# Research Group Presentation

# Overview of Bayesian analysis

# Motivating example: Likelihood

- Suppose $X_1, \ldots, X_n$ is an i.i.d. sample with $X_i \sim \mathrm{Exp}(\lambda)$. We would like to estimate $\lambda$.

# Motivating example: Likelihood

- Suppose $X_1, \dots, X_n$ is an i.i.d. sample with $X_i \sim \mathrm{Exp}(\lambda)$. We would like to estimate $\lambda$.
- Likelihood approach:

$$\hat{\lambda}_{\mathrm{MLE}} = \frac{n}{\sum_i x_i}$$

$$I(\lambda) = \lambda^{-2}$$

# Motivating example: Likelihood

- Suppose $X_1, \ldots, X_n$ is an i.i.d. sample with $X_i \sim \mathrm{Exp}(\lambda)$. We would like to estimate $\lambda$.
- Likelihood approach:

$$\hat{\lambda}_{\mathrm{MLE}} = \frac{n}{\sum_i x_i}$$

$$I(\lambda) = \lambda^{-2}$$

- If $z.025 = \mathrm{qnorm}(.975)$, then approximately and for large $n$, the interval

$$\left(\hat{\lambda}_{\mathrm{MLE}} - \frac{z_{.025}\lambda_{\mathrm{MLE}}}{\sqrt{n}}, \hat{\lambda}_{\mathrm{MLE}} - \frac{z_{.025}\lambda_{\mathrm{MLE}}}{\sqrt{n}}\right)$$

has $95\%$ coverage, meaning if we repeat the experiment 100 times, the interval will contain the true value of $\lambda$ in roughly 95 of them.

# Motivating example: Bayesian

- Think of $\lambda$ as a quantity with its own distribution

# Motivating example: Bayesian

- Think of $\lambda$ as a quantity with its own distribution
- Prior distribution: $p(\lambda) = \mathrm{Gamma}(\lambda, \alpha, \beta)$

# Motivating example: Bayesian

- Think of $\lambda$ as a quantity with its own distribution
- Prior distribution: $p(\lambda) = \mathrm{Gamma}(\lambda, \alpha, \beta)$
- Bayes' theorem:

$$\pi(\lambda, \mathbf{x}) \propto L(\lambda)p(\lambda)$$

$$= \mathrm{Gamma}(\lambda, \alpha + n, \beta + n\bar{x})$$

# Motivating example: Bayesian

- Think of $\lambda$ as a quantity with its own distribution
- Prior distribution: $p(\lambda) = \mathrm{Gamma}(\lambda, \alpha, \beta)$
- Bayes' theorem:

$$\pi(\lambda, \mathbf{x}) \propto L(\lambda)p(\lambda)$$

$$= \mathrm{Gamma}(\lambda, \alpha + n, \beta + n\bar{x})$$

- Based on this we can construct an interval $I$ such that we can literally say "$\lambda$ has a $95\%$ chance of being in $I$"

# Quantile regression

# General Framework

- OLS regression: $E[Y|\mathbf{X}] = \mathbf{X}^T\boldsymbol{\beta}$

# General Framework

- OLS regression: $E[Y|\mathbf{X}] = \mathbf{X}^T\boldsymbol{\beta}$
- Quantile regression: $q_p(Y|\mathbf{X}) = \mathbf{X}^T\boldsymbol{\beta}(p)$

# General Framework

- OLS regression: $E[Y|\mathbf{X}] = \mathbf{X}^T\boldsymbol{\beta}$
- Quantile regression: $q_p(Y|\mathbf{X}) = \mathbf{X}^T\boldsymbol{\beta}(p)$
- Frequentist approach:

$$\widehat{\boldsymbol{\beta}(p)} = \text{argmin}_{\boldsymbol{\beta}} \sum_i \rho_p(y_i - \mathbf{x}_i^T\boldsymbol{\beta})$$

where $\rho_p(u) = u(p - \mathbf{1}_{\{u<0\}})$

# Bayesian quantile regression with AL

- It can be shown that the previous minimization problem is equivalent to maximizing the likelihood associated to the distribution

$$f_p^{\mathrm{AL}}(u) = p(1-p)\exp(-\rho_p(u))$$

# Bayesian quantile regression with AL

- It can be shown that the previous minimization problem is equivalent to maximizing the likelihood associated to the distribution

$$f_p^{\mathrm{AL}}(u) = p(1-p)\exp(-\rho_p(u))$$

- More generally, we can add a location and a scale parameter:

$$f_p^{\mathrm{AL}}(u|\mu,\sigma) = p(1-p)\exp(-\rho_p((u-\mu)/\sigma))$$

  this distribution is called *asymmetric Laplace* [Yu and Moyeed, 2001]

