Lecture 1

~~Spatio-temporal data~~ & Linear Models

Colin Rundel

1/16/2018

(Bayesian) Linear Models

Pretty much everything we a going to see in this course will fall under the umbrella of either linear or generalized linear models.

In previous classes most of your time has likely been spent with the simple iid case,

$$Y_i = \beta_0 + \beta_1\, x_{i1} + \cdots + \beta_p\, x_{ip} + \epsilon_i$$
$$\epsilon_i \sim N(0, \sigma^2)$$

these models can also be expressed simply as,

$$Y_i \overset{iid}{\sim} N(\beta_0 + \beta_1\, x_{i1} + \cdots + \beta_p\, x_{ip},\ \sigma^2)$$

We can also express using matrix notation as follows,

$$\underset{n\times 1}{\mathbf{Y}} = \underset{n\times p}{\mathbf{X}}\,\underset{p\times 1}{\boldsymbol{\beta}} + \underset{n\times 1}{\boldsymbol{\epsilon}}$$

$$\underset{n\times 1}{\boldsymbol{\epsilon}} \sim N(\underset{n\times 1}{\mathbf{0}},\ \sigma^2 \underset{n\times n}{\mathbb{1}_n})$$

or alternative as,

$$\underset{n\times 1}{\mathbf{Y}} \sim N\left(\underset{n\times p}{\mathbf{X}}\,\underset{p\times 1}{\boldsymbol{\beta}},\ \sigma^2 \underset{n\times n}{\mathbb{1}_n}\right)$$

For an n-dimension multivate normal distribution with covariance $\boldsymbol{\Sigma}$ (positive semidefinite) can be written as

$$\underset{n\times 1}{\mathbf{Y}} \sim N(\underset{n\times 1}{\boldsymbol{\mu}}, \underset{n\times n}{\boldsymbol{\Sigma}}) \text{ where } \{\boldsymbol{\Sigma}\}_{ij} = \rho_{ij}\sigma_i\sigma_j$$

$$\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix}, \begin{pmatrix} \rho_{11}\sigma_1\sigma_1 & \rho_{12}\sigma_1\sigma_2 & \cdots & \rho_{1n}\sigma_1\sigma_n \\ \rho_{21}\sigma_2\sigma_1 & \rho_{22}\sigma_2\sigma_2 & \cdots & \rho_{2n}\sigma_2\sigma_n \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1}\sigma_n\sigma_1 & \rho_{n2}\sigma_n\sigma_2 & \cdots & \rho_{nn}\sigma_n\sigma_n \end{pmatrix}\right)$$

n $\qquad$ $\frac{n(n-1)}{2}$ $\qquad$ n

μ $\qquad$ ρ $\qquad$ σ

Multivariate Normal Distribution - Density

For the n dimensional multivate normal given on the last slide, its density is given by

$$(2\pi)^{-n/2} \det(\mathbf{\Sigma})^{-1/2} \exp\left(-\frac{1}{2}\underset{1\times n}{(\mathbf{Y}-\boldsymbol{\mu})'}\underset{n\times n}{\mathbf{\Sigma}^{-1}}\underset{n\times 1}{(\mathbf{Y}-\boldsymbol{\mu})}\right)$$

and its log density is given by

$$-\frac{n}{2}\log 2\pi - \frac{1}{2}\log\det(\mathbf{\Sigma}) - \frac{1}{2}\underset{1\times n}{(\mathbf{Y}-\boldsymbol{\mu})'}\underset{n\times n}{\mathbf{\Sigma}^{-1}}\underset{n\times 1}{(\mathbf{Y}-\boldsymbol{\mu})}$$

$$\mathcal{L}(\beta,\sigma^2 \mid \underline{X},\underline{Y}) = (2\pi)^{-n/2}\det(\underline{\Sigma})^{-1/2}\exp\left(-\frac{1}{2}(\underline{Y}-\underline{X}\beta)'\underline{\Sigma}^{-1}(\underline{Y}-\underline{X}\beta)\right)$$

$$\ell(\beta,\sigma^2) \propto -\frac{1}{2}\log\det(\underline{\Sigma}) - \frac{1}{2}(\underline{Y}-\underline{X}\beta)'\underline{\Sigma}^{-1}(\underline{Y}-\underline{X}\beta)$$

$$\det\underline{\Sigma} = \prod_{i=1}^{n}\sigma^2 = (\sigma^2)^n \qquad \underline{\Sigma}^{-1} = 1/\sigma^2\,\mathbb{1}$$

$$\ell(\beta,\sigma^2) \propto -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}(\underline{Y}-\underline{X}\beta)'(\underline{Y}-\underline{X}\beta)$$

$$\frac{\partial\,\ell(\beta,\sigma^2)}{\partial\beta} \propto -\frac{1}{2\sigma^2}\left[\left[(-X')(Y-X\beta)\right]' + (Y-X\beta)'(-X)\right]$$

