ELG 5218 - Uncertainty Evaluation in Engineering Measurements and Machine Learning: Proposed Midterm Exam Structure

Miodrag Bolic

February 2026

Overview

This document outlines a revised 75-minute midterm exam structure:

- **Short theoretical questions** on core concepts
- **Two computational/analysis problems**

The questions and problems will blend material from course slides, provided problems, course assignments and analysis of Python script results and diagnostics.

Total Duration: 75 minutes **Grading Distribution (Tentative):**

- around 45% Theoretical questions (Part A)
- around 35% Problem 1: Computation and analysis (Part B)
- around 35% Problem 2: Computation and analysis of model results and interpretation (Part C)

There will be about 15% bonus points.

Detailed Structure

Part A: Short Theoretical Questions

Format: 6–8 short conceptual questions requiring 1–3 sentence answers (or brief derivations).

Everything that is covered in the first 4 weeks, including:

- **Topic 1: Introduction and Bayesian modeling**
  - Bayesian modeling
- **Topic 2: Bayesian Inference for Gaussians; Bayesian Linear and Logistic Regression**
  - Gaussian Formulas
  - Gaussian Estimates
  - Bayesian Linear Regression
- **Topic 3: Bayesian Logistic Regression; Monte Carlo Sampling**
  - Bayesian Logistic Regression
  - Markov Chain Monte Carlo (MCMC)
  - Langevin Monte Carlo
- **Topic 4: Kalman and Particle Filters**
  - State Space Models
  - Kalman Filter
  - Particle Filters

Sample Questions:

1. State the detailed balance condition and explain why it is sufficient for MCMC design.
2. Why does the evidence $p(\mathcal{D})$ cancel in the Metropolis-Hastings acceptance ratio?
3. Define aleatoric and epistemic uncertainty and explain which decreases with more data.
4. What does a Gelman-Rubin statistic $\hat{R} = 1.05$ indicate about convergence?
5. Sketch the posterior covariance evolution in a Kalman filter over time.
6. In Bayesian linear regression, explain why the intercept and slope are negatively correlated.

Part B: Problem 1 – Computation and analysis

Type B1: Gaussian Identities or Bayesian Regression Derivation

Example: Bayesian linear regression with Gaussian identities.
Structure:

(a) **Model Setup:** State the likelihood, prior, and use Gaussian Identities (Theorems 2–3, Corollaries 1–2) to identify which theorems apply.

(b) **Posterior Derivation:** Derive $p(\mathbf{w}|\mathcal{D})$ in closed form. Write down the posterior mean and covariance explicitly.

(c) **Numerical Computation:** Apply formulas to concrete data (e.g., design matrix X, observations $\mathbf{y}$). Compute posterior mean and at least one element of covariance.

(d) **Posterior Predictive:** Derive the predictive distribution $p(y_*|\mathbf{x}_*, \mathcal{D})$ and decompose variance into aleatoric and epistemic components.

(e) **Interpretation:** Discuss the posterior correlation structure (e.g., centering effect) and how epistemic uncertainty grows away from training data.

Type B2: Laplace Approximation or MCMC Algorithm

Example: Metropolis-Hastings implementation for logistic regression or Laplace approximation.
Structure:

(a) **Model Setup:** Define the posterior (e.g., logistic regression likelihood + Gaussian prior) and state what makes it non-conjugate.

(b) **Algorithm Derivation:** For Laplace: find the posterior mode (gradient condition) and Hessian-based covariance. For MCMC: derive the acceptance ratio and explain the proposal.

(c) **Pseudocode or Implementation:** Write pseudocode or explain step-by-step how you would implement the method (e.g., MH iteration).

(d) **Tuning and Diagnostics:** Discuss how to tune hyperparameters (e.g., proposal variance for MH) and what diagnostics to check.

(e) **Comparison:** Compare this method to an alternative (e.g., Laplace vs. MCMC, random-walk vs. HMC) in terms of accuracy, computational cost, and assumptions.

