Gaussian signals exercise

Florent Leclercq[a)]

CNRS & Sorbonne Université, UMR 7095, Institut d'Astrophysique de Paris, 98 bis boulevard Arago, F-75014 Paris, France

(Dated: 8 April 2025)

This exercise focuses on Bayesian inference for the mean μ and variance σ^2 of a Gaussian signal using the Gaussian-inverse-Gamma distribution as a conjugate prior. The goal is to derive the posterior using implicit likelihood inference and compare to the result obtained analytically. We also explore the effect of data compression on inference quality and compare models using Bayesian evidence. Approximate Bayesian Computation (ABC) is applied with both standard and score-compressed statistics, while information-theoretic measures (e.g., mutual information) are used to evaluate compression efficiency.

Key concepts:

- Conjugate priors and the Gaussian-inverse-Gamma distribution
- Sufficient statistics
- Approximate Bayesian computation
- Score compression and Fisher-Rao distance
- Information theory for quantifying compression quality
- Bayesian model comparison

I. PROBLEM

The Gaussian-inverse-Gamma distribution is a four-parameter family of bivariate probability density functions (pdfs). Suppose that $\mu \sim \mathcal{G}\left(\eta, \frac{\sigma^2}{\lambda}\right)$ and $\sigma^2 \sim \mathit{\Gamma}^{-1}(\alpha, \beta)$, then by definition, (μ, σ^2) follows a Gaussian-inverse-Gamma distribution with parameters $(\eta, \lambda, \alpha, \beta)$, denoted $\mathcal{G}\mathit{\Gamma}^{-1}(\eta, \lambda, \alpha, \beta)$. The pdf is given by:

$$\mathcal{G}\mathit{\Gamma}^{-1}(\mu, \sigma^2|\eta, \lambda, \alpha, \beta) = \frac{\sqrt{\lambda}}{\sqrt{2\pi\sigma^2}} \cdot \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \exp\left(-\frac{2\beta + \lambda(\mu-\eta)^2}{2\sigma^2}\right), \tag{1}$$

for $\lambda > 0$, $\alpha > 0$, $\beta > 0$. Recall the usual Gaussian (with mean μ and variance σ^2), Gamma (with shape $\alpha > 0$ and scale $\theta > 0$), and inverse-Gamma (with shape $\alpha > 0$ and scale $\beta > 0$) pdfs:

$$\mathcal{G}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right], \tag{2}$$

$$\mathit{\Gamma}(x|\alpha, \theta) = \frac{1}{\Gamma(\alpha)\theta^\alpha} x^{\alpha-1} \exp\left(\frac{-x}{\theta}\right), \tag{3}$$

$$\mathit{\Gamma}^{-1}(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \exp\left(-\frac{\beta}{x}\right) \tag{4}$$

The problem considered is the joint inference of the unknown mean μ and unknown variance σ^2 of a Gaussian signal, from which we have n samples that constitute the observed data $\mathbf{d}_\mathrm{obs}$.

II. ANALYTIC SOLUTION

1. What is the likelihood $p(\mathbf{d}_\mathrm{obs}|\mu, \sigma^2)$ for this problem?
2. Show that the Gaussian-inverse-Gamma distribution is a conjugate prior for this likelihood. Derive the parameters $(\eta', \lambda', \alpha', \beta')$ of the posterior in terms of the parameters $(\eta, \lambda, \alpha, \beta)$ of the prior:

$$\eta' = \frac{\lambda\eta + n\Phi_\mathrm{O}^1}{\lambda + n}, \tag{5}$$

$$\lambda' = \lambda + n, \tag{6}$$

$$\alpha' = \alpha + \frac{n}{2}, \tag{7}$$

$$\beta' = \beta + \frac{n\lambda}{\lambda+n}\frac{(\Phi_\mathrm{O}^1 - \eta)^2}{2} + \frac{n-1}{2}\Phi_\mathrm{O}^2, \tag{8}$$

[a)]Electronic mail: florent.leclercq@iap.fr; https://www.florent-leclercq.eu/; ORCID: 0000-0002-9339-1404

where

$$\Phi^1_{\mathrm{obs}} = \frac{1}{n}\sum_{k=1}^{n} d_{\mathrm{obs},k}, \tag{9}$$

$$\Phi^2_{\mathrm{obs}} = \frac{1}{n-1}\sum_{k=1}^{n} \left(d_{\mathrm{obs},k} - \Phi^1(\mathbf{d}_{\mathrm{obs}})\right)^2. \tag{10}$$

$$\tag{11}$$

3. Assume the ground truth values $\mu_{\mathrm{true}} = 0.8$ and $\sigma^2_{\mathrm{true}} = 2.9$. Assume a prior with parameters $(\eta, \lambda, \alpha, \beta) = (0, 6, 22, 54)$. Give numerical values for the parameters $(\eta', \lambda', \alpha', \beta')$ of the posterior. Plot the analytical result of the problem (you can plot, for example, the 68%, 95% and 99% credible regions of the prior and the posterior in the μ-σ^2 plane, see appendix A 1).