Maximum Likelihood - β (iid)

$$\frac{\partial \ell(\beta, \sigma^2)}{\partial \beta} \propto \left[(y - X\beta)'(-X) + (y - X\beta)'(-X) \right]$$

$$\propto 2\left[y'X - \beta'X'X \right] = 0$$

$$\beta' X' X = y' X$$

$$\beta' = y'X(X'X)^{-1}$$

$$\beta = (X'X)X'y$$

$$\ell(\beta, \sigma^2) \propto -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} (\underline{y} - \underline{X}\underline{\beta})' (\underline{y} - \underline{X}\underline{\beta})$$

$$\frac{\partial \ell(\beta, \sigma^2)}{\partial \sigma^2} \propto -\frac{n}{2} \frac{1}{\sigma^2} + \frac{1}{2(\sigma^2)^2} (\underline{y} - \underline{X}\underline{\beta})' (\underline{y} - \underline{X}\underline{\beta})$$

$$\propto \frac{1}{2\sigma^2} \left(-n + \frac{1}{\sigma^2} (\underline{y} - \underline{X}\underline{\beta})' (\underline{y} - \underline{X}\underline{\beta}) \right) = 0$$

Maximum Likelihood - σ^2 (iid)

$$\left(-n + \frac{1}{\sigma^2}(\underline{y}-\underline{X}\underline{\beta})'(\underline{y}-\underline{X}\underline{\beta})\right) = 0$$

$$(y - X\beta)'(y - X\beta) = n\sigma^2$$

$$\sigma^2 = \frac{1}{n}(\underset{1\times n}{\underline{y}-\underline{X}\underline{\beta}})'(\underset{n\times 1}{\underline{y}-\underline{X}\underline{\beta}})$$

$$= \frac{1}{n}\left(\sum_{i=1}^{n}(y_i - \underline{x}_{i\cdot}\underline{\beta})^2\right)$$

$$= \frac{1}{n}\sum(y_i - x_i\beta)^2$$

Likelihood:

$$\mathbf{Y} \mid \boldsymbol{\beta}, \sigma^2 \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbb{1}_n)$$

Likelihood:

$$\mathbf{Y} \mid \boldsymbol{\beta},\, \sigma^2 \sim N(\mathbf{X}\boldsymbol{\beta},\, \sigma^2\, \mathbb{1}_n)$$

Priors:

$$\beta_i \sim N(0, \sigma_\beta^2) \text{ or } \boldsymbol{\beta} \sim N(\mathbf{0}, \sigma_\beta^2\, \mathbb{1}_p)$$

$$\sigma^2 \sim \text{Inv-Gamma}(a,\, b)$$

$$1/\sigma^2 \sim \text{gamma}(a, b)$$

Deriving the posterior

$$
\begin{aligned}
[\boldsymbol{\beta}, \sigma^2 \,|\, \mathbf{Y}, \mathbf{X}] &= \frac{[\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2]}{[\mathbf{Y} \,|\, \mathbf{X}]}[\boldsymbol{\beta}, \sigma^2] \\
&\propto [\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta},\ \sigma^2] \\
&\propto [\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} \,|\, \sigma^2][\sigma^2]
\end{aligned}
$$

Deriving the posterior

$$\begin{aligned}[\boldsymbol{\beta}, \sigma^2 \mid \mathbf{Y}, \mathbf{X}] &= \frac{[\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2]}{[\mathbf{Y} \mid \mathbf{X}]}[\boldsymbol{\beta}, \sigma^2] \\ &\propto [\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta},\ \sigma^2] \\ &\propto [\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} \mid \sigma^2][\sigma^2]\end{aligned}$$

where,

$$f(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})\right)$$

Deriving the posterior

$$
\begin{aligned}
[\boldsymbol{\beta}, \sigma^2 \,|\, \mathbf{Y}, \mathbf{X}] &= \frac{[\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2]}{[\mathbf{Y} \,|\, \mathbf{X}]} [\boldsymbol{\beta}, \sigma^2] \\
&\propto [\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta},\ \sigma^2] \\
&\propto [\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} \,|\, \sigma^2][\sigma^2]
\end{aligned}
$$

where,

$$
f(\mathbf{Y} \,|\, \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})\right)
$$

$$
f(\boldsymbol{\beta} \,|\, \sigma^2_\beta) = (2\pi\sigma^2_\beta)^{-p/2} \exp\left(-\frac{1}{2\sigma^2_\beta}\boldsymbol{\beta}'\boldsymbol{\beta}\right)
$$

