Chapter X: Some Advanced Topics in Bayesian Statistics

ST4234: Bayesian Statistics

Semester 2, AY 2019/2020

Department of Statistics and Applied Probability

National University of Singapore

LI Cheng

stalic@nus.edu.sg

Outline

Overview

Bayesian Regression Models

Bayesian Model Comparison

Variational Bayes

Software related to Bayesian Inference

10.1 Overview

- This chapter contains some advanced topics in Bayesian statistics that are not covered in our lectures due to the time constraint.

- The contents include the following topics:

  1. Bayesian regression models
  2. Bayesian model comparison
  3. Variational Bayes
  4. Software related to Bayesian inference

Outline

Overview

Bayesian Regression Models

Bayesian Model Comparison

Variational Bayes

Software related to Bayesian Inference

10.2 Bayesian Regression Models

- We will focus on the Bayesian inference of linear regression models.
- A linear model takes the form

$$y_i = \beta_1 x_{1i} + \ldots + \beta_p x_{pi} + \epsilon_i,$$

  for $i = 1, \ldots, n$.

- y is the response variable / dependent variable.
- $x_1, \ldots, x_p$ are called the predictor variables / explanatory variables / independent variables / covariates, depending on the context. The y's and x's are all observed.
- If the intercept term is included, we can consider $x_1 \equiv 1$ to represent the intercept term.
- The subscript i indexes the observations.
- ϵ is the error term. It is usually assumed that $\mathrm{E}(\epsilon | x_1, \ldots, x_p) = 0$.

10.2 Bayesian Regression Models

Matrix form

- We can write the linear model in the matrix form.
- Let $\boldsymbol{y} = (y_1, \dots, y_n)^\top$, $\boldsymbol{x}_i = (x_{i1}, \dots, x_{ip})^\top$, $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)^\top$, and $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)^\top$. Then for each individual observation,
$$y_i = \boldsymbol{\beta}^\top \boldsymbol{x}_i + \epsilon_i, \text{ for } i = 1, \dots, n.$$
- We can further stack up the vectors of $\boldsymbol{x}_i$'s and let
$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_n^\top \end{bmatrix} = \begin{bmatrix} x_{11} & \dots & x_{1p} \\ & \dots & \\ x_{n1} & \dots & x_{np} \end{bmatrix}$$
which is a $n \times p$ matrix.
- Then the linear model (of n observations) can be written compactly as
$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}. \tag{1}$$

10.2 Bayesian Regression Models

- In regression analysis, there are several possible assumptions on the error ϵ. The basic requirement is that $\mathrm{E}(\epsilon|x_1, \ldots, x_p) = 0$.
- For Bayesian inference, we need to model the distribution of ϵ. The commonly used assumption is that $\epsilon_1, \ldots, \epsilon_n$ are i.i.d. $\mathrm{N}(0, \sigma^2)$ random variables (conditional on $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$).
- Under this assumption, we can write $\boldsymbol{\epsilon} \sim \mathrm{N}(0, \sigma^2 \mathbf{I})$, where $\mathbf{I}$ is the identity matrix (of dimension $n \times n$).
- The linear model in (1) can be equivalently written as

$$\boldsymbol{y}|\mathbf{X}, \boldsymbol{\beta}, \sigma^2 \sim \mathrm{N}\left(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}\right). \tag{2}$$

- It is usually not necessary to model the distribution of $\boldsymbol{x}_i$'s (or $\mathbf{X}$), as we will see the derivation of posterior distributions later. We can treat them as given constants.

- The parameters in Model (2) are $(\boldsymbol{\beta}, \sigma^2)$.
- Similar to the normal model in Chapter 3, we can impose the following normal-inverse gamma prior on $(\boldsymbol{\beta}, \sigma^2)$.

$$\boldsymbol{\beta} | \sigma^2 \sim \mathrm{N}\left(\boldsymbol{\mu}_0, \sigma^2 \boldsymbol{\Sigma}_0\right), \quad \sigma^2 \sim \text{Inv-Gamma}\left(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2}\right).$$

- $\boldsymbol{\mu}_0$ is a $p \times 1$ vector. $\boldsymbol{\Sigma}_0$ is a $p \times p$ positive definite matrix. $\nu_0 > 0$ and $\sigma_0^2 > 0$ are scalars. They are all hyperparameters.
- We can derive the posterior $p(\boldsymbol{\beta}, \sigma^2 | \boldsymbol{y}, \mathbf{X})$.

10.2 Bayesian Regression Models

Likelihood

- The likelihood function is based on the density of multivariate normal distribution:

$$p(\boldsymbol{y}|\mathbf{X},\boldsymbol{\beta},\sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{(\boldsymbol{y}-\mathbf{X}\boldsymbol{\beta})^\top(\boldsymbol{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^2}\right\}.$$

- This likelihood function can also be derived from the product of n pairs of observations $(y_i, \boldsymbol{x}_i)$:

$$\begin{aligned}
p(\boldsymbol{y}|\mathbf{X},\boldsymbol{\beta},\sigma^2) &= p(y_1,\ldots,y_n|\mathbf{X},\boldsymbol{\beta},\sigma^2)\\
&= \prod_{i=1}^n p(y_i|\boldsymbol{x}_i,\boldsymbol{\beta},\sigma^2)\\
&= \prod_{i=1}^n \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{(y_i-\boldsymbol{\beta}^\top\boldsymbol{x}_i)^2}{2\sigma^2}\right\}\\
&= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{\sum_{i=1}^n(y_i-\boldsymbol{\beta}^\top\boldsymbol{x}_i)^2}{2\sigma^2}\right\}\\
&= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{(\boldsymbol{y}-\mathbf{X}\boldsymbol{\beta})^\top(\boldsymbol{y}-\mathbf{X}\boldsymbol{\beta})}{2\sigma^2}\right\}.
\end{aligned}$$

10.2 Bayesian Regression Models
Posterior

- Under the normal-inverse gamma prior, the posterior of $(\boldsymbol{\beta}, \sigma^2)$ is

$$
\begin{aligned}
p(\boldsymbol{\beta}, \sigma^2 | \boldsymbol{y}, \mathbf{X}) &\propto p(\boldsymbol{y} | \mathbf{X}, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta}, \sigma^2) \\
&\propto \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{ -\frac{(\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})}{2\sigma^2} \right\} \\
&\quad \times \frac{1}{(2\pi\sigma^2)^{p/2} [\det(\boldsymbol{\Sigma}_0)]^{1/2}} \exp\left\{ -\frac{(\boldsymbol{\beta} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1} (\boldsymbol{\beta} - \boldsymbol{\mu}_0)}{2\sigma^2} \right\} \\
&\quad \times (\sigma^2)^{-\frac{\nu_0}{2} - 1} \exp\left(-\frac{\nu_0 \sigma_0^2}{2\sigma^2} \right).
\end{aligned}
$$

- To derive the posterior, we define $\widetilde{\boldsymbol{\beta}}$ as follows.

$$
\widetilde{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{X} + \boldsymbol{\Sigma}_0^{-1} \right)^{-1} \left(\mathbf{X}^\top \boldsymbol{y} + \boldsymbol{\Sigma}_0^{-1} \boldsymbol{\mu}_0 \right)
$$

- $\widetilde{\boldsymbol{\beta}}$ does not depend on $\boldsymbol{\beta}$.

10.2 Bayesian Regression Models

Posterior

- We can complete the square for $\boldsymbol{\beta}$ inside the exponent:

$$
\begin{aligned}
&(\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^\top(\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}) + (\boldsymbol{\beta} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\beta} - \boldsymbol{\mu}_0) \\
&= \boldsymbol{\beta}^\top \left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)\boldsymbol{\beta} - 2(\boldsymbol{y}^\top\mathbf{X} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1})\boldsymbol{\beta} + \boldsymbol{y}^\top\boldsymbol{y} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 \\
&= (\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})^\top \left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}) \\
&\quad \underbrace{-\widetilde{\boldsymbol{\beta}}^\top \left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)\widetilde{\boldsymbol{\beta}} + \boldsymbol{y}^\top\boldsymbol{y} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0}_{=S_0,\ \text{constant for } \boldsymbol{\beta}}.
\end{aligned}
$$

10.2 Bayesian Regression Models

Posterior

- From the joint posterior in the last page, the conditional posterior of $p(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X})$ is

$$
\begin{aligned}
p(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X}) \propto \exp\left\{-\frac{(\boldsymbol{y}-\mathbf{X}\beta)^\top(\boldsymbol{y}-\mathbf{X}\beta)}{2\sigma^2}\right\} \\
\times \exp\left\{-\frac{(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^\top\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)}{2\sigma^2}\right\} \\
\propto \exp\left\{-\frac{(\boldsymbol{\beta}-\widetilde{\beta})^\top\left(\mathbf{X}^\top\mathbf{X}+\boldsymbol{\Sigma}_0^{-1}\right)(\boldsymbol{\beta}-\widetilde{\beta})}{2\sigma^2}\right\}
\end{aligned}
$$

- Therefore, we have shown that

$$
\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X} \sim \mathrm{N}\left(\widetilde{\boldsymbol{\beta}}, \sigma^2\left(\mathbf{X}^\top\mathbf{X}+\boldsymbol{\Sigma}_0^{-1}\right)^{-1}\right).
$$

10.2 Bayesian Regression Models

Posterior

- We can then integrate out $\boldsymbol{\beta}$ from the joint posterior to obtain the marginal posterior of $p(\sigma^2|\boldsymbol{y}, \mathbf{X})$:

$$
\begin{aligned}
p(\sigma^2|\boldsymbol{y}, \mathbf{X}) &= \int_{\mathbb{R}^p} p(\boldsymbol{\beta}, \sigma^2|\boldsymbol{y}, \mathbf{X})\mathrm{d}\boldsymbol{\beta} \\
&\propto (\sigma^2)^{p/2} \exp\left\{-\frac{-\widetilde{\boldsymbol{\beta}}^\top\left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)\widetilde{\boldsymbol{\beta}} + \boldsymbol{y}^\top\boldsymbol{y} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0}{2\sigma^2}\right\} \\
&\quad \times (\sigma^2)^{-n/2} \times (\sigma^2)^{-p/2} \times (\sigma^2)^{-\frac{\nu_0}{2}-1} \exp\left(-\frac{\nu_0\sigma_0^2}{2\sigma^2}\right) \\
&\propto (\sigma^2)^{-\frac{\nu_0+n}{2}-1} \exp\left\{-\frac{\boldsymbol{y}^\top\boldsymbol{y} - \widetilde{\boldsymbol{\beta}}^\top\left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)\widetilde{\boldsymbol{\beta}} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 + \nu_0\sigma_0^2}{2\sigma^2}\right\}
\end{aligned}
$$

- The blue part is the remaining terms after integrating out $\boldsymbol{\beta}$. The red part is from the likelihood. The teal part is from the prior of $(\boldsymbol{\beta}, \sigma^2)$.

10.2 Bayesian Regression Models

Posterior

- Therefore, we have derive that

$$\sigma^2|\boldsymbol{y},\mathbf{X} \sim \text{Inv-Gamma}\left(\frac{\nu_0+n}{2},\frac{\nu_0\sigma_0^2+S_0}{2}\right)$$

$$\text{where } S_0 = \boldsymbol{y}^\top\boldsymbol{y} - \widetilde{\boldsymbol{\beta}}^\top\left(\mathbf{X}^\top\mathbf{X}+\boldsymbol{\Sigma}_0^{-1}\right)\widetilde{\boldsymbol{\beta}} + \boldsymbol{\mu}_0^\top\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0.$$

- In summary,

$$\begin{aligned} & \boldsymbol{\beta}|\sigma^2 \sim \mathrm{N}(\boldsymbol{\mu}_0,\sigma^2\boldsymbol{\Sigma}_0), \quad \sigma^2 \sim \text{Inv-Gamma}\left(\tfrac{\nu_0}{2},\tfrac{\nu_0\sigma_0^2}{2}\right), \\ & \boldsymbol{y} \sim \mathrm{N}(\mathbf{X}\boldsymbol{\beta},\sigma^2\mathbf{I}) \\ \Longrightarrow \quad & \boldsymbol{\beta}|\sigma^2,\boldsymbol{y},\mathbf{X} \sim \mathrm{N}\left(\widetilde{\boldsymbol{\beta}},\sigma^2\left(\mathbf{X}^\top\mathbf{X}+\boldsymbol{\Sigma}_0^{-1}\right)^{-1}\right), \\ & \sigma^2|\boldsymbol{y},\mathbf{X} \sim \text{Inv-Gamma}\left(\tfrac{\nu_0+n}{2},\tfrac{\nu_0\sigma_0^2+S_0}{2}\right). \end{aligned}$$

10.2 Bayesian Regression Models

Marginal posterior for $\boldsymbol{\beta}$

- We can also find the marginal posterior of $\boldsymbol{\beta}$ by integrating out σ^2, similar to the normal model in Chapter 3.
- For short, let $S_1 = (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^\top(\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}) + (\boldsymbol{\beta} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\beta} - \boldsymbol{\mu}_0)$. Then

$$
\begin{aligned}
p(\boldsymbol{\beta}|\boldsymbol{y}, \mathbf{X}) &= \int_0^\infty p(\boldsymbol{\beta}, \sigma^2|\boldsymbol{y}, \mathbf{X}) \mathrm{d}\sigma^2 \\
&\propto \int_0^\infty (\sigma^2)^{-\frac{p+n+\nu_0}{2}-1} \exp\left\{-\frac{S_1 + \nu_0\sigma_0^2}{2\sigma^2}\right\} \mathrm{d}\sigma^2 \\
&\propto \Gamma\left(\frac{p+n+\nu_0}{2}\right)\left(\frac{S_1 + \nu_0\sigma_0^2}{2}\right)^{-\frac{p+n+\nu_0}{2}} \\
&\overset{(*)}{\propto} \left\{(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})^\top \left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}) + S_0 + \nu_0\sigma_0^2\right\}^{-\frac{p+n+\nu_0}{2}} \\
&\propto \left\{1 + \frac{1}{\nu_0 + n}(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}})\right\}^{-\frac{(\nu_0+n)+p}{2}},
\end{aligned}
$$

where we define $\boldsymbol{\Sigma}_1 = \dfrac{\nu_0\sigma_0^2 + S_0}{\nu_0 + n}\left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)^{-1}$, and the step $(*)$ is from page 11.