III. DATA COMPRESSION

1. For implicit likelihood inference, we are going to need to compress the data into summary statistics. We are going to use the empirical mean and empirical variance of the data, i.e.

$$\Phi^1(\mathbf{d}) = \frac{1}{n}\sum_{k=1}^{n} d_k, \tag{12}$$

$$\Phi^2(\mathbf{d}) = \frac{1}{n-1}\sum_{k=1}^{n} \left(d_k - \Phi^1(\mathbf{d})\right)^2. \tag{13}$$

Prove that $\mathbf{\Phi} = (\Phi^1(\mathbf{d}), \Phi^2(\mathbf{d}))$ is a sufficient statistic for the problem, i.e. $p(\mu, \sigma^2|\mathbf{d}) = p(\mu, \sigma^2|\mathbf{\Phi})$. For this, you can use the Neyman-Fisher factorisation theorem, which states that a statistic $\mathbf{\Phi}(\mathbf{d})$ is sufficient for parameters $\boldsymbol{\theta}$ if we can write the likelihood $p(\mathbf{d}|\boldsymbol{\theta})$ in the form

$$p(\mathbf{d}|\boldsymbol{\theta}) = g\left(\mathbf{\Phi}(\mathbf{d}), \boldsymbol{\theta}\right) h(\mathbf{d}), \tag{14}$$

where the function $h(\mathbf{d})$ does not depend on the parameters.

2. Furthermore, show that the distributions of $\Phi^1(\mathbf{d})$ and $\Phi^2(\mathbf{d})$ have the following closed form:

$$\Phi^1 \sim \mathcal{G}\left(\mu, \frac{\sigma^2}{n}\right), \tag{15}$$

$$\Phi^2 \sim \Gamma\left(\frac{n-1}{2}, \frac{2\sigma^2}{n-1}\right). \tag{16}$$

IV. SIMULATIONS

1. Express the forward problem of generating $\mathbf{\Phi}$ given $(\eta, \lambda, \alpha, \beta)$ as a Bayesian hierarchical model. Represent the model graphically using a directed acyclic graph (DAG).

2. Implement a simulator that accepts as input $\eta, \lambda, \alpha, \beta$, and returns simulated data $\mathbf{d}$. Write a compressor that compresses data $\mathbf{d}$ to $\mathbf{\Phi}$ using equations (12) and (13).

3. Run a single simulation at the ground truth values $\mu_{\mathrm{true}} = 0.8$ and $\sigma^2_{\mathrm{true}} = 2.9$ to generate your mock data $\mathbf{d}_{\mathrm{obs}}$. What is the value of $\mathbf{\Phi}_{\mathrm{obs}}$, the compression of your mock data?

4. Choose some values of μ and σ^2, run some simulations and compress them. Make an histogram of Φ^1 and Φ^2 and overplot the theoretical distribution given by equations (15) and (16).

5. Make a scatter plot of simulated $\mathbf{\Phi}$ and overplot $\mathbf{\Phi}_{\mathrm{obs}}$.

V. IMPLICIT LIKELIHOOD INFERENCE VIA APPROXIMATE BAYESIAN COMPUTATION

1. Given your simulator, your compressor, and some threshold $\varepsilon > 0$, develop an Approximate Bayesian Computation (ABC) code, to perform the inference of (μ, σ^2) given $\mathbf{d}_{\mathrm{obs}}$. Specify the distance you are using between simulated and observed statistics.

2. Run your algorithm to infer μ and σ^2 for different, decreasing values of ε. How does your acceptance rate vary as a function of your threshold? Plot your results. You may show the samples and/or credible contours estimated from your samples (see appendix A 2).

3. Experiment with the choice of summary statistics. You may, for example, drop either Φ^1 or Φ^2 from $\mathbf{\Phi}$. How does this affect your ABC results?

VI. SCORE COMPRESSION

1. Define an expansion point as the mode of your prior. Run simulations at the expansion point and visualise them.

2. Write a score compressor at this expansion point using the equations given in the lectures. You may want to involve a simplifying assumption and evaluate numerically the necessary quantities, e.g. using finite differences in a set of simulations. We denote the score-compressed statistics as $\mathbf{\Psi}$.

