Introductory Lecture: Methods Overview and Taxonomy

ELG 5218 - Uncertainty Evaluation in Engineering Measurements and Machine Learning

Miodrag Bolic

University of Ottawa

January 14, 2026

Course Overview

- **Focus**: Uncertainty Quantification (UQ) in Machine Learning & Deep Learning with some engineering aspects
- **Goal**: Understand sources, methods, and applications of uncertainty estimation
- **Topics**:
  - Types of uncertainty (aleatoric vs. epistemic)
  - Major UQ method categories
  - Advanced topics: Physics-informed ML, Generative models, State-space methods
- **Practical**: Implementation, calibration, evaluation metrics

About the course

This course is about:

- Bayesian approach
  - If a system can learn efficiently with a small amount of data $\Rightarrow$ strong modelling assumptions
- Deep learning with uncertainty
- Problems that engineers have when dealing with ML:
  - noise
  - small data
  - data is not independent
  - uncertainty

Aleatoric Uncertainty (Noise)

Definition: Irreducible randomness inherent in the data and measurement process.

Sources:

1. **Noise in Data / Measurement Error**
   - Sensor noise (LIDAR, camera measurement errors)
   - Environmental variations (weather, lighting)
   - Cannot be reduced with more data
2. **Label Ambiguity / Labeling Noise**
   - Subjective labeling (e.g., medical diagnosis disagreement)
   - Multiple valid answers for same input
3. **Missing Information / Occlusion**
   - Occluded/hidden objects, unmeasured variables
   - Incomplete features or temporal history

Key Property: *Irreducible* - persists even with infinite data

Epistemic Uncertainty (Model Uncertainty)

Definition: Reducible uncertainty due to lack of knowledge or incomplete model specification.

Sources:

1. **Limited Training Data / Small Dataset**
   - Few training samples, high-dimensional inputs
   - Data sparsity in input space
2. **Wrong or Incomplete Model Architecture**
   - Underspecified model (too simple)
   - Missing interactions or nonlinearity
3. **Parameters Cannot be Learned from Data**
   - Non-identifiability: multiple weight configs fit equally well
   - Flat likelihood: data doesn't constrain all parameters
4. **Distribution Shift / Domain Shift**
   - Covariate shift, out-of-distribution inputs
   - Domain adaptation challenges

Uncertainty Decomposition and Implication

Total Uncertainty = Aleatoric + Epistemic

Epistemic	**Aleatoric**	**Interpretation**	**Action**
High	Low	Need more data	Collect data / Active learning
Low	High	Task inherently ambiguous	Set safety margins / Wider intervals
Low	Low	High confidence	Proceed with confidence
High	High	Unknown unknowns	Reject or request human review

Practical Decision Making:

- **High Epistemic** → *Slow down, collect more data*
- **High Aleatoric** → *Predict well, set wider intervals*
- **Both Low** → *Confidence, proceed*
- **Both High** → *Reject, human intervention*

Taxonomy

Uncertainty Quantification Methods: Taxonomy

INFERENCE (Bayesian Methods)

Bayesian Linear Regression

Type: Epistemic
Requires: Prior on weights, Gaussian likelihood
Changes: Loss → Posterior computation
Output: Predictive distribution p(y|x,D)
Generative: No

Bayesian Neural Networks (BNN)

Type: Epistemic
Requires: Prior on all weights, VI or MCMC
Changes: Loss → ELBO or sampling
Output: Distribution over weights → predictions
Generative: No

Gaussian Processes (GP)

Type: Epistemic
Requires: Kernel, hyperparameters
Changes: Non-parametric → covariance function
Output: Posterior mean + variance
Generative: No

SAMPLING & ENSEMBLE METHODS (Multi-Pass)

Deep Ensembles

Type: Epistemic (disagreement)
Requires: M independently trained models
Changes: Train M models with random init
Output: Mean + variance of M predictions
Key: Gold standard for predictive UQ
Generative: No

Monte Carlo (MC) Dropout

Type: Epistemic
Requires: Dropout layers active at test time
Changes: Multiple forward passes (T passes)
Output: Variance across T stochastic passes
Key: Approx. BNN via Bernoulli dropout
Generative: No

Bootstrap Methods

Type: Epistemic (sampling variability)
Requires: Resampled datasets
Changes: Train on subsets/resamples
Output: Confidence intervals from samples
Key: Classic statistical approach
Generative: No

DIRECT / SINGLE-PASS DETERMINISTIC METHODS

Heteroscedastic Regression

Type: Aleatoric (Data Noise)
Requires: Dual head (μ, log σ^2)
Changes: Loss → Gaussian NLL
Output: Predicted variance σ^2(x)
Uncertainty: Input-dependent noise
Single-Pass: YES