Marginal posterior for $\boldsymbol{\beta}$

- We use $t_{p,\nu}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ to denote the p-dimensional multivariate-t distribution with center vector $\boldsymbol{\mu}$, scale matrix $\boldsymbol{\Sigma}$, and ν degrees of freedom.
- The derivation from the last slide shows that the marginal posterior of $\boldsymbol{\beta}$ is

$$\boldsymbol{\beta}|\boldsymbol{y}, \mathbf{X} \sim t_{p,\nu_0+n}\left(\widetilde{\boldsymbol{\beta}},\ \boldsymbol{\Sigma}_1\right),$$

$$\text{where } \boldsymbol{\Sigma}_1 = \frac{\nu_0\sigma_0^2 + S_0}{\nu_0 + n}\left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)^{-1},$$

and $\widetilde{\boldsymbol{\beta}}, S_0, S_1$ are defined as before.

- We can also use Gibbs sampler to sample from the posterior of $(\boldsymbol{\beta}, \sigma^2)$, similar to Example 1 in Chapter 8. We can draw $\boldsymbol{\beta}$ and σ^2 from their conditional posteriors iteratively. The details are omitted here.

- Suppose that we predict a future observation at the predictor or covariate $\boldsymbol{x}^*$. The predicted response should be distributed as $\tilde{y}|\boldsymbol{x}^*, \boldsymbol{\beta}, \sigma^2 \sim \mathrm{N}(\boldsymbol{\beta}^\top \boldsymbol{x}^*, \sigma^2)$, where $(\boldsymbol{\beta}, \sigma^2)$ are drawn from the posterior $p(\boldsymbol{\beta}, \sigma^2|\boldsymbol{y}, \mathbf{X})$.

- Computationally, we can use Monte Carlo approximation to obtain the predictive distribution of $\tilde{y}$: For $s = 1, \ldots, S$, we
  1. Draw $(\sigma^2)^{(s)}$ from the marginal posterior $\text{Inv-Gamma}\left(\frac{\nu_0 + n}{2}, \frac{\nu_0\sigma_0^2 + S_0}{2}\right)$;
  2. Draw $\boldsymbol{\beta}^{(s)}$ from the conditional posterior $\mathrm{N}\left(\widetilde{\boldsymbol{\beta}}, (\sigma^2)^{(s)}\left(\mathbf{X}^\top\mathbf{X} + \boldsymbol{\Sigma}_0^{-1}\right)^{-1}\right)$;
  3. Draw $\tilde{y}^{(s)}$ from $\mathrm{N}(\boldsymbol{\beta}^{(s)\top}\boldsymbol{x}^*, (\sigma^2)^{(s)})$.

10.2 Bayesian Regression Models

Noninformative prior

- The choice of the hyperparameters $\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \nu_0, \sigma_0^2$ affect the posterior distribution.

- Similar to the normal model, we can also find noninformative prior for $(\boldsymbol{\beta}, \sigma^2)$. For example, we can set $\boldsymbol{\Sigma}_0$ to ∞, $\nu_0 \to 0$ and $\sigma_0^2 \to 0$.

- Alternatively, we can use the Jeffreys prior $p(\boldsymbol{\beta}, \sigma^2) \propto 1/\sigma^2$, which gives a slightly different posterior.

- Another possibility is to use a weakly informative prior, which accounts for the covariance structure in $\boldsymbol{\beta}$ using the observed design matrix $\mathbf{X}$.

10.2 Bayesian Regression Models

Zellner's g-prior

- To derive the weakly informative prior for $\boldsymbol{\beta}$, we can think about the unit information from the observed data about the variance of $\boldsymbol{\beta}$.

- From standard linear regression analysis, we know that the ordinary least square (OLS) estimator $\widehat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \boldsymbol{y}$ has a variance $\sigma^2(\mathbf{X}^\top \mathbf{X})^{-1}$.

- The precision of $\widehat{\boldsymbol{\beta}}$ is the inverse of this variance, $(\mathbf{X}^\top \mathbf{X})/\sigma^2$, which can be viewed as the amount of information in n observations.

- Thus, the unit information prior sets the prior precision of $p(\boldsymbol{\beta}|\sigma^2)$ as $(\mathbf{X}^\top \mathbf{X})/(n\sigma^2)$, or $\sigma^2\boldsymbol{\Sigma}_0 = n\sigma^2(\mathbf{X}^\top \mathbf{X})^{-1}$.

- More generally, Zellner (1986) has used the choice $\sigma^2\boldsymbol{\Sigma}_0 = g\sigma^2(\mathbf{X}^\top \mathbf{X})^{-1}$, where $g > 0$ is a hyperparameter. Then $\boldsymbol{\beta}|\sigma^2 \sim \mathrm{N}(\boldsymbol{\mu}_0, g\sigma^2(\mathbf{X}^\top \mathbf{X})^{-1})$. This prior is called the Zellner's g-prior.

Zellner's g-prior

- Under the Zellner's g-prior $\beta|\sigma^2 \sim \mathrm{N}(\boldsymbol{\mu}_0, g\sigma^2(\mathbf{X}^\top\mathbf{X})^{-1})$, if we further assume that $\boldsymbol{\mu}_0 = 0$, then we can obtain that

$$\mathrm{E}(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X}) = \frac{g}{g+1}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{y},$$
$$\mathrm{Var}(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X}) = \frac{g}{g+1}\sigma^2(\mathbf{X}^\top\mathbf{X})^{-1}.$$

- Since the expression of $\mathrm{E}(\beta|\sigma^2, \boldsymbol{y}, \mathbf{X})$ does not contain σ^2, the law of iterate expectation implies that $\mathrm{E}(\beta|\boldsymbol{y}, \mathbf{X}) = \frac{g}{g+1}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{y}$.
- For the posterior mean of $\boldsymbol{\beta}$, the hyperparameter g controls the shrinkage from the OLS estimator $\widehat{\boldsymbol{\beta}}$ towards zero. If $g \to +\infty$, then $\mathrm{E}(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X}) \to \widehat{\boldsymbol{\beta}}$. If $g \to 0$, then $\mathrm{E}(\boldsymbol{\beta}|\sigma^2, \boldsymbol{y}, \mathbf{X}) \to 0$.
- The hyperparameter g also controls the conditional posterior variance $\mathrm{Var}(\beta|\sigma^2, \boldsymbol{y}, \mathbf{X})$. If $g \to +\infty$, then $\mathrm{Var}(\beta|\sigma^2, \boldsymbol{y}, \mathbf{X})$ recovers the variance of the OLS estimator.

- In frequentist statistics, there are many model checking and diagnostic techniques in standard linear regression analysis. For example, we can use residuals to verify the normality assumption and homoscedasticity.

- In Bayesian statistics, we can perform model checking using the posterior predictive distribution of y.

- For example, we can use the posterior predictive distribution to check outliers. For each observed y_i, we can plot the histogram from the predictive distribution $p(\tilde{y}|\boldsymbol{y}, \mathbf{X}, \boldsymbol{x}^* = \boldsymbol{x}_i)$, and check whether the observed value y_i falls in the tail part of the histogram. If so, then y_i can be classified as a potential outlier.

10.2 Bayesian Regression Models

Example: Bird extinction data

- The dataset is from the package `LearnBayes`. Measurements on breeding pairs of landbird species were collected from 16 islands around Britain over the course of several decades.

- The variables in the datasets are: `time`, the average time of extinction on the islands where it appeared; `nesting`, the average number of nesting pairs; `size`, the size of the species (large or small), and `status`, the migratory status of the species (migrant or resident).

- The objective is to fit a model that describes the variation in the time of extinction of the bird species in terms of the covariates `nesting`, `size`, and `status`.

10.2 Bayesian Regression Models

Example: Bird extinction data

- Since the response variable `time` displays a strong right-skewness, we take logarithm transformation and use log(time) as the response variable, denoted by y.

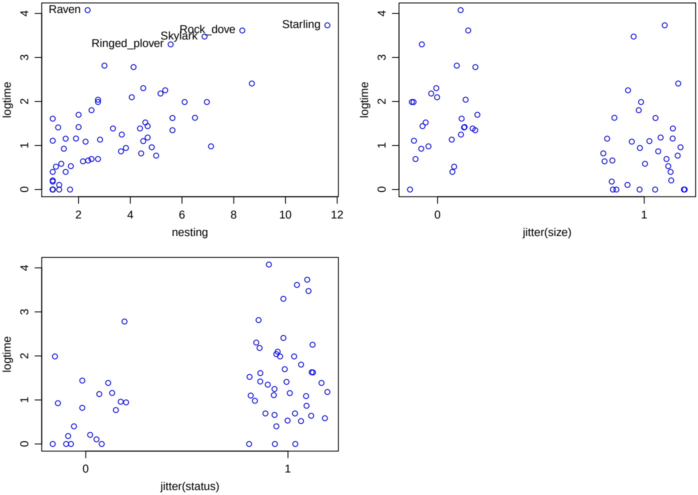
- `nesting` is a continuous variable (x_1). `size` (x_2) and `status` (x_3) are categorical variables, both taking values in $\{0, 1\}$.

- We fit a linear regression model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \epsilon$.

- First, we can plot the scatter plot from y against each covariate. For the two categorical covariates, we use the R command `jitter` to jitter the horizontal location of the points so that we can see overlapping points.

10.2 Bayesian Regression Models

Example: Bird extinction data



10.2 Bayesian Regression Models

Example: Bird extinction data

- Before we start Bayesian analysis, we first explore the dataset by performing the standard least-square fit of the linear model, using the R function `lm()`.

```
> fit1 <- lm(logtime~nesting+size+status,data=birdextinct,x=TRUE,y=TRUE)
> summary(fit1)

Call:
lm(formula = logtime ~ nesting + size + status, data = birdextinct,
    x = TRUE, y = TRUE)

Residuals:
    Min      1Q  Median      3Q     Max
-1.8410 -0.2932 -0.0709  0.2165  2.5167

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.43087    0.20706   2.081 0.041870 *
nesting      0.26501    0.03679   7.203 1.33e-09 ***
size        -0.65220    0.16667  -3.913 0.000242 ***
status       0.50417    0.18263   2.761 0.007712 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.6524 on 58 degrees of freedom
Multiple R-squared:  0.5982,    Adjusted R-squared:  0.5775
F-statistic: 28.79 on 3 and 58 DF,  p-value: 1.577e-11
```

10.2 Bayesian Regression Models

Example Bird extinction data

- From the summary, we can see that the t tests for the coefficients of all three covariates are significant at level 0.01.
- Now we perform Bayesian analysis. We make the following choices for the hyperparameters:

$$\begin{aligned}&\boldsymbol{\mu}_0 = 0, \quad \boldsymbol{\Sigma}_0 = g(\mathbf{X}^\top \mathbf{X})^{-1}, \quad g = 100,\\ &\nu_0 = 1, \quad \sigma_0^2 = 0.01.\end{aligned}$$

- This means that we use the Zellner's g-prior for $p(\boldsymbol{\beta}|\sigma^2, \mathbf{y}, \mathbf{X})$. We choose g to be large, such that the influence from prior is small.
- When we construct the design matrix $\mathbf{X}$, we need to add a column of "1"s because the model has an intercept term ($p = 4$).

- We first sample $T = 10^4$ of σ^2 from Inv-Gamma $\left(\frac{\nu_0 + n}{2},\ \frac{\nu_0\sigma_0^2 + S_0}{2}\right)$, and then sample $T = 10^4$ of β from $\mathrm{N}\left(\frac{g}{g+1}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{y},\ \frac{g}{g+1}\sigma^2\left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\right)$.

- For predictive distribution of y, we simulate additional $T = 10^4$ errors from the normal distribution with mean zero and variance equal to the $T = 10^4$ σ^2's drawn from the posterior.

- The R codes can be found in the following slides.

