# Introduction to (Bayesian) Estimation
# MAE 5020

Bayesian inference

Oklahoma State University

October 16, 2023

# Overview

- All the previous estimators provide a point estimate.

- Because of inherent uncertainty due to noise in the data, point estimates are lacking how good the estimate is, can be fragile, overfit
- Bayesian inference aims at obtaining the posterior distribution $p(\theta|Z(k))$ (of course, based on Bayes theorem)

# Discrete case

$P(D = 1) = 0.01$: prior probability of being infected by a disease
$P(T|D = 1) = 0.95$ : positive if infected
$P(T|D = 0) = 0.05$: positive if not infected.
Suppose that someone is tested positive. What can be concluded?

- MLE of $D$: no consideration of $P(D)$.

- MAP of $D$: maximize $P(D)P(T|D)$

# Bayesian inference

Posterior distribution

$$P(D=1|T) = \frac{P(D=1)P(T|D=1)}{P(T)}$$

$$= \frac{P(D=1)P(T|D=1)}{P(T|D=1)P(D=1)+P(T|D=0)P(D=0)}$$

Therefore,

$$P(D=1|T) \approx 0.161$$

Use the same information as in MAP, but provide a posterior probability distribution on the outcome.

# Continuous distributions

Bayes theorem:

$$p(\theta|z) = \frac{p(z|\theta)p(\theta)}{p(z)}$$

- Since $p(z)$ is independent of $\theta$,
- What is $p(z)$? Following a similar idea from the discrete case

$$p(z) = \int p(z|\theta)p(\theta)d\theta$$

- This integral (aka ‘evidence’ or ’marginal likelihood’) is challenging to compute most of time.
- Special cases including linear Gaussian models.

# The generic linear model $Z(k) = H(k)\theta + V(k)$

$\theta \sim N(\theta; m_\theta, P_\theta)$ and $V(k) \sim N(0, R(k))$.
Show $p(\theta | Z(k)) \sim N(b, D)$ where

$$b = (P_\theta^{-1} + H^T R^{-1} H)^{-1} (P_\theta^{-1} m_\theta + H^T R^{-1} z)$$

$$D^{-1} = (P_\theta^{-1} + H^T R^{-1} H)^{-1}$$

# Connection with the conditional mean identity

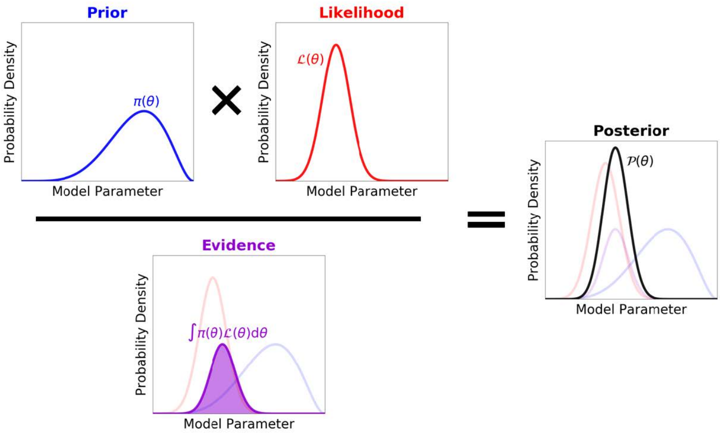
# Implications

- Posterior for linear Gaussian models

- Observe it in the information form.

- However, analytical posterior is rare!

# Go beyond the linear Gaussian models



- Full-blown Bayesian: Markov Chain Monte Carlo (MCMC) sampling
- Other approximation methods: linearization, Variational Inference, ...

# Benefits of posterior distribution

- Making educated guess: based on the posterior $p(\theta|Z(k))$, propose a point estimate $\hat{\theta}$
- Quantifying uncertainty: creating credible region/percentile, i.e., 1-sigma, 2-sigma.
- Generating predictions
- Comparing models

# Making prediction

Given a posterior distribution $p(\theta|Z(k))$, we can marginalize the distribution of $\theta$ to predict new data $\tilde{Z}$.

$$p(\tilde{Z}|Z(k)) = \int \underbrace{p(\tilde{Z}|\theta)}_{} \underbrace{p(\theta|Z(k))}_{} d\theta$$

- This is an expectation of $p(\tilde{Z}|\theta)$ w.r.t.