Quantile Regression

Type: Aleatoric
Requires: Pinball loss for quantiles
Changes: Loss → Quantile loss
Output: Prediction intervals [q_low, q_high]
Uncertainty: Distribution-free intervals
Single-Pass: YES

Evidential Deep Learning (EDL)

Type: Epistemic + Aleatoric (Higher-Order)
Class: Single-Pass Deterministic / Posterior Network
Theory: Subjective Logic / Dempster-Shafer
Changes: Softmax → ReLU/Softplus output
Output: Dirichlet parameters α (higher-order dist)
Uncertainty: Decomposition via mutual information
Single-Pass: YES (Direct parametrization)

POST-HOC METHODS (No Model Change)

Conformal Prediction

Type: Neither (Distribution-free coverage)
Requires: Calibration set
Changes: Wrapper around ANY predictor
Output: Prediction sets with guaranteed coverage
Key: Valid regardless of model quality
Generative: No

Temperature Scaling

Type: Neither (Calibration only)
Requires: Validation set
Changes: Scale logits by scalar T
Output: Calibrated probabilities
Key: Simple, effective for confidence
Generative: No

Isotonic Regression

Type: Neither (Calibration only)
Requires: Validation set
Changes: Fit monotonic mapping
Output: Calibrated probabilities
Key: Non-parametric calibration
Generative: No

GENERATIVE MODELS

Variational Autoencoders (VAE)

Type: Both (latent epistemic + output aleatoric)
Requires: Encoder, decoder, ELBO loss
Changes: Architecture + Loss → variational bound
Output: Latent q(z|x), samples p(x|z)
Uncertainty: Latent space variance
Generative: YES

Diffusion Models (DDPM)

Type: Both (epistemic in denoising, aleatoric output)
Requires: Forward noising, U-Net denoiser
Changes: Loss → denoising score matching
Output: Samples from p(x), iterative sampling
Uncertainty: Sample diversity, score variance
Generative: YES

DIFFUSION UNCERTAINTY DETAILS

Forward: q(x_t | x_{t-1}) Gaussian noise

Reverse: p_θ(x_{t-1} | x_t) learned

UQ: Generate M samples → variance

Conditional: Guide with y → p(x|y)

Use: Image/signal generation + forecasting

STATE-SPACE MODELS & SEQUENTIAL FILTERING

Kalman Filter / EKF / UKF

Type: Both (process=aleatoric, state=epistemic)
Requires: Linear dynamics, Gaussian noise
Changes: State-space formulation x_t = Fx_{t-1} + w
Output: Posterior p(x_t | y_{1:t}), covariance P_t
Uncertainty: Covariance matrix evolution
Generative: No (filtering)

Particle Filter (Sequential MC)

Type: Both (process + state uncertainty)
Requires: Nonlinear dynamics, importance sampling
Changes: Sequential sampling with resampling
Output: Particle distribution over states
Uncertainty: Particle spread, effective sample size
Generative: No (filtering)

Hidden Markov Models (HMM)

Type: Epistemic (latent state uncertainty)
Requires: Discrete states, transition matrix
Changes: Forward-backward algorithm
Output: p(state | observations)
Uncertainty: Posterior state distribution
Generative: YES (can sample sequences)

PHYSICS-INFORMED MACHINE LEARNING

Physics-Informed Neural Networks (PINNs)

Type: Epistemic (parameter + model uncertainty)
Requires: PDE residual, physics constraints
Changes: Loss = L_data + λ·L_physics
Output: Solution u(x,t) + uncertainty via ensemble
Uncertainty: Ensemble of PINNs or MC Dropout
Classification: Physics-constrained supervised learning
Generative: No

Neural Operators (FNO, DeepONet)

Type: Epistemic
Requires: Operator learning, functional space
Changes: Loss on function-to-function map
Output: Function-to-function map + uncertainty
Uncertainty: Ensemble of operators
Classification: Operator learning (supervised)
Generative: No

PINN UNCERTAINTY QUANTIFICATION

Loss Components:

1. Data loss: MSE on observed points

2. Physics loss: PDE residual $||R(u)||^2$

3. Boundary conditions loss

UQ Methods:

• Ensemble of PINNs (different inits)

• MC Dropout on PINN

SEQUENTIAL DECISION MAKING UNDER UNCERTAINTY

POMDPs (Partially Observable MDPs)

Type: Epistemic + Aleatoric (state + outcome)
Requires: State space S, actions A, observations O, transition T, observation Z, reward R
Changes: Planning over belief states b_t
Output: Policy π(b_t) and value estimates V(b_t)
Uncertainty: Belief distribution over hidden state
Generative: YES (simulated trajectories)