10.2 Bayesian Regression Models

Example Bird extinction data

```
T <- 10^4
nu0 <- 1
sigma02 <- 0.01
g <- 100     # Zellner's g-prior

y <- logtime
X <- cbind(1,nesting,size,status)
n <- length(y)
p <- ncol(X)

(beta.ols <- solve(t(X)%*%X, t(X)%*%y))
beta.tilde <- g/(g+1) * beta.ols
S0 <- t(y)%*%y - t(beta.tilde)%*%((g+1)/g*t(X)%*%X)%*%beta.tilde

set.seed(4234)
sigma2.samples <- rinvgamma(T, shape=(nu0+n)/2, rate=(nu0*sigma02+S0)/2)
beta.samples <- matrix(0, nrow=T, ncol=p)
for(i in 1:T) {
    beta.samples[i,] <- rmvnorm(1, mean=beta.tilde,
                       sigma=sigma2.samples[i]*g/(g+1)*solve(t(X)%*%X))
}
```

10.2 Bayesian Regression Models

Example Bird extinction data

- The posterior means of β are close to the OLS estimates, but have been shrunk slightly towards zero (which is the prior mean of β).

```
> beta.ols
              [,1]
         0.4308716
nesting  0.2650140
size    -0.6521982
status   0.5041655

> apply(beta.samples,2,mean)
[1]  0.4266432  0.2623187 -0.6457246  0.4978518
> apply(beta.samples,2,sd)
[1] 0.20704789 0.03672898 0.16842373 0.18094619
> apply(beta.samples,2,quantile,prob=c(.05,.5,.95))
          [,1]      [,2]       [,3]      [,4]
5%  0.08465528 0.2025567 -0.9203984 0.2013241
50% 0.42647050 0.2621077 -0.6461509 0.4994933
95% 0.76554919 0.3224902 -0.3696134 0.7976042
```

10.2 Bayesian Regression Models

Example: Bird extinction data

- We can search for the outliers using the predictive distribution of y. We predict y_i at each observed x_i and construct the 90% predictive CI. Those observed y_i's outside the CIs are outliers.

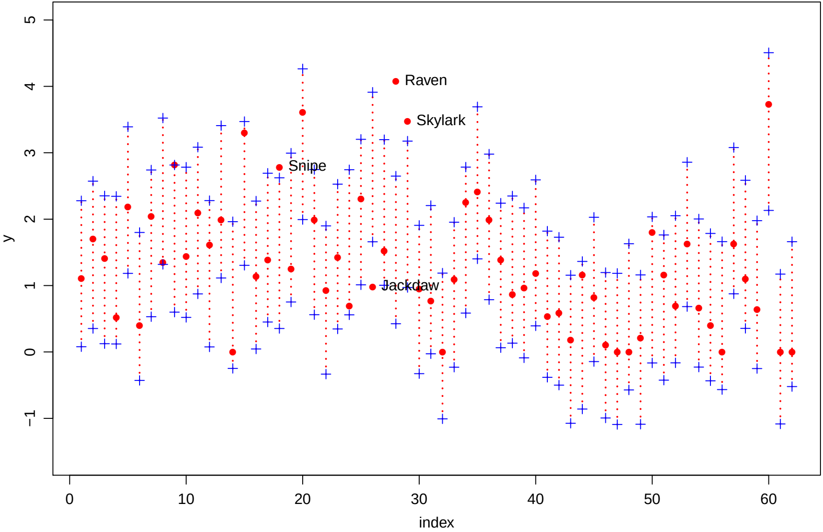
```
error.pred <- matrix(0, nrow=T, ncol=n)    # prediction error
for (i in 1:T) {
    error.pred[i,] <- rnorm(n,mean=0,sd=sqrt(sigma2.samples[i]))
}

# y.pred are the draws from the predictive distribution
y.pred <- beta.samples%*%t(X) + error.pred
y.pred.CI <- matrix(0,nrow=n,ncol=2)
for (j in 1:n) {
    y.pred.CI[j,] <- quantile(y.pred[,j],prob=c(0.05,0.95))
}
```

- The plot on the next slide shows that compared to the prediction based on the linear model, snipe, raven, and skylark have significantly longer average time of extinction, while jackdaw has significantly shorter average time of extinction.

10.2 Bayesian Regression Models

Example: Bird extinction data



Outline

Overview

Bayesian Regression Models

Bayesian Model Comparison

Variational Bayes

Software related to Bayesian Inference

10.3 Bayesian Model Comparison

- In Bayesian statistics, a model is the specification of probability distribution for the observed data $\boldsymbol{y}$.
- The Bayesian framework allows comparisons between different models. For example, for count data, we can fit either a Poisson model, or a negative binomial model. In linear regression models, if there are two predictor variables x_1 and x_2, we may consider the model that regresses y on either $\{x_1\}$, $\{x_2\}$, $\{x_1, x_2\}$, or $\emptyset$ (just the intercept term).
- There are many ways to perform Bayesian model comparison, depending on the nature of models themselves and the objective of interest.
- We introduce several important concepts first.

10.3 Bayesian Model Comparison

Models

- Suppose that we have two different models H_1 and H_2. On the model H_j ($j = 1, 2$), we write the parameter vector as θ_j.
- Based on the observed data $\boldsymbol{y}$, our objective is to decide which model gives a better fit to the data.
- We write the likelihood function $p(\boldsymbol{y}|\theta_j)$ for the model H_j, $j = 1, 2$.
- We assign the prior density $p(\theta_j|H_j)$ on the model H_j, $j = 1, 2$.
- We need to further specify the prior probabilities of the two models, i.e. $p(H_1)$ and $p(H_2)$, which satisfy $p(H_1) + p(H_2) = 1$.
- Thus, we have a unified hierarchical model that consists of two submodels H_1 and H_2:

$$\boldsymbol{y}|\theta_j \sim p(\boldsymbol{y}|\theta_j), \quad \theta_j|H_j \sim p(\theta_j|H_j), \quad H_j \sim p(H_j), \quad j = 1, 2.$$

10.3 Bayesian Model Comparison

Odds

- In statistics, odds are used to describe the ratio of the probabilities of two events. The odds of event A versus event B is given by $p(A)/p(B)$.
- Therefore, the prior odds of H_1 versus H_2 are $p(H_1)/p(H_2)$, where $p(H_j)$ is the prior probability of the model H_j, $j = 1, 2$.
- Similarly, the posterior odds of H_1 versus H_2 are $p(H_1|\boldsymbol{y})/p(H_2|\boldsymbol{y})$, where $p(H_j|\boldsymbol{y})$ is the posterior probability of the model H_j, $j = 1, 2$.
- From the hierarchical model structure, using the Bayes' theorem, the posterior probability $p(H_j|\boldsymbol{y})$ can be calculated by

$$\begin{aligned} p(H_j|\boldsymbol{y}) &= \frac{p(\boldsymbol{y}|H_j)p(H_j)}{p(\boldsymbol{y}|H_1)p(H_1) + p(\boldsymbol{y}|H_2)p(H_2)} && (3) \\ &= \frac{p(\boldsymbol{y}|H_j)p(H_j)}{p(\boldsymbol{y}|H_1)p(H_1) + p(\boldsymbol{y}|H_2)p(H_2)} && (4) \end{aligned}$$

10.3 Bayesian Model Comparison

Bayes Factor

- The marginal probability $p(\boldsymbol{y}|H_j)$ $(j = 1, 2)$ is

$$p(\boldsymbol{y}|H_j) = \int p(\boldsymbol{y}|\theta_j)p(\theta_j|H_j)\mathrm{d}\theta_j. \tag{5}$$

- Using (3) and (5), the posterior odds of H_1 versus H_2 is

$$\begin{aligned}\frac{p(H_1|\boldsymbol{y})}{p(H_2|\boldsymbol{y})} &= \frac{p(\boldsymbol{y}|H_1)p(H_1)}{p(\boldsymbol{y}|H_2)p(H_2)} \\ &= \frac{p(\boldsymbol{y}|H_1)}{p(\boldsymbol{y}|H_2)} \times \frac{p(H_1)}{p(H_2)} \\ &= \frac{\int p(\boldsymbol{y}|\theta_1)p(\theta_1|H_1)\mathrm{d}\theta_1}{\int p(\boldsymbol{y}|\theta_2)p(\theta_2|H_2)\mathrm{d}\theta_2} \times \frac{p(H_1)}{p(H_2)} \end{aligned} \tag{6}$$

- The blue part $\frac{p(\boldsymbol{y}|H_1)}{p(\boldsymbol{y}|H_2)}$ is called Bayes factor. Equation (6) reveals that

$$\text{posterior odds} = \text{Bayes factor} \times \text{prior odds}.$$

- The Bayes factor B_{12} is defined by

$$B_{12} = \frac{\int p(\boldsymbol{y}|\theta_1)p(\theta_1|H_1)\mathrm{d}\theta_1}{\int p(\boldsymbol{y}|\theta_2)p(\theta_2|H_2)\mathrm{d}\theta_2}. \tag{7}$$

- If we treat H_1 and H_2 as two hypotheses, then the Bayes factor is closely related to the concept of likelihood ratio.
- For example, if H_j only contains 1 parameter value θ_j $(j = 1, 2)$, i.e. $p(\theta_j|H_j)$ is a point mass at θ_j, then the Bayes factor simplifies to $B_{12} = p(\boldsymbol{y}|\theta_1)/p(\boldsymbol{y}|\theta_2)$, which is the likelihood ratio.

Bayes Factor

- In general, if H_j contains an (open) space of parameters, then B_{12} can be viewed as the ratio of "weighted" likelihoods in the models H_1 versus H_2. In this case, In this case, the Bayes factor depends on the prior density $p(\theta_j|H_j)$ and cannot be regarded as a measure of the relative support for the model provided solely by the data.
- However, sometimes, B_{12} will be relatively little affected within reasonable limits by the choice of $p(\theta_j|H_j)$ $(j = 1, 2)$. When this is so, B_{12} is a reasonably objective criterion to compare the strength of two models.
- If $B_{12} > 1$, then the model H_1 is more strongly supported by the data than H_2, and vice versa.
- People have given (different) scales for interpretation of the Bayes factor. In general, $B_{12} > 10$ is considered as "strong evidence" favoring model H_1; $B_{12} > 100$ is considered "decisive".

Example: Count data

- Suppose that we observe an i.i.d. sample of count data $\boldsymbol{y} = \{0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 2, 2, 3, 4, 4, 4, 5, 6, 7\}$, with sample size $n = 20$ and $\sum_{i=1}^{n} y_i = 42$.
- We consider two models, the geometric model and the Poisson model.
- For model H_1, $y_i \sim \text{Geo}(\theta_1)$ with density $p(y_i|\theta_1) = (1 - \theta_1)\theta_1^{y_i}$. We assign a $\text{Uniform}(0, 1)$ prior on θ_1.
- For model H_2, $y_i \sim \text{Poisson}(\theta_2)$ with density $p(y_i|\theta_2) = \theta_2^{y_i} \mathrm{e}^{-\theta_2}/y_i!$. We assign a $\text{Gamma}(2, 1)$ prior on θ_2.
- We compute the Bayes factor of model H_1 versus model H_2.

10.3 Bayesian Model Comparison

Example: Count data

- For model H_1, the marginal probability is

$$
\begin{aligned}
p(\boldsymbol{y}|H_1) &= \int_0^1 p(\boldsymbol{y}|\theta_1)p(\theta_1|H_1)\mathrm{d}\theta_1 \\
&= \int_0^1 \prod_{i=1}^{n}(1-\theta_1)\theta_1^{y_i} \cdot 1\mathrm{d}\theta_1 \\
&= \int_0^1 (1-\theta_1)^n \theta_1^{\sum_{i=1}^{n} y_i}\mathrm{d}\theta_1 \\
&= \mathrm{B}\left(\sum_{i=1}^{n} y_i + 1, n+1\right) = \mathrm{B}(43, 21) \\
&= 1.724 \times 10^{-18}.
\end{aligned}
$$

10.3 Bayesian Model Comparison

Example: Count data

- For model H_2, the marginal probability is

$$
\begin{aligned}
p(\boldsymbol{y}|H_2) &= \int_0^1 p(\boldsymbol{y}|\theta_2)p(\theta_2|H_2)\mathrm{d}\theta_2 \\
&= \int_0^1 \prod_{i=1}^n \frac{\theta_2^{y_i}\mathrm{e}^{-\theta_2}}{y_i!} \cdot \theta_2\mathrm{e}^{-\theta_2}\mathrm{d}\theta_2 \\
&= \frac{1}{\prod_{i=1}^n y_i!}\int_0^1 \theta_2^{\sum_{i=1}^n y_i+1}\mathrm{e}^{-(n+1)\theta_2}\mathrm{d}\theta_2 \\
&= \frac{\Gamma(\sum_{i=1}^n y_i + 2)}{(n+1)^{\sum_{i=1}^n y_i+2}\prod_{i=1}^n y_i!} \\
&= \frac{\Gamma(44)}{21^{44}\prod_{i=1}^n y_i!} \\
&= 2.778 \times 10^{-20}.
\end{aligned}
$$

- Therefore, the Bayes factor of model H_1 versus model H_2 is

$$B_{12} = \frac{p(\boldsymbol{y}|H_1)}{p(\boldsymbol{y}|H_2)} = 62.07.$$

- Since $B_{12} > 10$, this implies that the count data provides strong evidence in support of the geometric model over the Poisson model.
- But since $B_{12} < 100$, the evidence is not decisive.
- We should note that the calculation of $p(\boldsymbol{y}|H_1)$ and $p(\boldsymbol{y}|H_2)$ depends heavily on the prior distributions we assign on the parameters.