Deriving the posterior

$$
\begin{aligned}
[\boldsymbol{\beta}, \sigma^2 \mid \mathbf{Y}, \mathbf{X}] &= \frac{[\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2]}{[\mathbf{Y} \mid \mathbf{X}]}[\boldsymbol{\beta}, \sigma^2] \\
&\propto [\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta},\ \sigma^2] \\
&\propto [\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} \mid \sigma^2][\sigma^2]
\end{aligned}
$$

where,

$$
f(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\beta}, \sigma^2) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})\right)
$$

$$
f(\boldsymbol{\beta} \mid \sigma^2_\beta) = (2\pi\sigma^2_\beta)^{-p/2} \exp\left(-\frac{1}{2\sigma^2_\beta}\boldsymbol{\beta}'\boldsymbol{\beta}\right)
$$

$$
f(\sigma^2 \mid a,\ b) = \frac{b^a}{\Gamma(a)}(\sigma^2)^{-a-1} \exp\left(-\frac{b}{\sigma^2}\right)
$$

Deriving the Gibbs sampler (σ^2 step)

$$[\boldsymbol{\beta}, \sigma^2 \mid \mathbf{Y}, \mathbf{X}] \propto (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})\right)$$

$$(2\pi\sigma_\beta^2)^{-p/2} \exp\left(-\frac{1}{2\sigma_\beta^2}\boldsymbol{\beta}'\boldsymbol{\beta}\right)$$

$$\frac{b^a}{\Gamma(a)}(\sigma^2)^{-a-1} \exp\left(-\frac{b}{\sigma^2}\right)$$

$$\propto (\sigma^2)^{-a-\frac{n}{2}-1} \exp\left(-\frac{1}{\sigma^2}\left(b + \frac{(Y-X\beta)'(Y-X\beta)}{2}\right)\right)$$

$$\sigma^2 \mid \ldots \sim \text{InvGamma}\left(a + \frac{n}{2},\ b + \frac{(Y-X\beta)'(Y-X\beta)}{2}\right)$$

Deriving the Gibbs sampler (β step)

$$[\boldsymbol{\beta}, \sigma^2 \mid \mathbf{Y}, \mathbf{X}] \propto (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})\right)$$

$$(2\pi\sigma_\beta^2)^{-p/2} \exp\left(-\frac{1}{2\sigma_\beta^2}\boldsymbol{\beta}'\boldsymbol{\beta}\right)$$

$$\frac{b^a}{\Gamma(a)}(\sigma^2)^{-a-1} \exp\left(-\frac{b}{\sigma^2}\right)$$

$S = \frac{\sigma^2}{\sigma_\beta^2}$

$\sigma_\beta^2 = \sigma^2 / S$

$$\beta \mid \ldots \propto \exp\left(-\frac{1}{2}\left[\frac{1}{\sigma^2}(y' - \beta' x')(y - x\beta) + \frac{S}{\sigma^2}\beta'\beta\right]\right)$$

$$\propto \exp\left(-\frac{1}{2\sigma^2}\left[y'y - \beta' x' y + y' x\beta + \beta' x' x\beta) + S\beta'\beta\right]\right)$$

~~Backwards~~

$$\beta|\cdots \sim N(\mu_p, \Sigma_p)$$

$$\beta|\cdots \propto \exp\left(-\tfrac{1}{2}(\beta-\mu_p)'\Sigma_p^{-1}(\beta-\mu_p)\right)$$

$$\propto \exp\left(-\tfrac{1}{2}\left(\beta'\Sigma_p^{-1}\beta - \beta'\Sigma_p^{-1}\mu_p - \mu_p'\Sigma_p\beta\right)\right)$$

$$\beta'X'X\beta + \beta' s \mathbb{1} \beta = \beta'(X'X + s\mathbb{1})\beta = \Sigma_p^{-1}$$

$$\Sigma_p = \left(X'X + \frac{\sigma^2}{\sigma_\beta^2}\mathbb{1}\right)^{-1}$$

$$\beta' X y = \beta' \Sigma_p^{-1} \mu_p$$

$$\mu_p = \Sigma_p X'y = \left(X'X + \frac{\sigma^2}{\sigma_\beta}\mathbb{1}\right)^{-1} X'y$$