Bayesian RL / Thompson Sampling

Type: Epistemic (model/parameter)
Requires: Prior over dynamics or value function
Changes: Sampled models/policies for exploration
Output: Policy with exploration driven by uncertainty
Uncertainty: Posterior over model or value estimates
Generative: YES (simulate rollouts under sampled models)

Major Categories of UQ Methods

Inference (Bayesian Methods)

- Bayesian Linear Regression, Bayesian Neural Networks, Gaussian Processes

Sampling & Ensemble Methods (Multi-Pass)

- Deep Ensembles, MC Dropout, Bootstrap, Snapshot Ensembles

Direct / Single-Pass Deterministic

- Heteroscedastic Regression, Quantile Regression, Evidential Deep Learning

Post-hoc Methods (No Model Change)

- Conformal Prediction, Temperature Scaling, Isotonic Regression

Generative Models

- VAE, Normalizing Flows, Diffusion Models

State-Space & Sequential

- Kalman Filter, Particle Filter, Hidden Markov Models

Physics-Informed ML

- Physics-Informed Neural Networks (PINNs), Neural Operators

Sequential Decision Making

- Reinforcement learning

Bayesian Linear Regression

Type: Epistemic Uncertainty

Core Idea:

- Place a prior distribution over weights: $p(\mathbf{w})$
- Compute posterior given data: $p(\mathbf{w}|\mathcal{D})$
- Output: Predictive distribution $p(y|\mathbf{x}, \mathcal{D})$

Advantages:

- Closed-form posterior (Gaussian)
- Interpretable weight uncertainty
- Uncertainty decreases with data

Disadvantages:

- Limited to linear models
- Does not scale to deep networks

Output: Mean prediction + variance (credible intervals)

Bayesian Neural Networks (BNN)

Type: Epistemic Uncertainty

Core Idea:

- Place prior over ALL weights $p(\mathbf{W})$
- Approximate posterior via Variational Inference (VI) or MCMC
- Sample from posterior at test time: $\hat{y} \sim \int p(y|\mathbf{x}, \mathbf{W})p(\mathbf{W}|\mathcal{D})d\mathbf{W}$

Advantages:

- Principled uncertainty quantification
- Works with deep networks
- Clear epistemic/aleatoric separation

Disadvantages:

- Computationally expensive (requires VI or MCMC)
- Posterior approximation quality varies

Output: Distribution over predictions

Gaussian Processes (GP)

Type: Epistemic Uncertainty

Core Idea:

- Non-parametric Bayesian method
- Define distribution over functions via kernel
- Posterior: $p(f|\mathcal{D}) \sim \mathcal{GP}(\mu(x), k(x, x'))$

Advantages:

- Exact posterior (no approximation)
- Principled uncertainty quantification
- Closed-form predictions with variance

Disadvantages:

- Scales as $O(n^3)$ in data size
- Limited to relatively small datasets
- Difficult to extend to high dimensions

Output: Mean + variance predictions

Deep Ensembles

Type: Epistemic Uncertainty

Core Idea:

- Train M independent models with different random initializations [2]
- At test time: get predictions from all M models
- Uncertainty from disagreement: $\mathrm{Var}[\hat{y}] = \frac{1}{M}\sum_i(\hat{y}_i - \bar{y})^2$

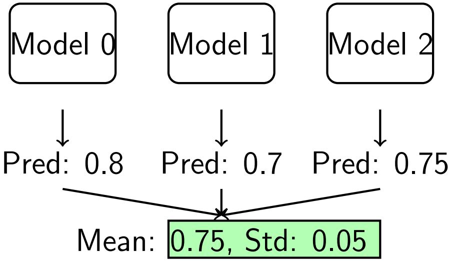
Advantages:

- Simple, embarrassingly parallel
- Gold standard for predictive uncertainty
- No model modification required
- Works with any architecture

Disadvantages:

- M$\times$ computational cost
- Requires multiple training runs

Output: Mean + variance of M predictions



Monte Carlo Dropout

Type: Epistemic Uncertainty

Core Idea:

- Keep dropout active at test time (not just training)
- Perform T stochastic forward passes
- Uncertainty from prediction variance across T passes

Key Insight: Dropout approximates Bayesian Neural Network [1]

Advantages:

- Single training run
- Minimal code changes (just enable dropout at test)

Disadvantages:

- Requires T forward passes at test (costly)
- Quality depends on dropout placement

Output: Mean + variance from T stochastic passes

Heteroscedastic Regression

Type: Aleatoric Uncertainty

Core Idea:

- Network has TWO output heads: $\mu(\mathbf{x})$ and $\log \sigma^2(\mathbf{x})$
- Learn input-dependent noise level
- Loss: Gaussian NLL $= \frac{1}{2}\exp(-\log \sigma^2)(y-\mu)^2 + \frac{1}{2}\log \sigma^2$