Model comparison in linear regression

- We can consider choosing the best combination of predictor variables in linear regression models (and in generalized linear models as well).
- For the linear model $y = \beta_1 x_1 + \ldots + \beta_p x_p + \epsilon$, we may exclude some insignificant predictor variables with their coefficients equal to zero.
- For example, if we have 3 predictor variables $\{x_1, x_2, x_3\}$, then we can consider 8 candidate models that includes part of the three covariates:

$$H_1 = \emptyset,\ H_2 = \{x_1\},\ H_3 = \{x_2\},\ H_4 = \{x_3\},\ H_5 = \{x_1, x_2\},$$
$$H_6 = \{x_1, x_3\},\ H_7 = \{x_2, x_3\},\ H_8 = \{x_1, x_2, x_3\}.$$

- If there are p predictor variables, then the total number of candidate models is 2^p.

Model comparison in linear regression

- Under the conjugate normal-inverse gamma prior, we can calculate the marginal probability $p(\boldsymbol{y}|H_j)$ in closed form, though the expression is quite complicated. See Section 9.3.1 in Peter Hoff's book.

- The marginal probabilities of $p(\boldsymbol{y}|H_j)$ can be compared through all models H_j. The model with the largest marginal probability can be selected as the best model.

- However, in practice, often there exist a few candidate models with very close marginal probabilities. Therefore, instead of model selection, we can use Bayesian model averaging, i.e. we account for the posterior uncertainty in different models.

- The posterior from Bayesian model averaging is simply

$$p(\boldsymbol{\beta}, \sigma^2|\boldsymbol{y}) = \frac{\sum_{\text{all } H_j \text{ that contains } \boldsymbol{\beta}} p(\boldsymbol{y}|\boldsymbol{\beta}, \sigma^2, H_j)p(\boldsymbol{\beta}, \sigma^2|H_j)p(H_j)}{\sum_{\text{all } H_j} p(\boldsymbol{y}|H_j)p(H_j)}.$$

10.3 Bayesian Model Comparison

Model comparison for general models

- We can also use the information criterion. Information criteria are related to the predictive performance of models. Some examples are AIC, BIC and DIC.

- Consider a general model $p(\boldsymbol{y}|\theta_j)$ where θ_j is the parameter vector under model H_j.

- Suppose that we consider the model H_j. Let p_j be the number of active parameters in H_j. Let $\widehat{\theta}_{j,\mathrm{mle}}$ be the maximum likelihood estimator of the parameter θ under model H_j. In the linear model, $\widehat{\theta}_{j,\mathrm{mle}}$ is also the OLS estimator from regressing y on those x_k's contained in H_j.

- AIC (Akaike information criterion): The AIC for model H_j is defined by

$$\mathrm{AIC}(H_j) = -2\log p(\boldsymbol{y}|\widehat{\theta}_{j,\mathrm{mle}}) + 2p_j. \tag{8}$$

- The model with smaller AIC value is considered better.
- BIC (Bayesian information criterion): The BIC for model H_j is defined by

$$\mathrm{BIC}(H_j) = -2\log p(\boldsymbol{y}|\widehat{\theta}_{j,\mathrm{mle}}) + p_j \log n. \tag{9}$$

- The model with smaller BIC value is considered better.

10.3 Bayesian Model Comparison

Model comparison in linear regression

- Both AIC and BIC balance the model fitting (evaluated by the negative log likelihood at the MLE) and the number of parameters in the model (evaluated by a multiple of the parameters p_j).
- Both AIC and BIC are motivated by asymptotic analysis as $n \to \infty$. AIC is motivated by asymptotic approximation to the Kullback-Leibler divergence. BIC is motivated by the asymptotic expansion of Bayes factors.
- However, for Bayesian models, especially when the model is hierarchical, sometimes the definition of "p_j" and "n" may not be clear. For example, the number of latent variables can increase with the sample size. Should they be counted as part of p_j?
- The use of MLE in AIC and BIC means that they are more frequentist motivated, as plug-in MLE may underestimate the posterior uncertainty from Bayesian estimation.

10.3 Bayesian Model Comparison

Model comparison in linear regression

- The deviance information criterion (DIC) is a Bayesian version of AIC:

$$\mathrm{DIC}(H_j) = -2 \log p(\boldsymbol{y} | \widehat{\theta}_{j,\mathrm{bayes}}) + 2 p_{j,\mathrm{DIC}}. \tag{10}$$

- $\widehat{\theta}_{j,\mathrm{bayes}}$ is the posterior mode of θ_j under the model H_j.
- $p_{j,\mathrm{DIC}}$ is the effective number of parameters in the model H_j. It has two different definitions.
- Version 1

$$p_{j,\mathrm{DIC}} = 2 \left[\log p(\boldsymbol{y} | \widehat{\theta}_{j,\mathrm{bayes}}) - \mathrm{E}_{\cdot|\boldsymbol{y}} \log p(\boldsymbol{y} | \theta_j) \right]. \tag{11}$$

$\mathrm{E}_{\cdot|\boldsymbol{y}}$ is the expectation under the posterior distribution $p(\theta_j | \boldsymbol{y})$, with respect to θ_j.

- Version 2

$$p_{j,\mathrm{DIC}} = 2\mathrm{Var}_{\cdot|\boldsymbol{y}} \log p(\boldsymbol{y}|\theta_j). \tag{12}$$

  $\mathrm{Var}_{\cdot|\boldsymbol{y}}$ is the expectation under the posterior distribution $p(\theta_j|\boldsymbol{y})$, with respect to θ_j.

- The two versions agree when the dimension of θ_j is fixed and the sample size n goes to infinity.

- For linear regression model, both versions of $p_{j,\mathrm{DIC}}$ are equal to p_j, the number of predictor variables in model H_j.

- In practice, Version 1 is numerically more stable, while Version 2 has the advantage of always being positive.

Model comparison in linear regression

- In practice, it is convenient to estimate the two versions of $p_{j,\mathrm{DIC}}$ using Monte Carlo approximation.
- Suppose that we have draws $\theta_j^{(1)}, \ldots, \theta_j^{(S)}$ from the posterior $p(\theta_j|\boldsymbol{y})$. Then we can estimate $p_{j,\mathrm{DIC}}$ in Version 1 and Version 2 by

$$\hat{p}_{j,\mathrm{DIC}} = 2\left[\log p(\boldsymbol{y}|\widehat{\theta}_{j,\mathrm{bayes}}) - \frac{1}{S}\sum_{s=1}^{S}\log p(\boldsymbol{y}|\theta_j^{(s)})\right],$$

$$\text{and } \hat{p}_{j,\mathrm{DIC}} = \frac{2}{S-1}\sum_{s=1}^{S}\left[\log p(\boldsymbol{y}|\theta_j^{(s)}) - \frac{1}{S}\sum_{s=1}^{S}\log p(\boldsymbol{y}|\theta_j^{(s)})\right]^2,$$

  respectively.

- We can plug $\hat{p}_{j,\mathrm{DIC}}$ in the expression of $\mathrm{DIC}(H_j)$ to calculate the DIC. The model with smaller DIC value is preferred.

Model comparison in linear regression

- DIC values are often reported in the outcomes of MCMC softwares such as OpenBUGS and JAGS.

- There are also other criteria for model comparison, such as WAIC (Watanabe-Akaike information criterion, or widely applicable information criterion), WBIC(Watanabe Bayesian information criterion), and LOO-CV (leave-one-out cross validation). For more details, you can find in Chapter 7 of the BDA3 book.

Outline

Overview

Bayesian Regression Models

Bayesian Model Comparison

Variational Bayes

Software related to Bayesian Inference

Kullback-Leibler Divergence

- We start with the concept of Kullback-Leibler (KL) divergence. For two probability density functions $f(x)$ and $g(x)$, we define the KL divergence as

$$\mathrm{KL}(f||g) = \int f(x) \log \frac{f(x)}{g(x)} \mathrm{d}x.$$

- The KL divergence measures the difference between $f(x)$ and $g(x)$. Its value is always nonnegative: Since the logarithm function is a concave function, we can apply Jensen's inequality to derive that

$$\begin{aligned}\mathrm{KL}(f||g) &= \int f(x) \log \frac{f(x)}{g(x)} \mathrm{d}x = \mathrm{E}_f \left[\log \frac{f(x)}{g(x)} \right] \\ &= -\mathrm{E}_f \left[\log \frac{g(x)}{f(x)} \right] \geq -\log \mathrm{E}_f \left[\frac{g(x)}{f(x)} \right] \\ &= -\log \int f(x) \cdot \frac{g(x)}{f(x)} \mathrm{d}x = -\log \int g(x) \mathrm{d}x = -\log 1 = 0.\end{aligned}$$

10.4 Variational Bayes

Kullback-Leibler Divergence

- The derivation in the last slide also shows that the inequality is an equality if and only if the ratio $g(x)/f(x)$ is a constant, i.e. $g(x) \equiv f(x)$ for (almost surely) all x. Essentially, $\mathrm{KL}(f||g)$ is only zero when $g \equiv f$. If $g \neq f$, then $\mathrm{KL}(f||g) > 0$.

- The KL divergence is asymmetric, since $\mathrm{KL}(f||g) \neq \mathrm{KL}(f||g)$. Therefore, it is called divergence instead of distance.

- The idea of variational Bayes method is to use the KL divergence as a criterion function. Given a posterior density $p(\theta|\boldsymbol{y})$, we find the density $q(\theta)$ within some class of densities, such that $\mathrm{KL}(q||p)$ is minimized.

10.4 Variational Bayes

- Suppose that $p(\theta|\boldsymbol{y})$ is a posterior density of θ given the observed data $\boldsymbol{y}$. Suppose that $\mathcal{G}$ is a class of probability density functions.

- The variational Bayes (VB) method finds the optimal density function $g(\theta)$ from $\mathcal{G}$, such that $\mathrm{KL}(q||p)$ is minimized. In other words, the output of VB is the approximate posterior $q^*(\theta)$ defined by

$$
\begin{aligned}
q^* &= \arg\min_{q\in\mathcal{G}} \mathrm{KL}(q||p) \\
&= \arg\min_{q\in\mathcal{G}} \int q(\theta)\log\frac{q(\theta)}{p(\theta|\boldsymbol{y})}\mathrm{d}\theta. \qquad (13)
\end{aligned}
$$

- Warning: In this chapter, when we write $\mathrm{KL}(q||p)$, the "p" here refers exclusively to the posterior density $p(\theta|\boldsymbol{y})$, not the prior density $p(\theta)$.

10.4 Variational Bayes

- Like normal approximation, VB gives an approximate posterior to the truth $p(\theta|\boldsymbol{y})$. In general, $q^*(\theta)$ is not $p(\theta|\boldsymbol{y})$.

- VB has been widely used in Bayesian statistics and machine learning. There are several practical motivations for using VB, instead of other methods:

  - The model structure is complicated, such as multi-layer hierarchical models. Conjugate priors are generally unavailable.

  - The dimension of parameters (including latent variables) is high, making MCMC algorithms such as MH algorithms inefficient in computation.

- Basic idea of VB: Replace the difficult posterior sampling problem by the relatively easier optimization problem in (13).

Evidence lower bound

- We use the notation $\mathrm{E}_f(\cdot)$ to denote the expectation under a generic density f. We can rearrange some terms in the definition of KL divergence in (13):

$$\begin{aligned}\mathrm{KL}(q||p) &= \int q(\theta)\log\frac{q(\theta)}{p(\theta|\boldsymbol{y})}\mathrm{d}\theta = \mathrm{E}_q\left[\log\frac{q(\theta)}{p(\theta|\boldsymbol{y})}\right] \\ &= \mathrm{E}_q[\log q(\theta)] - \mathrm{E}_q[\log p(\theta|\boldsymbol{y})] \\ &= \mathrm{E}_q[\log q(\theta)] - \mathrm{E}_q[\log p(\theta,\boldsymbol{y})] + \mathrm{E}_q[\log p(\boldsymbol{y})] \\ &= \mathrm{E}_q[\log q(\theta)] - \mathrm{E}_q[\log p(\theta,\boldsymbol{y})] + \log p(\boldsymbol{y}), \qquad (14)\end{aligned}$$

where the last step is because the posterior $p(\theta|\boldsymbol{y}) = p(\theta,\boldsymbol{y})/p(\boldsymbol{y})$, and $p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\theta)p(\theta)\mathrm{d}\theta$ is the marginal probability of $\boldsymbol{y}$ and does not depend on θ.

10.4 Variational Bayes

Evidence lower bound

- In (14), since the last term $\log p(\boldsymbol{y})$, we can omit it from the optimization problem and define

$$\mathrm{ELBO}(q) = \mathrm{E}_q[\log p(\theta, \boldsymbol{y})] - \mathrm{E}_q[\log q(\theta)]$$

  as the evidence lower bound (ELBO), or variational lower bound.