**We are more interested in computing integrals over the posterior rather than knowing the posterior itself.**

# Markov Chain Monte Carlo (MCMC) Sampling

- Sampling: a set of $K$ values $\theta_k$ drawn from a pdf $p(\theta)$.

$$\begin{array}{rl} \textit{Definition} & \{\theta_k\}_{k=1}^{K} \\ E_{p(\theta)}(\theta) = \int \theta p(\theta) d\theta & E_{p(\theta)}(\theta) \approx \frac{1}{K}\sum_{k=1}^{K}\theta_k \\ E_{p(\theta)}(g(\theta)) = \int g(\theta) p(\theta) d\theta & E_{p(\theta)}(g(\theta)) =\approx \frac{1}{K}\sum_{k=1}^{K} g(\theta_k) \end{array}$$

- Sampling approximation becomes exact when $K$ goes to infinity.

# Unnormalized posterior

Typically we do not have $p(\theta|Z(k))$, but only have $f(\theta) \triangleq p(\theta)p(Z(k)|\theta)$

- $p(\theta|Z(k)) \propto f(\theta)$. Thus, the normalizing constant $C$ is

$$E_{p(\theta)}(g(\theta)) = \int g(\theta)p(\theta)d\theta = \frac{\int g(\theta)f(\theta)d\theta}{C}$$

- When samples from $f(\theta)$ are available,

$$E_{p(\theta)}(g(\theta)) \approx$$

# MCMC

- MCMC seeks to generate samples proportional to the posterior $P(\theta|Z(k))$
- Simulate a *Markov chain* (a series of values) $\theta^1 \rightarrow \theta^2 \cdots \rightarrow \theta^K$ in a way that their density after "burning-in" period follows the posterior $p(\theta|Z(k))$.
- Markov chain:
- Monte Carlo:
- Key objective of MCMC: not to approximate/explore the posterior, but to **estimate the expectation**

# Metropolic-Hastings (M-H) MCMC

- Key idea: generate new samples $\theta^{i+1}$ from $\theta^i$ such that as $K \to \infty$
  1. the distribution of the samples converges
  2. the converging distribution is $p(\theta|Z(k))$
- M-H: Simplest MCMC algorithm
- The first condition is satisfied by *detailed balance* of the sample generating process $P(\theta^{i+1}|\theta^i)$:

$$P(\theta^{i+1}|\theta^i)P(\theta^i) = P(\theta^{i+1}, \theta^i) = P(\theta^i|\theta^{i+1})P(\theta^{i+1})$$

$$\frac{P(\theta^{i+1}|\theta^i)}{P(\theta^i|\theta^{i+1})} = \frac{P(\theta^{i+1})}{P(\theta^i)} = \frac{p(\theta^{i+1}|Z(k))}{p(\theta^i|Z(k))} \qquad *$$

since we want the converging distribution to be $p(\theta^{i+1}|Z(k))$.

# Sample generating process

1. Propose a new sample $\hat{\theta}^{i+1}$ based on a proposal distribution $Q(\hat{\theta}^{i+1}|\theta^i)$
2. Accept $\theta^{i+1} = \hat{\theta}^{i+1}$ or reject $\theta^{i+1} = \theta^i$ with some transition probability $T(\theta^{i+1}|\theta^i)$

- The proposal distribution chosen to be simple to generate new samples from simulations.
- $T(\theta^{i+1}|\theta^i)$ determined by the detailed balance equation (*)

$$P(\theta^{i+1}|\theta^i) = Q(\theta^{i+1}|\theta^i)\,T(\theta^{i+1}|\theta^i)$$

$$\frac{T(\theta^{i+1}|\theta^i)}{T(\theta^i|\theta^{i+1})} = \frac{Q(\theta^i|\theta^{i+1})p(\theta^{i+1}|Z(k))}{Q(\theta^{i+1}|\theta^i)p(\theta^i|Z(k))} = \frac{Q(\theta^i|\theta^{i+1})f(\theta^{i+1})}{Q(\theta^{i+1}|\theta^i)f(\theta^i)}$$

Metropolis criterion:

$$T(\theta^{i+1}|\theta^i) = \min\left\{1, \frac{Q(\theta^i|\theta^{i+1})f(\theta^{i+1})}{Q(\theta^{i+1}|\theta^i)f(\theta^i)}\right\} \qquad **$$