Advantages:

- Single forward pass
- Captures input-dependent noise
- Simple to implement

Disadvantages:

- Only captures aleatoric uncertainty
- Assumes Gaussian output noise
- Does not capture epistemic uncertainty

Output: Mean + predicted variance $\sigma^2(\mathbf{x})$

Conformal Prediction

Type: Neither (Distribution-Free Coverage)

Core Idea:

- Wrapper around ANY pre-trained predictor [4]
- Split calibration data by non-conformity score

Advantages:

- Model-agnostic
- Finite-sample coverage guarantee
- NO retraining required
- Valid regardless of model quality

Disadvantages:

- Requires held-out calibration data
- Intervals can be conservative
- Exchangeability assumption

Output: Prediction sets/intervals with guaranteed coverage

Diffusion Models (DDPM)

Type: Both (Epistemic in denoising, Aleatoric output) — **Generative**: YES

Core Idea:

- **Forward process**: Add Gaussian noise iteratively $q(x_t|x_{t-1})$
- **Reverse process**: Learn denoiser $p_\theta(x_{t-1}|x_t)$ via score matching
- **Sampling**: Iterate from $x_T \sim \mathcal{N}(0, I)$ down to x_0 [5]

Uncertainty Quantification:

- Generate M samples from $p(x|y)$
- Compute variance across samples
- Score network variance as epistemic uncertainty

Advantages:

- State-of-the-art generation quality
- Flexible conditioning
- Sample diversity = natural uncertainty

Disadvantages:

- Computationally expensive (many reverse steps)
- Long sampling time

Output: Samples from $p(x|y)$ with inherent diversity

Kalman Filter / EKF

Type: Both (Process noise = Aleatoric, State = Epistemic)

Core Idea:

- Linear-Gaussian state-space model:

$$\mathbf{x}_t = \mathbf{F}\mathbf{x}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q})$$
$$\mathbf{y}_t = \mathbf{H}\mathbf{x}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R})$$

- Recursive Bayesian filter: Prediction $\rightarrow$ Update

Output: Posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ with covariance $\mathbf{P}_t$

Uncertainty:

- **Aleatoric**: Process noise $\mathbf{Q}$, measurement noise $\mathbf{R}$
- **Epistemic**: State covariance $\mathbf{P}_t$ (shrinks with observations)

Applications: Navigation, tracking, time-series forecasting

Physics-Informed Neural Networks (PINNs)

Type: Epistemic (Parameter + Model Uncertainty)

Core Idea:

- Incorporate partial differential equation as a constraint in loss:

$$\mathcal{L} = \mathcal{L}_{\text{data}} + \lambda \mathcal{L}_{\text{physics}}$$

  where $\mathcal{L}_{\text{physics}}$ is a loss function that includes physical model
- Network learns to satisfy both data and physics simultaneously

Advantages:

- Learn from sparse data (PDE provides regularization)
- Physically consistent solutions
- Inverse problem capability

Applications: Fluid dynamics, heat transfer, inverse problems

Summary: 7 UQ Method Categories

Category	**Methods**	**Single-Pass?**	**Uncertain Type**	**Multi-Pass**
1. Inference	BNN, GP, Bayes LR	Partial	Epistemic	Yes (MCMC)
2. Sampling/Ensemble	Deep Ens, MC Drop, Bootstrap	No	Epistemic	Multiple
3. Direct Single-Pass	Hetero, Quantile, EDL	**YES**	Aleatoric/Both	Single
4. Post-hoc	Conformal, Temp Scale	**YES**	Neither	Single
5. Generative	VAE, Diffusion, Flows	Partial	Both	Multiple (sampling)
6. State-Space	Kalman, Particle, HMM	N/A	Both	Sequential
7. Physics-Informed	PINNs, Operators	No	Epistemic	Ensemble

References

- Gal, Y., & Ghahramani, Z. (2016). *Dropout as a Bayesian approximation: Representing model uncertainty in deep learning*. ICML.
- Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). *Simple and scalable predictive uncertainty estimation using deep ensembles*. NeurIPS.
- Sensoy, M., Kaplan, L., & Kannan, M. (2018). *Evidential deep learning to quantify classification uncertainty*. NeurIPS.
- Angelopoulos, A. N., & Bates, S. (2023). *A gentle introduction to conformal prediction and distribution-free uncertainty quantification*. Foundations and Trends in Machine Learning.
- Ho, J., Jain, A., & Abbeel, P. (2020). *Denoising diffusion probabilistic models*. NeurIPS.

Acknowledgements/ Document preparation

This document was converted into LaTeX slides and formatted with assistance from ChatGPT (OpenAI). The instructor provided the source text and verified the final structure and wording.