- The optimization problem in (13) can be then equivalently written as

$$q^* = \arg\max_{q \in \mathcal{G}} \mathrm{ELBO}(q).$$

10.4 Variational Bayes

Evidence lower bound

- The name evidence lower bound (ELBO) comes from the fact that it serves as a lower bound to the log of marginal probability. From (14), we can see that

$$\begin{aligned} \log p(\boldsymbol{y}) &= \mathrm{KL}(q||p) + \mathrm{E}_q[\log p(\theta, \boldsymbol{y})] - \mathrm{E}_q[\log q(\theta)] \\ &= \mathrm{KL}(q||p) + \mathrm{ELBO}(q) \\ &\geq \mathrm{ELBO}(q), \end{aligned}$$

  since the KL divergence $\mathrm{KL}(q||p) \geq 0$.

- We can further rewrite the ELBO using the fact $p(\theta, \boldsymbol{y}) = p(\boldsymbol{y}|\theta)p(\theta)$ (likelihood times prior).

$$\begin{aligned}
\text{ELBO}(q) &= \mathrm{E}_q[\log(p(\boldsymbol{y}|\theta)p(\theta)] - \mathrm{E}_q[\log q(\theta)] \\
&= \mathrm{E}_q[\log(p(\boldsymbol{y}|\theta)] + \mathrm{E}_q[p(\theta)] - \mathrm{E}_q[\log q(\theta)] \\
&= \mathrm{E}_q[\log(p(\boldsymbol{y}|\theta)] - \mathrm{E}_q\left[\log \frac{q(\theta)}{p(\theta)}\right] \\
&= \mathrm{E}_q[\log(p(\boldsymbol{y}|\theta)] - \mathrm{KL}(q||\text{prior}).
\end{aligned}$$

- We use $\mathrm{KL}(q||\text{prior})$ to distinguish it from $\mathrm{KL}(q||p)$, where p refers exclusively to the posterior.

10.4 Variational Bayes

Evidence lower bound

- The expression above shows that to maximize the ELBO over q, we need to
  - (i) make the expected log-likelihood ($\mathrm{E}_q[\log(p(\boldsymbol{y}|\theta)])$ as large as possible;
  - (ii) make the KL divergence from q to the prior ($\mathrm{KL}(q||\mathrm{prior})$) as small as possible.

- As a result, the optimal choice of q balances between the likelihood and the prior, within a prespecified class of densities $\mathcal{G}$.

- It is now clear that the class $\mathcal{G}$ is crucial for the success of variational Bayes. The VB solution $q^*(\theta)$ will be an accurate approximation to the true posterior $p(\theta|\boldsymbol{y})$ if $\mathcal{G}$ is chosen as a rich class.

10.4 Variational Bayes

Choice of density class

- Practically, there are at least two popular classes of densities used in variational Bayes literature: the mean field variational Bayes, and the Gaussian variational Bayes.

- Mean field variational Bayes: Suppose that the parameter θ can be partitioned into components/blocks $\theta = (\theta_1, \ldots, \theta_K)$. Then the mean field variational Bayes assumes that $\mathcal{G}$ consists of all possible density q in the following format

$$q(\theta) = \prod_{i=1}^{K} q_i(\theta_i).$$

- In other words, the mean field variational Bayes assumes that the joint distribution $q(\theta)$ can be factored into the product of K independent marginal distributions for $\theta_1, \ldots, \theta_K$.

- Usually there is no further assumption on the form of q_i's. Their forms are determined by the optimization problem in (13).

10.4 Variational Bayes

Mean field variational Bayes

- The mean field variational Bayes is widely used in many applications. We will provide a normal model example to illustrate the implementation of mean field variational Bayes.

- The mean field variational Bayes is most suitable if the posterior $p(\theta|\boldsymbol{y})$ can also be approximately factorized into the product of independent marginal distributions, under the same partition as in the mean field variational Bayes. How to best partition the parameter θ into blocks of $\theta_1, \dots, \theta_K$ depends heavily on the model structure, and therefore, heavily affects the accuracy of mean field VB approximation.

- If the Bayesian model (even hierarchically) is specified by exponential family distributions, then it is likely that the q_i's are also exponential family distributions, which makes the mean field VB computation easier.

10.4 Variational Bayes

Gaussian variational Bayes

- The Gaussian variational Bayes instead assumes that the class $\mathcal{G}$ consists of multivariate normal distributions, i.e. for any $q \in \mathcal{G}$,

$$q(\theta) = \text{density of Multivariate N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

- Gaussian variational Bayes relies less on the specific model structure. It is more suitable for variational Bayes inference on models with very complex structures, such as deep neural networks (DNN). Some recent development in VB relies on Gaussian VB, such as ADVI (automatic differentiation variational inference).

- Gaussian variational Bayes requires that all parameters to be transformed to the whole real line. However, it should be noted that Gaussian variational Bayes is fundamentally different from normal approximation.

10.4 Variational Bayes

Mean field variational Bayes

- Back to mean field variational Bayes using $q(\theta) = \prod_{i=1}^{K} q_i(\theta_i)$. We now consider the practical implementation of solving the optimization problem in (13).
- Since we can write $q^*(\theta) = \prod_{i=1}^{K} q_i^*(\theta_i)$, we only need to find individual $q_i^*(\theta_i)$. Denote $\theta_{-i} = (\theta_1, \ldots, \theta_{i-1}, \theta_i, \ldots, \theta_K)$.
- For mean field VB, the ELBO can be written as

$$\begin{aligned} \mathrm{ELBO}(q) &= \mathrm{E}_q[\log p(\theta, \boldsymbol{y})] - \mathrm{E}_q[\log q(\theta)] \\ &= \mathrm{E}_q[\log p(\theta_i | \theta_{-i}, \boldsymbol{y})] + \mathrm{E}_q[\log p(\theta_{-i}, \boldsymbol{y})] \\ &\qquad - \sum_{i=1}^{K} \mathrm{E}_{q_i}[\log q_i(\theta_i)]. \end{aligned} \tag{15}$$

10.4 Variational Bayes

Mean field variational Bayes

- We first focus on only θ_i. To maximize the ELBO with respect to θ_i, from the right-hand side of (15), it suffices to maximize

$$\mathrm{E}_q[\log p(\theta_i|\theta_{-i},\boldsymbol{y})] - \mathrm{E}_{q_i}[\log q_i(\theta_i)].$$

This is because only these two terms depend on θ_i.

- Now we rewrite this expression as

$$\begin{aligned} &\mathrm{E}_{q_i}\mathrm{E}_{q_{-i}}[\log p(\theta_i|\theta_{-i},\boldsymbol{y})] - \mathrm{E}_{q_i}[\log q_i(\theta_i)] \\ &= -\mathrm{E}_{q_i} \log \frac{q_i(\theta_i)}{\exp\left\{\mathrm{E}_{q_{-i}}[\log p(\theta_i|\theta_{-i},\boldsymbol{y})]\right\}}, \end{aligned}$$

where $\mathrm{E}_{q_{-i}}$ is the expectation with respect to θ_{-i} under the joint density $q_{-i} = \prod_{i'\neq i}^{K} q_{i'}(\theta_{i'})$.

10.4 Variational Bayes

Mean field variational Bayes

- Suppose that we let the density $q_i^* \propto \mathrm{E}_{q^*_{-i}}[\log p(\theta_i|\theta_{-i}, \boldsymbol{y})]$. Then to maximize the ELBO with respect to θ_i, we only need to maximize

$$-\mathrm{E}_{q_i} \log \frac{q_i(\theta_i)}{\exp\left\{\mathrm{E}_{q_{-i}}[\log p(\theta_i|\theta_{-i}, \boldsymbol{y})]\right\}} = -\operatorname{KL}\left(q_i(\theta_i)||q_i^*(\theta_i)\right) + \text{const}$$

  where const is a constant that does not depend on θ.

- Since the KL divergence is nonnegative, to maximize the display above, the optimal choice of q_i is then $q_i = q_i^*$ (such that $-\operatorname{KL}\left(q_i(\theta_i)||q_i^*(\theta_i)\right) = 0$).

- Thus, we have derived that the mean field variational Bayes distribution $q^*(\theta)$ must satisfy that for all $i = 1, \ldots, K$,

$$\boxed{q_i^* \propto \exp\left\{\mathrm{E}_{q^*_{-i}}[\log p(\theta_i|\theta_{-i}, \boldsymbol{y})]\right\}} \qquad (16)$$

- Equation (16) gives an important property of the mean field VB distribution $q^*(\theta) = \prod_{i=1}^{K} q_i^*(\theta_i)$.
- However, since the right-hand side of (16) still depends on the optimal distribution q_{-i}^* on θ_{-i}, it is not an explicit solution of q^* to the VB optimization problem (13).
- Equation (16) provides a way to **iteratively** solve for q^*. We can initialize all the individual distributions of $q_i(\theta_i)$ and use Equation (16) to cycle through $i = 1, \ldots, K$ repeatedly until convergence.

10.4 Variational Bayes

CAVI algorithm (Bishop 2006)

We can now write down a general CAVI algorithm to find the mean field VB solution $q^*(\theta) = \prod_{i=1}^{K} q_i^*(\theta_i)$.

Coordinate Ascent Variational Inference (CAVI) algorithm (Bishop 2006)

1. Initialize $q_i(\theta_i)$, for $i = 1, \ldots, K$.
2. While ELBO(q) has not converged, do
   - For $i = 1, \ldots, K$, set

     $$q_i \propto \mathrm{E}_{q_{-i}}[\log p(\theta_i|\theta_{-i}, \boldsymbol{y})]$$

   - Update the ELBO with

     $$\mathrm{ELBO}(q) = \mathrm{E}_q[\log p(\theta, \boldsymbol{y})] - \mathrm{E}_q[\log q(\theta)]$$

CAVI algorithm (Bishop 2006)

- In the CAVI algorithm, ELBO(q) is a number that keeps on increasing as the algorithm iterates.

- Because the q_i's are usually parametric models (as a result of (16) and Bayesian model structure), the step of updating q_i's requires the updating of some parameters in the distribution q_i's.

- The CAVI algorithm still appears quite abstract. Let us illustrate how to implement it through the normal model example.

10.4 Variational Bayes

Example 1: Normal model

- Suppose that we observe an i.i.d. sample $\boldsymbol{y} = \{y_1, \ldots, y_n\}$ from $\mathrm{N}(\mu, \sigma^2)$.
- We use the same prior as in Chapter 3: $p(\mu, \sigma^2) = p(\mu|\sigma^2)p(\sigma^2)$, and

$$\mu|\sigma^2 \sim \mathrm{N}\left(\mu_0, \frac{\sigma^2}{n_0}\right), \quad \sigma^2 \sim \text{Inv-Gamma}\left(\frac{\nu_0}{2}, \frac{\nu_0\sigma_0^2}{2}\right).$$

- For this example, we know the true posterior in closed form:

$$\theta|\sigma^2, \boldsymbol{y} \sim \mathrm{N}\left(\mu_1, \frac{\sigma^2}{n_1}\right), \quad \sigma^2|\boldsymbol{y} \sim \text{Inv-Gamma}\left(\frac{\nu_1}{2}, \frac{\nu_1\sigma_1^2}{2}\right),$$

$$\mu_1 = \frac{n\bar{y} + n_0\mu_0}{n + n_0}, \qquad \nu_1 = \nu_0 + n,$$

$$n_1 = n + n_0, \qquad \sigma_1^2 = \frac{1}{\nu_1}\left[\nu_0\sigma_0^2 + (n-1)s^2 + \frac{nn_0(\bar{y} - \mu_0)^2}{n + n_0}\right]$$

10.4 Variational Bayes

Example 1: Normal model

- To illustrate the mean field VB, we now consider the natural partition of $\theta = (\mu, \sigma^2)$ into $\theta_1 = \mu$ and $\theta_2 = \sigma^2$, and find an approximation density of the form

$$q(\theta) = q_\mu(\mu) q_{\sigma^2}(\sigma^2).$$

- We emphasize that in this example, the true posterior does not have this factorization, since clearly μ and σ^2 are not independent in the posterior. Therefore, the optimal $q^*(\theta) \neq p(\theta|\boldsymbol{y})$.

- We find q_μ and q_{σ^2} using the CAVI algorithm. The first step is to determine the format of q_μ and q_{σ^2}, using the property in Equation (16).