# Overall algorithm

1. Generate a new sample $\hat{\theta}^{i+1}$ from a proposal distribution $Q(\hat{\theta}^{i+1}|\theta^{i})$
2. Compute $T(\theta^{i+1}|\theta^{i})$ from (**)
3. Generate a random number $u_{i+1}$ uniformly distributed in $[0,1]$
4. If $u_{i+1} \leq T(\theta^{i+1}|\theta^{i})$, accept the move and set $\theta^{i+1} = \hat{\theta}^{i+1}$. Else, reject the move and set $\theta^{i+1} = \theta^{i}$.
5. Increment $i = i + 1$ and repeat.

# A simplified version

1. Generate a new sample $\hat{\theta}^{i+1}$ from a proposal distribution $Q(\hat{\theta}^{i+1}|\theta^i)$
2. Generate a random number $u_{i+1}$ uniformly distributed in $[0, 1]$
3. If $f(\hat{\theta}^{i+1})/f(\theta^i) > u_{i+1}$, accept and set $\theta^{i+1} = \hat{\theta}^{i+1}$. Else, reject the move and set $\theta^{i+1} = \theta^i$.
   **Or** 3)' If $\log f(\hat{\theta}^{i+1}) - \log f(\theta^i) > \log u_{i+1}$, accept and set $\theta^{i+1} = \hat{\theta}^{i+1}$. Else, reject the move and set $\theta^{i+1} = \theta^i$.

- A candidate $Q(\hat{\theta}^{i+1}|\theta^i)$: multi-variate Gaussian distribution for $\hat{\theta}^{i+1}$ with mean $\theta^i$ and some simple (e.g., diagonal) covariance.
- Algorithm requires $\frac{Q(\theta^i|\theta^{i+1})}{Q(\theta^{i+1}|\theta^i)} = 1$.

# Example

Sample from $p(\theta) \sim N(2, 2)$, using $Q(\hat{\theta}|\theta) \sim N(\theta, 1)$. Initialize the sampler with $\theta = 0$. Run the sampler for more than $10^4$ steps and plot the results as a histogram.

# Caveats

- Proposal distribution: user controlled function.
- Convergence: No simple answer. *You cannot know you have sampled the full posterior.*
- Autocorrelation: nearby points are strongly correlated, but sufficiently distant points will be less correlated.
- Initialization: typical/pretty good place in the posterior pdf. For example, run MAP estimation for a few steps.
- Burn-in period: Discard the beginning of your MCMC run befor using the samples.
- Multi-modal: may need multiple chains with different initializations.

# Introduction to Variational Inference (VI/VB)

- VI: approximating probability densities
- The same problem as MCMC

$$p(\theta|Z(k)) \propto p(\theta)p(Z(k)|\theta)$$

- MCMC does not scale well to large models or datasets (active investigation)
- VI: Alternative to MCMC sampling, faster and easier to scale to large data

# Main idea: Use optimization

1. Consider *a family* of approximate densities $\mathcal{Q}$.
2. Find $q \in \mathcal{Q}$ that minimizes the Kullback-Leibler divergence to $p(\theta|Z(k))$, i.e.,

$$q^*(\theta) = \arg\min_{q \in \mathcal{Q}} KL(q(\theta)||p(\theta|Z(k)))$$

3. Take $q^*(\theta)$ as the approximate posterior

- $\mathcal{Q}$ should be flexible to capture the target density but also simple for efficient optimization!
- Graphically:

# Kullback-Leiber (KL) Divergence (relative entropy)

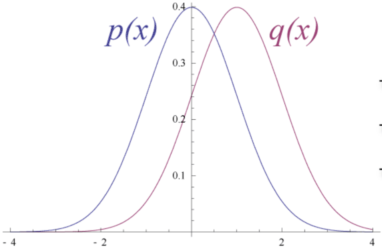
- Statistical distance measuring how one probability distribution differs from a second distribution