VII. INFORMATION THEORY AND INFORMATION GEOMETRY

1. Estimate the mutual information between $\mathbf{\Phi}$ and $\mathbf{d}$, using a set of simulations. Similarly, estimate the mutual information between the score-compressed data $\mathbf{\Psi}$ and the full data $\mathbf{d}$. How good is the score compression with respect to the compression to the sufficient summary statistics?

2. Using the Mahalanobis distance, how far are the ground truth and the maximum a posteriori values from your prior? How far is the expansion point from the posterior?

3. Using the Fisher information matrix that appeared in your score compressor, write a function to compute the Fisher-Rao distance between two score-compressed data vectors.

VIII. IMPLICIT LIKELIHOOD INFERENCE, AGAIN

1. Modify your ABC algorithm to use the score-compressed data $\mathbf{\Psi}$ rather than $\mathbf{\Phi}$, and to use the Fisher-Rao distance rather than the distance you previously chose.

2. Run your algorithm to infer μ and σ^2 for different, decreasing values of the threshold ε. Compare to the posteriors obtained section V for a given simulation budget. What do you conclude regarding the quality of the compression?

IX. BAYESIAN MODEL COMPARISON

Suppose we have two models: $\mathcal{M}_1$ (the model used so far) has two free parameters μ and σ^2, and $\mathcal{M}_0$ is a simpler model where the variance is fixed to a given value $\sigma^2 = \sigma_0^2$.

1. Is it possible to compute the evidence for $\mathcal{M}_0$ and $\mathcal{M}_1$ analytically? You may want to refer to the calculation of the evidence in the case of the Laplace approximation. What is the result of Bayesian model comparison for your observed $\mathbf{d}_{\mathrm{obs}}$, given different values of σ_0^2?

2. Show that $\mathcal{M}_0$ and $\mathcal{M}_1$ are nested models and that the Bayes factor is given by the Savage-Dickey density ratio. Compute it numerically. Do you get to the same conclusion?

A. USEFUL PIECES OF CODE

1. Plotting contours

To visualise the 68%, 95% and 99% credible regions of a two-dimensional pdf using Matplotlib, you can use the following approach:

```
import numpy as np
import matplotlib.pyplot as plt

def get_contours(Z, nBins, confLevels=(.3173, .0455, .0027)):
    Z /= Z.sum()
```

```
    nContourLevels = len(confLevels)
    chainLevels = np.ones(nContourLevels+1)
    histOrdered = np.sort(Z.flat)
    histCumulative = np.cumsum(histOrdered)
    nBinsFlat = np.linspace(0., nBins**2, nBins**2)

    for l in range(nContourLevels):
        # Find location of contour level in 1d histCumulative
        temp = np.interp(confLevels[l], histCumulative, nBinsFlat)
        # Find "height" of contour level
        chainLevels[nContourLevels-1-l] = np.interp(temp, nBinsFlat, histOrdered)

    return chainLevels

mu_s = np.linspace(bounds['mu'][0],bounds['mu'][1],meshsize)
sigma_sq_s = np.linspace(bounds['sigma_sq'][0],bounds['sigma_sq'][1],meshsize)
mu_g, sigma_sq_g = np.meshgrid(mu_s,sigma_sq_s)
prior_g = prior_pdf(mu_g, sigma_sq_g, eta, lambda_, alpha, beta)
prior_g /= prior_g.sum()
prior_contours = get_contours(prior_g, meshsize)

plt.figure()
plt.contour(mu_g, sigma_sq_g, prior_g, levels=prior_contours)
```

2. Estimating the density from samples

To estimate the 68%, 95% and 99% credible regions of a two-dimensional pdf based on samples, first construct a 2D histogram. Then, use the function provided in A 1 to extract and plot the contours. For improved visualisation, you may optionally apply a Gaussian filter. Experiment to determine an appropriate variance for the filter. The following code snippet serves as an example:

```
nBins=30
mu_s = np.linspace(bounds['mu'][0],bounds['mu'][1],nBins)
sigma_sq_s = np.linspace(bounds['sigma_sq'][0],bounds['sigma_sq'][1],nBins)
mu_g, sigma_sq_g = np.meshgrid(mu_s,sigma_sq_s)

rejection_g, xedges, yedges = np.histogram2d(rejection_samples.T[1], rejection_samples.T[0],
    bins=nBins, range=[[bounds['sigma_sq'][0],bounds['sigma_sq'][1]] ,[bounds['mu'][0],bounds['
    mu'][1]]])
rejection_g /= rejection_g.sum()
rejection_g = scipy.ndimage.gaussian_filter(rejection_g, sigma=1)
rejection_contours = get_contours(rejection_g, nBins)

plt.figure()
plt.contour(mu_g, sigma_sq_g, rejection_g, levels=rejection_contours)
```