10.4 Variational Bayes

Example 1: Normal model

- From Chapter 3, we have the following format of the unnormalized posterior density for the normal model.

$$\begin{aligned} p(\mu, \sigma^2 | \boldsymbol{y}) &\propto p(\boldsymbol{y}|\mu, \sigma^2) p(\mu, \sigma^2) \\ &= p(\boldsymbol{y}, \mu, \sigma^2) \\ &= (2\pi)^{-\frac{n+1}{2}} \exp\left\{ -\frac{(\mu - \frac{n\bar{y} + n_0\mu_0}{n+n_0})^2}{2\sigma^2/(n+n_0)} \right\} \\ &\quad \times \frac{(\nu_0\sigma_0^2/2)^{\nu_0/2}}{\Gamma(\nu_0/2)} (\sigma^2)^{-\frac{\nu_0+n+1}{2}-1} \exp\left\{ -\frac{S_1}{2\sigma^2} \right\}, \end{aligned}$$

where

$$S_1 = (n-1)s^2 + \nu_0\sigma_0^2 + \frac{nn_0(\bar{y} - \mu_0)^2}{n+n_0}.$$

- For q_μ, from Equation (16), we have

$$\begin{aligned}
q_\mu^*(\mu) &\propto \exp\left\{\mathrm{E}_{q^*_{\sigma^2}}[\log p(\mu|\sigma^2, \boldsymbol{y})]\right\} \\
&\propto \exp\left\{\mathrm{E}_{q^*_{\sigma^2}}\left[\log\left(\text{const} \times \exp\left\{-\frac{(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0})^2}{2\sigma^2/(n+n_0)}\right\}\right)\right]\right\} \\
&\propto \exp\left\{\mathrm{E}_{q^*_{\sigma^2}}\left[-\frac{(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0})^2}{2\sigma^2/(n+n_0)} + \text{const}\right]\right\} \\
&\propto \exp\left\{-\mathrm{E}_{q^*_{\sigma^2}}\left(\frac{1}{\sigma^2}\right)\frac{(n+n_0)(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0})^2}{2}\right\}.
\end{aligned}$$

- This implies that $q_\mu^*(\mu)$ is the density of

$$\mathrm{N}\left(\frac{n\bar{y}+n_0\mu_0}{n+n_0},\ \frac{1}{(n+n_0)\mathrm{E}_{q^*_{\sigma^2}}\left(\frac{1}{\sigma^2}\right)}\right).$$

Example 1: Normal model

- For q_{σ^2}, from Equation (16), we have

$$
\begin{aligned}
q^*_{\sigma^2}(\sigma^2) &\propto \exp\left\{ \mathrm{E}_{q^*_{-\sigma^2}}[\log p(\sigma^2|\mu, \boldsymbol{y})] \right\} \\
&\propto \exp\left\{ \mathrm{E}_{q^*_\mu}\left[\log\left(\text{const} \times (\sigma^2)^{-\frac{\nu_0+n+1}{2}-1} \right.\right.\right. \\
&\qquad \left.\left.\left. \exp\left\{ -\frac{(n+n_0)(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0})^2 + S_1}{2\sigma^2} \right\} \right) \right] \right\} \\
&\propto \exp\left\{ \mathrm{E}_{q^*_\mu}\left[\left(-\frac{\nu_0+n+1}{2} - 1 \right) \log \sigma^2 \right.\right. \\
&\qquad \left.\left. - \frac{(n+n_0)(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0})^2 + S_1}{2\sigma^2} \right] \right\} \\
&\propto (\sigma^2)^{-\frac{\nu_0+n+1}{2}-1} \exp\left\{ -\frac{(n+n_0)\mathrm{E}_{q^*_\mu}\left(\mu - \frac{n\bar{y}+n_0\mu_0}{n+n_0}\right)^2 + S_1}{2\sigma^2} \right\}.
\end{aligned}
$$

- This implies that $q^*_{\sigma^2}(\sigma^2)$ is the density of

$$\text{Inv-Gamma}\left(\frac{\nu_0 + n + 1}{2},\ \frac{(n + n_0)\mathrm{E}_{q^*_\mu}(\mu - \mu_1)^2 + S_1}{2}\right),$$

  where

$$\begin{aligned} \mu_1 &= \frac{n\bar{y} + n_0\mu_0}{n + n_0}, \\ S_1 &= (n-1)s^2 + \nu_0\sigma_0^2 + \frac{nn_0(\bar{y} - \mu_0)^2}{n + n_0}. \end{aligned}$$

10.4 Variational Bayes

Example 1: Normal model

- To complete the calculation, we need to find the formulas to compute $\mathsf{E}_{q_\mu^*}(\cdot)$ and $\mathsf{E}_{q_{\sigma^2}^*}(\cdot)$ in the expression of $q_\mu^*(\mu)$ and $q_{\sigma^2}^*(\sigma^2)$.
- In the expression of $q_{\sigma^2}^*(\sigma^2)$, clearly, we only need to know $\mathsf{E}_{q_\mu^*}(\mu)$ and $\mathsf{E}_{q_\mu^*}(\mu^2)$. Since $q_\mu^*(\mu)$ is a normal distribution, if the last iteration gives $q_\mu^{(t-1)} = \mathsf{N}(u^{(t-1)}, v^{(t-1)})$, then $\mathsf{E}_{q_\mu^{(t-1)}}(\mu) = u^{(t-1)}$ and $\mathsf{E}_{q_\mu^{(t-1)}}(\mu^2) = (u^{(t-1)})^2 + v^{(t-1)}$.
- In the expression of $\mathsf{E}_{q_\mu^*}(\cdot)$, we need $\mathsf{E}_{q_{\sigma^2}^*}\left(1/\sigma^2\right)$. For $Z \sim \text{Inv-Gamma}(a, b)$, it is known that $1/Z \sim \text{Gamma}(a, b)$ and hence $\mathsf{E}(1/Z) = a/b$. Since $q_{\sigma^2}^*(\sigma^2)$ is an inverse-gamma distribution, if the last iteration gives $q_{\sigma^2}^{(t-1)} = \text{Inv-Gamma}(a^{(t-1)}, b^{(t-1)})$, then $\mathsf{E}_{q_{\sigma^2}^{(t-1)}}(1/\sigma^2) = a^{(t-1)}/b^{(t-1)}$.

10.4 Variational Bayes

Example 1: Normal model

- To complete the CAVI algorithm, we need to find $\text{ELBO}(q^*)$. We assume that $q^*_\mu = \text{N}(u^*, v^*)$, $q^*_{\sigma^2} = \text{Inv-Gamma}(a^*, b^*)$.

$$
\begin{aligned}
&\text{ELBO}(q^*) = \text{E}_{q^*}[\log p(\theta, \boldsymbol{y})] - \text{E}_{q^*}[\log q^*(\theta)] \\
&= \text{E}_{q^*}\Bigg[-\frac{n+1}{2}\log(2\pi) + \frac{\nu_0}{2}\log(\nu_0\sigma_0^2/2) - \log\Gamma(\nu_0/2) + \frac{1}{2}\log n_0 \\
&\quad - \left(\frac{\nu_0+n+1}{2}+1\right)\log\sigma^2 - \frac{(n+n_0)(\mu-\mu_1)^2 + S_1}{2\sigma^2}\Bigg] \\
&\quad - \text{E}_{q^*}[\log q^*(\theta)] \\
&= -\frac{n+1}{2}\log(2\pi) + \frac{\nu_0}{2}\log(\nu_0\sigma_0^2/2) - \log\Gamma(\nu_0/2) + \frac{1}{2}\log n_0 \\
&\quad - \left(\frac{\nu_0+n+1}{2}+1\right)\text{E}_{q^*_{\sigma^2}}[\log\sigma^2] - \frac{\text{E}_{q^*_{\sigma^2}}(1/\sigma^2)}{2}\left[(n+n_0)\text{E}_{q^*_\mu}(\mu-\mu_1)^2 + S_1\right] \\
&\quad + \frac{\text{E}_{q^*_\mu}(\mu-u^*)^2}{2v^*} + \frac{1}{2}\log(2\pi v^*) \\
&\quad - a^*\log b^* + \log\Gamma(a^*) + (a^*+1)\text{E}_{q^*_{\sigma^2}}[\log\sigma^2] + b^*\text{E}_{q^*_{\sigma^2}}[1/\sigma^2]
\end{aligned}
$$

continued on next page...

- The formula to compute ELBO is then

$$
\begin{aligned}
\mathrm{ELBO}(q^*) = & -\frac{n+1}{2}\log(2\pi) + \frac{\nu_0}{2}\log(\nu_0\sigma_0^2/2) - \log\Gamma(\nu_0/2) + \frac{1}{2}\log n_0 \\
& + \left(\frac{\nu_0+n+1}{2} - a^*\right)[\psi(a^*) - \log b^*] \\
& - \frac{a^*}{2b^*}\left[(n+n_0)\left(u^{*2} + v^* - 2\mu_1 u^* + \mu_1^2\right) + S_1\right] \\
& + \frac{1}{2} + \frac{1}{2}\log(2\pi v^*) - a^*\log b^* + \log\Gamma(a^*) + a^*, \qquad (17)
\end{aligned}
$$

  where $\psi(\cdot)$ is the digamma function.

- In the derivation above, we have used the following fact: If $Z \sim \text{Inv-Gamma}(a, b)$, then $1/Z \sim \text{Gamma}(a, b)$, and $\mathrm{E}[\log Z] = -\mathrm{E}[\log(1/Z)] = -[\psi(a) - \log b]$.

10.4 Variational Bayes

Example 1: Normal model

Based on these calculations, we can now describe the CAVI algorithm to find the mean field VB approximation to the posterior $p(\mu, \sigma^2 | \boldsymbol{y})$.

- Initialize $u^{(0)} \in \mathbb{R}$, $v^{(0)} > 0$, $a^{(0)} > 0$, $b^{(0)} > 0$. Set the error tolerance ε to be a small number (such as 10^{-4}). Set $t = 0$.
- While $\left| \text{ELBO}(q^{(t)}) - \text{ELBO}(q^{(t-1)}) \right| > \varepsilon$, do
    - Set $t \leftarrow t + 1$, and

$$\begin{aligned} u^{(t)} &= \mu_1, \quad v^{(t)} = \frac{b^{(t-1)}}{(n + n_0) a^{(t-1)}}, \\ a^{(t)} &= \frac{\nu_0 + n + 1}{2}, \\ b^{(t)} &= \frac{1}{2} \left\{ (n + n_0) \left[(u^{(t)})^2 + v^{(t)} - 2u^{(t)} \mu_1 + \mu_1^2 \right] + S_1 \right\}. \end{aligned}$$

    - Set $\text{ELBO}(q^{(t)})$ using Equation (17), with u^*, v^*, a^*, b^* replaced by $u^{(t)}, v^{(t)}, a^{(t)}, b^{(t)}$.

Example 1: Midge wing length

- We implement this algorithm on the midge wing length data used in Chapter 3. The observations are $\boldsymbol{y} = (1.64, 1.70, 1.72, 1.74, 1.82, 1.82, 1.82, 1.90, 2.08)$, giving $\bar{y} = 1.804$.
- We choose the same prior hyperparameters as in Chapter 3:

$$\mu_0 = 1.9, \quad n_0 = 1, \quad \nu_0 = 1, \quad \sigma_0^2 = 0.01.$$

```
y <- c(1.64, 1.70, 1.72, 1.74, 1.82, 1.82, 1.82, 1.90, 2.08)
n <- length(y)
ybar <- mean(y)
s2 <- var(y)
mu0 <- 1.9
sigma02 <- 0.01
n0 <- 1
nu0 <- 1
mu1 <- (n*ybar+n0*mu0)/(n+n0)
S1 <- (n-1)*s2+nu0*sigma02+n*n0*(ybar-mu0)^2/(n+n0)
```

10.4 Variational Bayes

Example 1: Midge wing length

- We write a function to compute the ELBO.

```
# ELBO function
elbo.normal <- function(u,v,a,b,data) {
    n <- data$n
    n0 <- data$n0
    nu0 <- data$nu0
    sigma02 <- data$sigma02
    mu1 <- data$mu1
    S1 <- data$S1

    -(n+1)/2*log(2*pi)+nu0/2*log((nu0*sigma02)/2)-
    log(gamma(nu0/2))+1/2*log(n0)+
    ((nu0+n+1)/2-a)*(digamma(a)-log(b))-
    1/2*(a/b)*((n+n0)*(u^2+v-2*u*mu1+mu1^2)+S1)+
    1/2+log(2*pi*v)/2-a*log(b)+log(gamma(a))+a
}
```

10.4 Variational Bayes

Example 1: Midge wing length

- For the normal model example, the parameters u in q_μ^* and a in $q_{\sigma^2}^*$ actually do not change with the iterations. So we only need to update v in q_μ^* and b in $q_{\sigma^2}^*$.

```
wingdata <- list(n=n,n0=n0,nu0=nu0,sigma02=sigma02,mu1=mu1,S1=S1)

# u and a are fixed
u <- mu1
a <- (nu0+n+1)/2

# initialize v and b
v <- 1
b <- 1
iter <- 0
elbo.diff <- 1
elbo.old <- elbo.normal(u,v,a,b,data=wingdata)
```

10.4 Variational Bayes

Example 1: Midge wing length

- The CAVI algorithm runs as follows:

```
# find the mean field VB distribution using the CAVI algorithm
while(elbo.diff > 1e-6) {
    iter <- iter + 1
    v <- b/a/(n+n0)
    b <- ((n+n0)*(u^2+v-2*u*mu1+mu1^2)+S1)/2
    elbo.new <- elbo.normal(u,v,a,b,data=wingdata)
    elbo.diff <- abs(elbo.new - elbo.old)
    elbo.old <- elbo.new
    cat("iter",iter,"; elbo=",elbo.new,"\n")
}
```

```
iter= 1 ; elbo= 0.7884674
iter= 2 ; elbo= 3.200674
iter= 3 ; elbo= 3.328391
iter= 4 ; elbo= 3.330097
iter= 5 ; elbo= 3.330112
iter= 6 ; elbo= 3.330112
```

10.4 Variational Bayes

Example 1: Midge wing length

- We can see that the CAVI algorithm has converged in 6 steps.
- The ELBO is increasing with the iterations, which is something we expect to happen. In fact, it can be shown that the ELBO is increasing every step in the CAVI algorithm. This is similar to the EM (Expectation-Maximization) algorithm.
- The final solution of parameters u^*, v^*, a^*, b^* are

```
> u
[1] 1.814
> v
[1] 0.001532503
> a
[1] 5.5
> b
[1] 0.08428252
```