### KL divergence

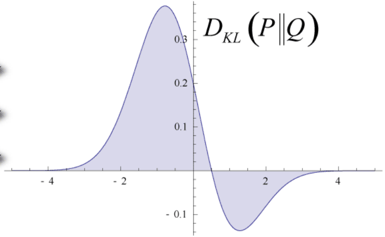
For two pdfs $p(x)$ and $q(x)$ of a continuous rv $x$, the KL divergence $KL(p||q)$ is defined as

$$KL(p||q) = \int p(x) \log \left( \frac{p(x)}{q(x)} \right) = E_p[\log p] - E_p[\log q] dx.$$

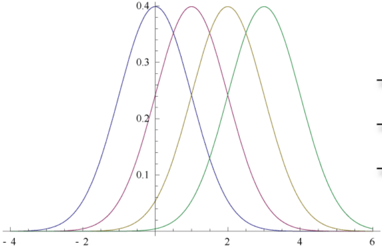
- Asymmetric:
- It does not satisfy triangle inequality.
- Thus, it is not a metric.

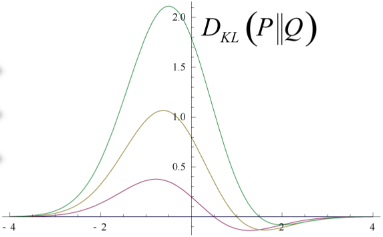


Original Gaussian PDF's



KL Area to be Integrated





# Properties of KL divergence

- $KL(p||q) \geq 0$. When $KL(p||q) = 0$, $p = q$ almost everywhere.
- KL divergence is invariant under parameter transformations.
- For any $\lambda \in [0, 1]$,

$$KL(\lambda_1 p_1 + (1-\lambda)p_2 || \lambda_1 q_1 + (1-\lambda)q_2) \leq$$

$$\lambda_1 KL(p_1||q_1) + (1-\lambda_1)KL(p_2||q_2)$$

# Example: Gaussian distribution

- Let $p_0 \sim N(\mu_0, \Sigma_0)$, $p_1 \sim N(\mu_1, \Sigma_1)$.

$$KL(p_0||p_1) = \frac{1}{2}\left(tr(\Sigma_1^{-1}\Sigma_0) - n + (\mu_1 - \mu_0)^T \Sigma_1^{-1}(\mu_1 - \mu_0) + \left(\ln \frac{\det \Sigma_1}{\det \Sigma_0}\right)\right).$$

Show the simple case where $\mu_0 = \mu_1 = 0$.

# Back to VI: Evidence lower bound (ELBO)

$$q^*(\theta) = \arg\min_{q \in \mathcal{Q}} KL(q(\theta)||p(\theta|Z(k)))$$

- The KL divergence objective is not computable because it depends on $\log p(Z(k))$.

$$KL(q(\theta)||p(\theta|Z(k))) = E_q[\log q(\theta)] - E_q[\log p(\theta|Z(k))].$$

- Therefore, we maximize an alternative objective

$$ELBO(q) = E_q[\log p(\theta, Z(k)) - E_q[\log q(\theta)]]$$

- Why are they equivalent?

# Rewrite ELBO

$$
\begin{aligned}
ELBO(q) &= E_q[\log p(\theta, Z(k)) - E_q[\log q(\theta)]] \\
&= E_q[\log p(Z(k)|\theta)p(\theta)] - E_q[\log q(\theta)] \\
&= E_q[\log p(Z(k)|\theta)] - KL(q||p)
\end{aligned}
$$

Two aspects

# Why is it called ELBO?

Evidence $\log p(Z(k)) \geq ELBO(q)$. Why?

$$\log p(Z(k)) = KL(q(\theta)||p(\theta|Z(k))) + ELBO(q)$$

# Maximization of ELBO

$$
\begin{aligned}
ELBO(q) &= E_q[\log p(\theta, Z(k)) - E_q[\log q(\theta)]] \\
&= E_q[\log p(Z(k)|\theta)] - KL(q||p)
\end{aligned}
$$

Two approaches:

- Mean-field variational family
- Fixed-field optimization

# Mean-field

- $\theta$ contains mutually independent components and each governed by a distinct factor in $q(\theta)$, i.e.,

$$q(\theta) = \prod_{j=1}^{m} q_j(\theta_j), \quad \theta \in \mathbb{R}^n$$

- Note that $q_j(\cdot)$'s are used to approximate the posterior $p(\theta_j|Z(k))$
- The correlation between $\theta_j$'s in $p(\theta_j|Z(k))$ is not captured in $q(\theta)$.
- $q_j(\cdot)$ can take any parametric form appropriate to the corresponding random variable, e.g., Gaussian.