# Bayesian quantile regression with AL

- It can be shown that the previous minimization problem is equivalent to maximizing the likelihood associated to the distribution

$$f_p^{\mathrm{AL}}(u) = p(1-p)\exp(-\rho_p(u))$$

- More generally, we can add a location and a scale parameter:

$$f_p^{\mathrm{AL}}(u|\mu,\sigma) = p(1-p)\exp(-\rho_p((u-\mu)/\sigma))$$

  this distribution is called *asymmetric Laplace* [Yu and Moyeed, 2001]
- Prior: $p(\boldsymbol{\beta}) \propto 1$ (improper)

# Bayesian quantile regression with AL

- It can be shown that the previous minimization problem is equivalent to maximizing the likelihood associated to the distribution

$$f_p^{\mathrm{AL}}(u) = p(1-p)\exp(-\rho_p(u))$$

- More generally, we can add a location and a scale parameter:

$$f_p^{\mathrm{AL}}(u|\mu,\sigma) = p(1-p)\exp(-\rho_p((u-\mu)/\sigma))$$

  this distribution is called *asymmetric Laplace* [Yu and Moyeed, 2001]
- Prior: $p(\boldsymbol{\beta}) \propto 1$ (improper)
- Theorem: This leads to a proper posterior distribution

# Flexible quantile regression with GAL

- Problems: AL is very rigid

# Flexible quantile regression with GAL

- Problems: AL is very rigid
  - Skewness is determined by $p$, in particular it is symmetric at $p = .5$

# Flexible quantile regression with GAL

- Problems: AL is very rigid
  - Skewness is determined by $p$, in particular it is symmetric at $p = .5$
  - Mode is always at 0

# Flexible quantile regression with GAL

- Problems: AL is very rigid
  - Skewness is determined by $p$, in particular it is symmetric at $p = .5$
  - Mode is always at 0
- Solution [Yan and Kottas, 2017]:

# Flexible quantile regression with GAL

- Problems: AL is very rigid
  - Skewness is determined by $p$, in particular it is symmetric at $p = .5$
  - Mode is always at 0
- Solution [Yan and Kottas, 2017]:
  - We can write

$$f_p^{\mathrm{AL}}(y|\mu,\sigma) = \int_{\mathbb{R}} N(y|\mu + \sigma A(p)z, \sigma^2 B(p)z) Exp(z|1) dz$$

# Flexible quantile regression with GAL

- Problems: AL is very rigid
  - Skewness is determined by $p$, in particular it is symmetric at $p = .5$
  - Mode is always at 0
- Solution [Yan and Kottas, 2017]:
  - We can write

$$f_p^{\mathrm{AL}}(y|\mu,\sigma) = \int_{\mathbb{R}} N(y|\mu + \sigma A(p)z, \sigma^2 B(p)z) Exp(z|1) dz$$

  - So, let us add a parameter

$$f_p^{\mathrm{GAL}}(y|\mu,\sigma)$$

$$= \int_{\mathbb{R}} N(y|\mu+\sigma\alpha s+\sigma A(p)z, \sigma^2 B(p)z) Exp(z|1) N^+(s|0,1) dz ds$$

# Parameter Estimation

# Overview of Stan

- Likelyhood above is very complicated, so sampling from the posterior distribution of parameters becomes an issue

# Overview of Stan

- Likelyhood above is very complicated, so sampling from the posterior distribution of parameters becomes an issue
- Usually, we deal with this using Markov Chain Monte Carlo (MCMC) techniques

# Overview of Stan

- Likelyhood above is very complicated, so sampling from the posterior distribution of parameters becomes an issue
- Usually, we deal with this using Markov Chain Monte Carlo (MCMC) techniques
- Coding MCMCs by hand is cumbersome. It can also lead to convergence issues

# Overview of Stan

- Likelyhood above is very complicated, so sampling from the posterior distribution of parameters becomes an issue
- Usually, we deal with this using Markov Chain Monte Carlo (MCMC) techniques
- Coding MCMCs by hand is cumbersome. It can also lead to convergence issues
- Software packages like BUGS and JAGS can automatically code MCMCs based on model structure

# Overview of Stan

- Likelyhood above is very complicated, so sampling from the posterior distribution of parameters becomes an issue
- Usually, we deal with this using Markov Chain Monte Carlo (MCMC) techniques
- Coding MCMCs by hand is cumbersome. It can also lead to convergence issues
- Software packages like BUGS and JAGS can automatically code MCMCs based on model structure
- Stan [Stan development team, 2012] improves on those packages by implementing HMC and NUTS

# Overview of BRMS

- Stan is low level: it's a programming language on it's own, built on top of C++

# Overview of BRMS

- Stan is low level: it's a programming language on it's own, built on top of C++
- BRMS [Buckner, 2017] is a high level interface for Stan on R

# Overview of BRMS

- Stan is low level: it's a programming language on it's own, built on top of C++
- BRMS [Buckner, 2017] is a high level interface for Stan on R
  - It uses the same formula syntax as `lme4`

# Overview of BRMS

- Stan is low level: it's a programming language on it's own, built on top of C++
- BRMS [Buckner, 2017] is a high level interface for Stan on R
  - It uses the same formula syntax as `lme4`
  - Under the hood, it automatically generates and compiles Stan code, which we can retrieve