10.4 Variational Bayes

Example 1: Midge wing length

- Therefore, the mean field variational Bayes posterior is

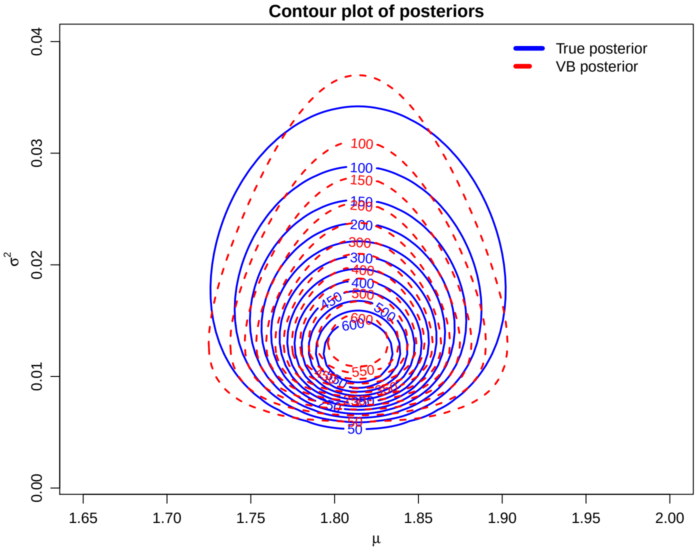
$$
\begin{aligned}
&q^*_{\mu,\sigma^2}(\mu,\sigma^2) = q^*_\mu(\mu) \times q^*_{\sigma^2}(\sigma^2),\\
&q^*_\mu = \mathrm{N}(1.814, 0.0015), \quad q^*_{\sigma^2} = \text{Inv-Gamma}(5, 5, 0.084).
\end{aligned}
$$

- From Chapter 3, the true posterior $p(\mu, \sigma^2|\boldsymbol{y})$ is

$$
\begin{aligned}
&p(\mu, \sigma^2|\boldsymbol{y}) = p(\mu|\sigma^2, \boldsymbol{y})p(\sigma^2|\boldsymbol{y}),\\
&p(\mu|\sigma^2, \boldsymbol{y}) = \mathrm{N}\left(1.814, \frac{\sigma^2}{10}\right), \quad p(\sigma^2|\boldsymbol{y}) = \text{Inv-Gamma}(5, 0.077).
\end{aligned}
$$

- We can plot their contour plots together and see the difference.

Example 1: Midge wing length



10.4 Variational Bayes

Example 1: Midge wing length

- For the normal model, we can also compute the log of marginal probability, $\log p(\boldsymbol{y})$, by using

$$\log p(\boldsymbol{y}) = \log \iint p(\boldsymbol{y}|\mu, \sigma^2) p(\mu|\sigma^2) p(\sigma^2) \mathrm{d}\mu \mathrm{d}\sigma^2.$$

- After some calculation, we can obtain that

$$\begin{aligned} \log p(\boldsymbol{y}) = {} & \log \Gamma\left(\frac{\nu_0 + n}{2}\right) - \log \Gamma\left(\frac{\nu_0}{2}\right) + \frac{\nu_0}{2} \log(\nu_0 \sigma_0^2) \\ & - \frac{\nu_0 + n}{2} \log S_1 - \frac{n}{2} \log \pi + \frac{1}{2} \log n_0 - \frac{1}{2} \log(n + n_0). \end{aligned}$$

10.4 Variational Bayes

Example 1: Midge wing length

- We can use the formula for $\log p(\boldsymbol{y})$ to compute it for the midge wing length data.

```
> (logpy <- log(gamma((nu0+n)/2))-log(gamma(nu0/2))+
+           nu0/2*log(nu0*sigma02)-
+           (nu0+n)/2*log(S1)-n/2*log(pi)+
+           1/2*log(n0)-1/2*log(n+n0))
[1] 3.379277
```

- The final output of $\text{ELBO}(q^*)$ is 3.330 from page 83, which is smaller than $\log p(\boldsymbol{y}) = 3.379$.

- This is not surprising. On page 58, we have shown that the ELBO is a lower bound for the log of marginal probability. Their difference is the KL divergence from the VB posterior $q^*(\cdot)$ to the true posterior $p(\cdot|\boldsymbol{y})$.

- We note that although we can obtain the closed form for $p(\boldsymbol{y})$ for the normal model, this is not generally the case. In most applications, it is unlikely to have analytical solution for $p(\boldsymbol{y})$.

- The CAVI algorithm is guaranteed to converge to a local maximum of ELBO. The ELBO is guaranteed to increase with the iterations. This is similar to the EM algorithm. In fact, there is an intrinsic relation between EM and VB (see the references later).

Summary: Pros and Cons of VB

- The variational Bayes method is now an important method in Bayesian statistics and machine learning. We summarize some pros and cons of using VB.
- Pros of VB
  - Computationally (much) faster than MCMC; sometimes VB is the only feasible choice;
  - Applicable to a wide range of complex models in statistics and machine learning;
  - Suitable for massive data using stochastic optimization methods.

10.4 Variational Bayes

Summary: Pros and Cons of VB

- Cons of VB
  - VB is only an approximation to the true posterior. There is no guarantee of approximation accuracy.
  - Underestimation of posterior uncertainty - the VB posterior variance tends to be smaller than the true posterior variance;
  - Mean field VB typically requires very complicated paper-and-pencil derivations for the forms of VB posteriors, before software implementation.

10.4 Variational Bayes

Further references on VB

- The variational Bayes method has wide applications and is a highly active research area. Some of the models with successful VB applications are
  - Generalized linear models (e.g. linear/logistic/probit/Poisson regression)
  - Mixture models (e.g. Gaussian mixture models)
  - Deep exponential families (e.g. deep latent Gaussian models)
  - Topic models (e.g. latent Dirichlet allocation)
  - Linear dynamic systems (e.g. state space models, hidden Markov models)
  - Gaussian process models
- Some good references for variational Bayes:
  - Chapter 10 of Pattern Recognition and Machine Learning by Christopher Bishop.
  - Variational Inference: A Review for Statisticians by David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. `https://arxiv.org/abs/1601.00670`

Outline

Overview

Bayesian Regression Models

Bayesian Model Comparison

Variational Bayes

Software related to Bayesian Inference

10.5 Software related to Bayesian Inference

- There are several software available for analyzing Bayesian hierarchical models using MCMC methods. Examples are Stan, OpenBUGS and JAGS (Just Another Gibbs Sampler, `http://mcmc-jags.sourceforge.net/`).
- These software generally little input from the user (besides specification of the model), thus allowing the user to focus on data analysis without being encumbered with the MCMC implementation details.
- However, each software has its own limitations (e.g. the types of distributions available) and it may not be possible to fit certain models.
- We will give a brief introduction to Stan and OpenBUGS and demonstrate how they can be used to fit a generalized linear mixed model. Please refer to the websites of the respective software for further information.

10.5.1 Example: Generalized linear mixed model

- The epilepsy data (Thall and Vail, 1990) includes $n = 59$ epileptics who were randomized to a new drug, progabide (Trt=1) or a placebo (Trt=0) in a clinical trial.

- The response y_{ij} is the number of seizures patients have during four follow-up periods. Other information available include number of baseline seizures and age.

```
> df <-  read.table("data_epilepsy.txt", header=TRUE)
> head(df)
   ID Y1 Y2 Y3 Y4 Base Age     Trt Ysum Age10 Base4
1 104  5  3  3  3   11  31 placebo   14   3.1  2.75
2 106  3  5  3  3   11  30 placebo   14   3.0  2.75
3 107  2  4  0  5    6  25 placebo   11   2.5  1.50
4 114  4  4  1  4    8  36 placebo   13   3.6  2.00
5 116  7 18  9 21   66  22 placebo   55   2.2 16.50
6 118  5  2  8  7   27  29 placebo   22   2.9  6.75
```

10.5.1 Example: Generalized linear mixed model

- Let us consider the following covariates:
  1. The log of $\frac{1}{4}$ the number of baseline seizures (Base)
  2. Trt (1 for progabide, 0 for placebo)
  3. Base × Trt
  4. The log of age (Age). We further center Age by replacing Age_i by $\text{Age}_i - \text{mean}(\text{Age})$.
  5. A binary variable V4 which is 1 for the fourth visit and 0 otherwise.

```
n <- dim(df)[1]                           # no. subj
lb4 <- log(df$Base4)                      # log 1/4 Base
trt <- as.numeric(df$Trt == "progabide")  # 1 drug, 0 placebo
lb4trt <- lb4*trt
lage <- log(df$Age)                       # log(Age)
clage <- scale(lage, scale=FALSE)         # center log(Age)
V4 <- c(0,0,0,1)                          # 1 for 4th visit
```

10.5.1 Example: Generalized linear mixed model

- Consider the following Poisson random intercept model: For $i = 1, \ldots, n$, $j = 1, \ldots, 4$,

$$
\begin{aligned}
y_{ij} &\sim \text{Poisson}(\mu_{ij}), \\
\log \mu_{ij} &= \beta_0 + \beta_{\text{Base}} \text{Base}_i + \beta_{\text{Trt}} \text{Trt}_i + \beta_{\text{Base}\times\text{Trt}} \text{Base}_i \times \text{Trt}_i \\
&\quad + \beta_{\text{Age}} \text{Age}_i + \beta_{\text{V4}} \text{V4}_{ij} + b_i, \\
b_i &\sim N(0, \mathrm{e}^{2\zeta}), \\
\beta &\sim N(0, 100) \\
\zeta &\sim N(0, 100)
\end{aligned}
$$

- Let $\mu = [\mu_{ij}]$, $\beta = (\beta_0, \beta_{\text{Base}}, \beta_{\text{Trt}}, \beta_{\text{Base}\times\text{Trt}}, \beta_{\text{Age}}, \beta_{\text{V4}})$ and $b = (b_1, \ldots, b_n)$.
- The unknown parameters in this model are (μ, b, β, ζ).
- Of particular interest are the global parameters β and ζ.

10.5.1 Example: Generalized linear mixed model

- We convert the data into matrix form that is suitable for analysis using in Stan and OpenBUGS.

```
vni <- rep(4, n)                               # no. obs per subj
startindex <- c(0, cumsum(vni)[1:(n-1)]) + 1
endindex <- cumsum(vni)
N <- sum(vni)                                  # total no. of obs
k <- 6                                         # length of beta
y <- as.vector(rbind(df$Y1, df$Y2, df$Y3, df$Y4))
Z <- rep(1, N)
X <- matrix(0, N, k)
for (i in 1:n){
  X[startindex[i]:endindex[i],] <- cbind(rep(1,4), rep(lb4[i],4),
           rep(trt[i],4), rep(lb4trt[i],4), rep(clage[i],4), V4)
}
data <- list(n=n, N=N, k=k, y=y, X=X, Z=Z,
             startindex=startindex, endindex=endindex)
pars <- c("beta", "zeta")
```

10.5.2 Stan

- Stan is an open-source software which provides a language for specifying statistical models and a library of statistical algorithms for computing inferences using those models (http://mc-stan.org/).

- It provides full Bayesian inference with MCMC sampling (No-U-Turn Sampler (NUTS) and Hamiltonian Monte Carlo (HMC)).
- R can be used to interface with Stan. To get started, install RStan by following the instructions at this quick start guide: (https://github.com/stan-dev/rstan/wiki/RStan-Getting-Started.)
  1. Download and install Rtools. Verify that Rtools can be used in R.
  2. Configuration.
  3. Install RStan.

10.5.2 Stan

- Let us fit the GLMM in R using Stan. First we write the model and save it in a file called `model_epilepsy.stan`.
- See http://mc-stan.org/users/documentation/ for further details.

```
data {
 int<lower=0> n; // no. subj
 int<lower=0> N; // no. obs
 int<lower=0> k; // no. fix eff
 int<lower=0> y[N]; // resp
 matrix[N,k] X; // fix eff cov
 vector[N] Z; // rand eff cov
 int<lower=1> startindex[n];
 int<lower=1> endindex[n];
}
parameters {
  vector[k] beta;
  real zeta;
  vector[n] b;
}
```

```
model {
 vector[N] prob;
 zeta ~ normal(0, 10); // 10 is sd
 beta ~ normal(0, 10); // vect form
 for (i in 1:n) {
  b[i] ~ normal(0, exp(zeta));
  for (j in startindex[i]:endindex[i]){
   prob[j] = dot_product(X[j,], beta)
                + Z[j]* b[i];
  }
 }
 y ~ poisson_log(prob);
}
```

10.5.2 Stan

Coding a model in Stan

- The model should be stored in a `.stan` file.
- The model specification consists of three main blocks:
    - `data`: declaration of variables that are read in as data.
    - `parameters`: declaration variables being sampled by Stan's samplers (HMC and NUTS).
    - `model`: optional variable declarations followed by statements that define the model.
- Every variable used must have a declared data type, such as an unconstrained primitive (e.g. `real` for continuous values and `int` for integer values), vector, matrix or array.
- Adding comments: Any characters on a line following two forward slashes (//) is ignored along with the slashes.