Part C: Problem 2 – Computation and analysis of model results and interpretation

Format: Analyze output and diagnostics (no coding required).
Three possible scenarios (choose one per exam):

Scenario C1: Kalman Filter Output Analysis

Provided:

- Time series of state estimates: $\hat{z}_{1,t}$ (position) and $\hat{z}_{2,t}$ (velocity)
- Evolution of posterior covariance: $\Sigma_{1,1,t|t}, \Sigma_{2,2,t|t}, \Sigma_{1,2,t|t}$ (diagonal and cross terms)
- Innovation sequence (measurement residuals): $\nu_t = y_t - \hat{y}_{t|t-1}$
- Plots: trace of Σ_t, trace of innovations, filter output vs. measurements

Questions:

(a) Interpret the convergence of Σ_t to steady state. Why does the posterior covariance stabilize, and what does this imply about the filter's information gain?

(b) Analyze the innovation sequence. Is it approximately zero-mean and white noise? If not, what could this reveal about model misspecification or filter tuning?

(c) Compare one-step-ahead predictions vs. filtered estimates. Where is epistemic uncertainty larger, and why does it grow away from measurement updates?

(d) If measurement noise R were doubled, how would you expect Σ_t, $\mathbf{K}_t$, and the innovation magnitude to change qualitatively? Explain the trade-offs.

(e) Based on the diagnostics, assess whether the constant-velocity model is appropriate for this data. What additional analysis would you perform?

Scenario C2: MCMC Sampler Comparison

Provided:

- Trace plots for two samplers (e.g., random-walk MH vs. NUTS)
- Autocorrelation function (ACF) at lags 0–40
- Summary statistics: ESS, $\hat{R}$, acceptance rate, posterior mean/SD
- Diagnostics table: mixing time, effective sample size per iteration
- Posterior density plots (overlaid for both samplers)

Questions:

(a) Visually assess the trace plots. Do both chains appear to have converged to stationarity? Which shows faster mixing (i.e., smaller autocorrelation)?

(b) Compare the ACF behavior. How many effective samples does each sampler produce per 1000 iterations? Calculate or explain the ESS ratio and interpret its meaning.

(c) Interpret the Gelman-Rubin statistic $\hat{R}$ for each sampler. What does $\hat{R} < 1.01$ indicate about across-chain variability?

(d) Discuss the computational cost trade-off. If NUTS is slower per iteration but has higher ESS per iteration, which is more efficient overall for obtaining a fixed number of effective samples?

(e) Based on all diagnostics, which posterior distribution (sampler A or B) would you use for downstream inference, and why? What caveats or additional checks would you recommend?

Key Design Principles

1. Balanced Coverage

- **Part A (40%):** Core theory without computation—conceptual understanding.
- **Part B (30%):** Direct application of course assignments—technical skills.
- **Part C (30%):** Real data and script output interpretation—practical reasoning.

2. Mixed Assignment and Analysis Focus

- **Part B** reinforces mathematical derivations (what students worked on in assignments).
- **Part C** trains interpretation skills (how to read diagnostics from Python scripts).
- Together, they connect theory → implementation → validation.

3. Time Management

- Part A: 12–15 min (theory is quick for prepared students).
- Part B: 20–25 min (computational problem with moderate complexity).
- Part C: 15–20 min (no computation; reading and reasoning only).
- Buffer: 5 min remaining for review.

4. Flexibility and Consistency

- **Part A:** Always 6–8 theory questions across all topics.
- **Part B:** Choose one type (Gaussian identities/regression OR Laplace/MCMC) based on recent assignments.
- **Part C:** Choose one scenario (KF analysis OR MCMC comparison OR LR analysis) based on current topic.
- All three parts can be adapted to cover different course milestones.

Assessment Rubric

Part A: Theoretical Questions (40 pts total)

- 2 pts per question × 6–8 questions
- Quick grading; partial credit for incomplete reasoning

Part B: Computational Problem (25 pts total)

- (a) Model setup and theorem identification: 4 pts
- (b) Derivation: 7 pts (credit for clear reasoning and correct form)
- (c) Numerical computation: 6 pts (credit for correct formula application)
- (d) Prediction/analysis: 5 pts
- (e) Interpretation: 3 pts

Part C: Script Analysis Problem (25 pts total)

- (a)–(e) Analysis questions: 4–5 pts each
- Graded on reasoning and interpretation, not computation accuracy

Total: 90 pts, scaled to 100 for final grade