability measures epistemic uncertainty”

MYTH: One softmax mixes both and is often over-confident far from the data  
 Epistemic needs several plausible models (sheets 02-03).

MYTH OR FACT?

“Aleatoric noise is the same everywhere”

MYTH: It often depends on x  
 Heteroscedastic models predict  $\sigma^2(x)$ ; a constant  $\sigma$  hides risky regions.

MYTH OR FACT?

“More data always reduces uncertainty”

MYTH: Only the epistemic part  
 The aleatoric floor stays unless you add new information (features).

MYTH OR FACT?

“High uncertainty means the model is bad”

MYTH: It may be honest  
 An honest 'I am unsure here' is what downstream decisions need.

Common misreadings

Tick a box only when you could explain the point to someone else



Name the uncertainty before fixing it

aleatoric → features or risk pricing; epistemic → data or abstention



Model aleatoric with the likelihood

mean-variance heads, Gaussian NLL



Model epistemic with several models

ensembles, MC dropout, Bayesian posteriors



Add variances, not standard deviations

total = mean  $\sigma^2$  + variance of means



Check the numbers on held-out data

calibration and coverage (sheets 04-05)

Q1 Aleatoric vs epistemic in one line each?

A Aleatoric: noise in the data, irreducible with more of the same data. Epistemic: the model's ignorance, shrinks with relevant data.

Q2 Gaussian NLL for one example?

A  $\frac{1}{2} \log \sigma^2 + (y - \mu)^2 / (2\sigma^2)$ , with  $\mu$  and  $\sigma^2$  predicted by the network.

Q3 Means 30, 34, 31, 33; variances 4.0, 5.0, 4.4, 4.6. Epistemic part?

A Variance of the means around 32:  $(4 + 4 + 1 + 1) / 4 = 2.5 \text{ min}^2$ .

**Q4 Same example: aleatoric and total?**

**A** Aleatoric =  $4.5 \text{ min}^2$ ; total =  $7.0 \text{ min}^2$ ,  $\sigma \approx 2.65 \text{ min}$ .

**Q5 Classification: epistemic term?**

**A** Entropy of the averaged prediction minus the average entropy of each model (mutual information).

**Q6 Why does one mean-variance network give no epistemic uncertainty?**

**A** It is a single model; far from the data it extrapolates  $\sigma^2(x)$  without any sign of ignorance.

**Q7 Epistemic high, aleatoric low: what to do?**

**A** Abstain or defer now, and collect labels near such inputs.

**Q8 Aleatoric high, epistemic low: what to do?**

**A** Widen the interval or price the risk; add features or better sensors if the noise matters.

**Q9 Why is the split model-relative?**

**A** Noise unexplained by today's features can become explainable with new features: aleatoric depends on what the model sees.