# Example

Choose $q_1 \sim N(0, s_1)$ and $q_2 \sim N(0, s_2)$ to approximate $p \sim N(0, \Sigma)$.

First, what is the distribution for $q$?

Second, calculate the KL divergence from formula.

$$KL(q||p) = \frac{1}{2}\left(tr(\Sigma^{-1} diag(s_1, s_2)) - 2 + \ln \det \Sigma - \ln s_1 s_2\right)$$

Third, minimize the KL divergence by optimizing $s_1$ and $s_2$

# Comparison in plotting

# Coordinate ascent VI (CAVI)

CAVI iteratively optimizes each $q_j$ while holding the other fixed. It climbs the ELBO to a local optimum.

$$q^*(\theta_j) \propto \exp\{E_{-j}[\log p(\theta_j|\theta_{-j}, Z(k))]\}$$

$$\propto \exp\{E_{-j}[\log p(\theta_j, \theta_{-j}, Z(k))]$$

- $\theta_{-j}$ :

- The expectation $E_{-j}$ is over $\theta_{-j}$, i.e.,

# CAVI Algorithm

```
**Input**: A model $p(\theta, Z(k)) = p(Z(k)|\theta)p(\theta)$, and data $Z(k)$
**Output**: variational density $q(\theta) = \prod_{j=1}^{m} q_j(\theta_j)$
**Initialization**: $q_j(\theta_j)$
**While** the ELBO has not converged **do**
    **for** $j \in \{1, \cdots, m\}$ **do**
        Set $q(\theta_j) \propto \exp\{E_{-j}[\log p(\theta_j|\theta_{-j}, Z(k))]$ or
        $\exp\{E_{-j}[\log p(\theta_j, \theta_{-j}, Z(k))$
    **end**
    Compute $ELBO(q) = E[\log p(\theta, Z(k))] - E[\log q(\theta)]$
**return** $q(z)$
```

# Example: Bayesian linear regression with Automatic Relevance Determination

Suppose that we are given data $Z \in \mathbb{R}^n$ and input $x \in \mathbb{R}^{n \times D}$. We are interested in finding a linear coefficient $\beta$ and relationship

$$z_i \approx \beta^T x_i = x_i^T \beta, \quad \forall i$$

where $\beta \in \mathbb{R}^D$, $z_i \in \mathbb{R}$ and $x_i \in \mathbb{R}^D$.

## Automatic relevance determination (ARD)

Assigns a separate prior for each $\beta_i$. Automatically shrinks $\beta_i$ if it is not relevant in the regression. ARD works by setting a hyper-prior for the prior on each $\beta_i$ to encourage small values.

# Formulation

Gaussian likelihood for the data ($\tau$: precision):

$$p(y|\beta,\tau) = \prod_{i=1}^{n} N(y_i|\beta^T x_i, \tau^{-1})$$

Priors on $\beta$ and $\tau$:

$$p(\beta,\tau|\alpha) = N(0, [\tau diag(\alpha)]^{-1}) Gam(\tau|a_0, b_0)$$

Hyper-prior on prior parameter $\alpha$:

$$p(\alpha) = \prod_{d=1}^{D} Gam(\alpha_d|c_0, d_0)$$

*Gam*: Gamma distribution, $a_0, b_0, c_0, d_0$ are fixed constants.

# CAVI

Infer the posterior $p(\beta, \tau, \alpha|y, x)$ using CAVI

$$q(\beta, \tau, \alpha) = q(\beta, \tau)q(\alpha)$$

Conditioned on $q(\alpha)$, identify the optimal $q(\beta, \tau)$:

$$\begin{aligned}
\log q(\beta, \tau) &= E_{q(\alpha)} \log[p(\alpha)p(\beta, \tau|\alpha)p(y|\beta, \tau)] + const. \\
&= E_{q(\alpha)} \log p(\beta, \tau|\alpha) + \log p(y|\beta, \tau) + const. \\
&= \log N(\beta|\beta_*, \tau^{-1} V_*) + \log Gam(\tau|a_*, b_*)
\end{aligned}$$