10.5.2 Stan

- We can fit the model using MCMC methods via the stan function. By default, the first half of the iterations are discarded as burn-in. The length of burn-in can be adjusted using the warmup argument.
- For diagnosing convergence, it is often useful to run multiple MCMC chains from different starting points. On a multi-core machine, multiple MCMC chains can be executed in parallel by setting cores to the number available on the machine.

```
> library(rstan)
> system.time(fit <- stan(file = "model_epilepsy.stan", pars = pars,
+        data = data, iter = 5000, chains = 2, thin = 1, cores = 1))
    user  system elapsed
 105.81    0.18  106.26

> system.time(fit <- stan(file = "model_epilepsy.stan", pars = pars,
+        data = data, iter = 5000, chains = 2, thin = 1, cores = 2))
   user  system elapsed
   0.11    0.17   55.22
```

10.5.2 Stan

- Use print to view a summary of the fitted model.

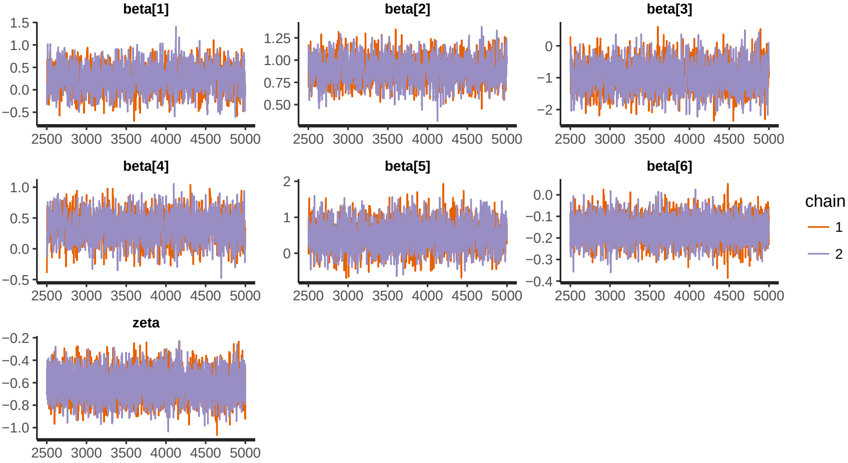
```
> print(fit, digits = 2)
Inference for Stan model: model_epilepsy.
2 chains, each with iter=5000; warmup=2500; thin=1;
post-warmup draws per chain=2500, total post-warmup draws=5000.

           mean se_mean   sd    2.5%     25%     50%     75%   97.5% n_eff Rhat
beta[1]    0.26    0.01 0.27   -0.28    0.08    0.27    0.45    0.76  1233    1
beta[2]    0.89    0.00 0.14    0.62    0.80    0.89    0.98    1.16  1058    1
beta[3]   -0.95    0.01 0.42   -1.76   -1.23   -0.96   -0.67   -0.10  1456    1
beta[4]    0.34    0.01 0.21   -0.08    0.20    0.35    0.48    0.75  1338    1
beta[5]    0.50    0.01 0.37   -0.21    0.25    0.50    0.74    1.23  1425    1
beta[6]   -0.16    0.00 0.05   -0.27   -0.20   -0.16   -0.12   -0.06  5000    1
zeta      -0.63    0.00 0.12   -0.85   -0.71   -0.63   -0.55   -0.39  3340    1
lp__    3208.94    0.15 5.99 3196.68 3205.10 3209.28 3213.16 3219.88  1579    1

Samples were drawn using NUTS(diag_e) at Wed Nov 01 09:48:36 2017.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

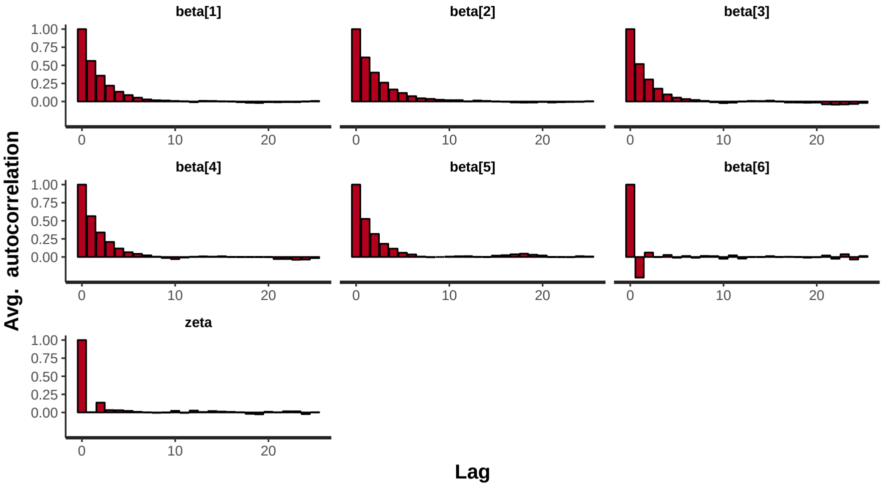
10.5.2 Stan

```
stan_trace(fit)
```



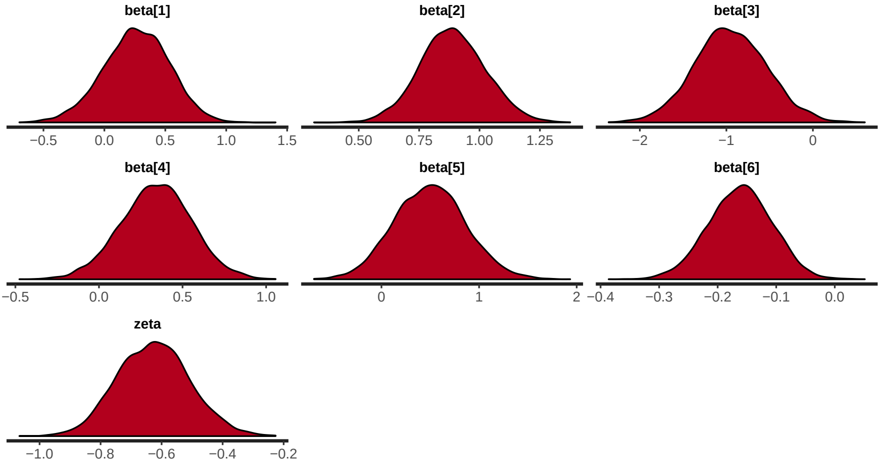
10.5.2 Stan

```
stan_ac(fit)
```



10.5.2 Stan

```
stan_dens(fit)
```



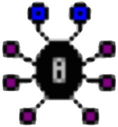
10.5.2 Stan

- Use the extract function to extract samples from the fitted model for further analysis. The output is a list of components with names corresponding to the saved parameters. Each component contain the samples for a specific variable.

```
> draws <- extract(fit)
> names(draws)
[1] "beta" "zeta" "lp__"
> summary(draws)
     Length Class  Mode   
beta 30000  -none- numeric
zeta  5000  -none- numeric
lp__  5000  -none- numeric
> dim(draws$beta)
[1] 5000    6
> dim(draws$zeta)
[1] 5000
> quantile(draws$zeta)
        0%        25%        50%        75%       100% 
-1.0662759 -0.7092433 -0.6267178 -0.5468350 -0.2263093 
```

10.5.3 BUGS

- BUGS is an open-source software for performing Bayesian inference using Gibbs Sampling (http://www.openbugs.net/w/FrontPage).



- There are two versions of BUGS, WinBUGS (not further developed) and OpenBUGS (more flexible and extensible).
- Further details: Lunn, D., Spiegelhalter, D., Thomas, A. and Best, N. (2009) The BUGS project: Evolution, critique and future directions (with discussion), Statistics in Medicine 28: 3049–3082.
- Download OpenBUGS from www.openbugs.net/w/Downloads.
- OpenBUGS can be run from within R by using the R package R2OpenBUGS. The output can be then analyzed in R using the Coda package.

10.5.3 BUGS

- The user is only required to specify a statistical model by stating the relationships between related variables. OpenBUGS includes an "expert system" which automatically determines an appropriate MCMC scheme for analyzing the specified model.
- The user is able to control certain aspects of the execution of the MCMC scheme, such as initialization values, length of burn-in, number of iterations and number of chains.
- The R package `dclone` enables multiple chains to be run in parallel on a multi-core processor. Thus computation time can be reduced by up to a factor equal to the number of chains.

10.5.3 BUGS

- The model for OpenBUGS is written in a file called `model_epilepsy.R`

```
model{
for (i in 1:n) {
    b[i] ~ dnorm(0, P)
    for (j in startindex[i]:endindex[i]){
        y[j] ~ dpois(m[j])
        log(m[j]) <- inprod(X[j,], beta[]) + Z[j]*b[i]
    }
}

for (l in 1:k){
    beta[l] ~ dnorm(0, 0.01)
}

P <- exp(-2*zeta)
zeta ~ dnorm(0, 0.01)
}
```

10.5.3 BUGS

- The model can be specified using the in a text-based language, headed by the model statement.
- It is recommended that the first step is to construct a directed graphical model (all quantities are represented as nodes and arrows run to nodes from their direct influences (parents)). Nodes are of three types:
  1. Constants.
  2. Stochastic nodes are variables that are given a distribution, e.g. `x ~ dnorm(mu, tau)`
  3. Deterministic nodes are logical functions of other nodes.
- Distributions that can be used in OpenBUGS are described in Appendix I Distributions. The parameters of a distribution must be explicit nodes (may not be expressions).

10.5.3 BUGS

- The function bugs is used to fit the model using OpenBUGS.
- When using OpenBUGS, it is important to specify initial values for (at least some of) the parameters.

```
library(R2OpenBUGS)

data <- list(n=n, k=k, y=y, X=X, Z=Z,
                  startindex=startindex, endindex=endindex)

inits <- function(){list(beta=rnorm(6), zeta=runif(1))}

out <- bugs(data=data, inits=inits, model.file="model_epilepsy.R",
            n.chain=2, n.iter=5000, n.burnin=2500, n.thin=1,
            parameters.to.save=pars, debug=TRUE, codaPkg=TRUE)

out.coda <- read.bugs(out)
```

10.5.3 BUGS

- We can use the dclone package to run multiple MCMC chains in parallel. Do note that it is necessary to install JAGS in order for this parallel processing to work.

```
require(dclone)

cl <- makePSOCKcluster(n.chains)

out <- bugs.parfit(cl, data=data, params=pars,
            model="model_epilepsy.R", inits=inits, n.chains=n.chains,
            seed=1:n.chains, program="openbugs", DIC=FALSE,
            n.iter=5000, n.thin=1, debug=TRUE)

stopCluster(cl)
```

10.5.3 BUGS

- The `read.bugs` function reads MCMC output produced by OpenBUGS in the coda format and returns an object of class mcmc.list for further output analysis using the coda package.

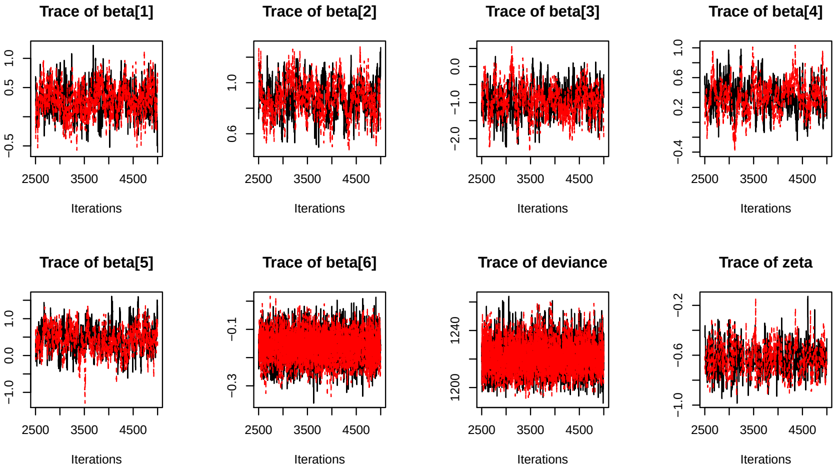
```
> out.coda <- read.bugs(out)
> (out.summary <- summary(out.coda))
Iterations = 2501:5000
Thinning interval = 1
Number of chains = 2
Sample size per chain = 2500
1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

              Mean       SD  Naive SE Time-series SE
beta[1]     0.2808  0.27883 0.0039432      0.0171261
beta[2]     0.8767  0.14297 0.0020219      0.0110059
beta[3]    -0.9345  0.41153 0.0058199      0.0264931
beta[4]     0.3419  0.20901 0.0029558      0.0197259
beta[5]     0.4819  0.38707 0.0054740      0.0248724
beta[6]    -0.1606  0.05433 0.0007684      0.0009048
deviance 1221.0902 11.08492 0.1567644      0.1863108
zeta       -0.6352  0.11941 0.0016887      0.0080495
```

10.5.3 BUGS

- Trace plot of MCMC output from OpenBUGS.

```
traceplot(out.coda)
```


Trace of beta[1]
Iterations
Trace of beta[2]
Iterations
Trace of beta[3]
Iterations
Trace of beta[4]
Iterations
Trace of beta[5]
Iterations
Trace of beta[6]
Iterations
Trace of deviance
Iterations
Trace of zeta
Iterations