$V_*^{-1} = E_\alpha[diag(alpha)] + \sum_i x_i x_i^T$, $\beta_* = V_* \sum_i x_i y_i$,
$a_* = a_0 + n/2$, $b_* = b_0 + 1/2(\sum_i y_i^2 - \beta_*^T V_*^{-1} \beta_*)$

# On $q(\alpha)$

$$\begin{aligned} \log q(\alpha_d) &= E_{\beta,\tau}[\log p(\beta, \tau|\alpha_d)] + \log p(\alpha_d) + const. \\ &= \log Gam(\alpha_d|c_*, d_{*d}) \end{aligned}$$

$c_* = c_0 + 1/2,\ d_{*d} = d_0 + 1/2E_{\beta,\tau}[\tau\beta_d^2].$

The expectations can be computed as
$E_\alpha[diag(\alpha)] = c_* diag(1/d_*),\ E_{\beta,\tau}[\tau\beta_d^2] = \beta_{*d}^2 a_*/b_* + [V_*]_d.$

CAVI: Iteratively update $a_*$, $b_*$, $c_*$, $d_*$, $V_*^{-1}$, and $\beta_*$.

# Example: sparse linear regression

# Second approach: Fixed form VI

Assumes a fixed parametric form $q(\theta) = q_\lambda(\theta)$.

Example

Gaussian

Maximize the ELBO(q) by optimizing the parameters $\lambda$

$$ELBO(q_\lambda) = E_{q_\lambda}[\log p(\theta, Z(k)) - E_{q_\lambda}[\log q_\lambda(\theta)]]$$

$$= E_{q_\lambda}[\log p(Z(k)|\theta)] - KL(q_\lambda||p)$$

# The key step in the optimization: gradient of ELBO

$$
\begin{aligned}
\nabla_\lambda ELBO(q_\lambda) &= \nabla_\lambda \int q_\lambda(\theta) \log \frac{p(\theta)p(Z(k)|\theta)}{q_\lambda(\theta)} d\theta \\
&= \int \nabla_\lambda q_\lambda(\theta) \log \frac{p(\theta)p(Z(k)|\theta)}{q_\lambda(\theta)} d\theta \\
&\qquad - \int q_\lambda(\theta) \nabla_\lambda \log q_\lambda(\theta) d\theta \\
&= \int q_\lambda(\theta) \nabla_\lambda \log q_\lambda(\theta) \log \frac{p(\theta)p(Z(k)|\theta)}{q_\lambda(\theta)} d\theta \\
&\qquad - \int \nabla_\lambda q_\lambda(\theta) d\theta \\
&= E_{q_\lambda}[\nabla_\lambda \log q_\lambda(\theta) \log \frac{p(\theta)p(Z(k)|\theta)}{q_\lambda(\theta)}] - \nabla_\lambda \int q_\lambda(\theta) d\theta \\
&= E_{q_\lambda}\left[\nabla_\lambda \log q_\lambda(\theta) \cdot \log \frac{p(\theta)p(Z(k)|\theta)}{q_\lambda(\theta)}\right].
\end{aligned}
$$

“score-function gradient”

# Stochastic optimization: estimation of the gradient

- Draw samples $\theta_s \sim q_\lambda(\theta)$, $s = 1, \cdots, S$
- Compute an estimate of the gradient of ELBO by sample average:

$$\nabla_\lambda \widehat{ELBO} \triangleq \frac{1}{S} \sum_{s=1}^{S} \nabla_\lambda \log q_\lambda(\theta_s) \cdot \log \frac{p(\theta_s) p(Z(k)|\theta_s)}{q_\lambda(\theta_s)}$$

- Update $\lambda \leftarrow \lambda + a_t \nabla_\lambda \widehat{ELBO}$ with a step-size $a_t$

# Final comment: comparison between MCMC and VI

From *Variational Inference: A Review for Statisticians* by Blei, et al. 2016

“MCMC methods tend to be more computationally intensive than variational inference but they also provide guarantees of producing (asymptotically) exact samples from the target density. Variational inference does not enjoy such guarantees—it can only find a density close to the target—but tends to be faster than MCMC.”

“Thus, variational inference is suited to large data sets and scenarios where we want to quickly explore many models; MCMC is suited to smaller datasets and scenarios where we happily pay a heavier computational cost for more precise samples.”