A Quick Example

Some Fake Data

Lets generate some simulated data where the underlying model is known and see how various regression procedures function.

$$\beta_0 = 0.7, \quad \beta_1 = 1.5, \quad \beta_2 = -2.2, \quad \beta_3 = 0.1$$

$$n = 100, \quad \epsilon_i \sim N(0, 1)$$

Generating the data

```
set.seed(01162018)
n = 100
beta = c(0.7,1.5,-2.2,0.1)
eps = rnorm(n)

d = data_frame(
  X1 = rt(n,df=5),
  X2 = rt(n,df=5),
  X3 = rt(n,df=5)
) %>%
  mutate(Y = beta[1] + beta[2]*X1 + beta[3]*X2 + beta[4]*X3 + eps)
```

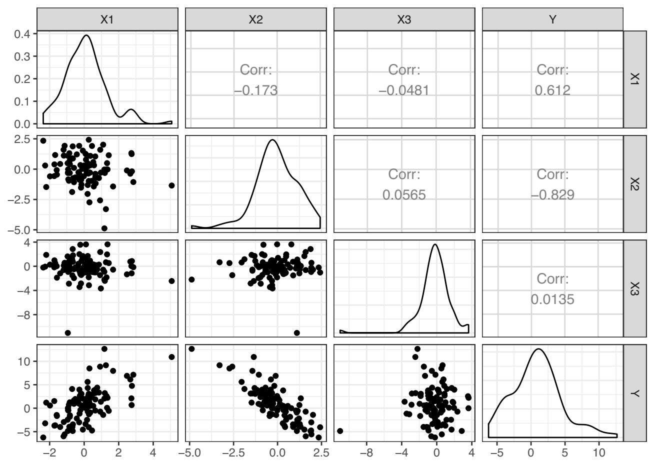
Model Matrix

```r
X = model.matrix(~X1+X2+X3, d)
tbl_df(X)
```

```
## # A tibble: 100 x 4
##    `(Intercept)`      X1     X2     X3
##            <dbl>   <dbl>  <dbl>  <dbl>
##  1             1  1.26   -3.30   0.664
##  2             1  1.17   -4.88  -2.20
##  3             1 -0.590  -1.08  -1.95
##  4             1  0.358   0.319 -0.582
##  5             1  1.20   -0.314 -2.41
##  6             1  0.856  -0.733  0.274
##  7             1  0.649  -0.385  0.112
##  8             1 -1.58   -0.449 -1.89
##  9             1 -0.424   1.22  -0.193
## 10             1  0.0808  1.27  -0.488
## # ... with 90 more rows
```

Pairs plot

```
GGally::ggpairs(d, progress = FALSE)
```



Least squares fit

Let $\hat{\mathbf{Y}}$ be our estimate for $\mathbf{Y}$ based on our estimate of $\boldsymbol{\beta}$,

$$\hat{\mathbf{Y}} = \hat{\beta}_0 + \hat{\beta}_1\,\mathbf{X}_1 + \hat{\beta}_2\,\mathbf{X}_2 + \hat{\beta}_3\,\mathbf{X}_3 = \mathbf{X}\,\hat{\boldsymbol{\beta}}$$

Let $\hat{\mathbf{Y}}$ be our estimate for $\mathbf{Y}$ based on our estimate of $\boldsymbol{\beta}$,

$$\hat{\mathbf{Y}} = \hat{\beta}_0 + \hat{\beta}_1 \, \mathbf{X}_1 + \hat{\beta}_2 \, \mathbf{X}_2 + \hat{\beta}_3 \, \mathbf{X}_3 = \mathbf{X} \, \hat{\boldsymbol{\beta}}$$

The least squares estimate, $\hat{\boldsymbol{\beta}}_{ls}$, is given by

$$\underset{\boldsymbol{\beta}}{\arg\min} \sum_{i=1}^{n} \left(Y_i - \mathbf{X}_{i\cdot} \boldsymbol{\beta}\right)^2$$

Least squares fit

Let $\hat{\mathbf{Y}}$ be our estimate for $\mathbf{Y}$ based on our estimate of $\boldsymbol{\beta}$,

$$\hat{\mathbf{Y}} = \hat{\beta}_0 + \hat{\beta}_1\,\mathbf{X}_1 + \hat{\beta}_2\,\mathbf{X}_2 + \hat{\beta}_3\,\mathbf{X}_3 = \mathbf{X}\,\hat{\boldsymbol{\beta}}$$

The least squares estimate, $\hat{\boldsymbol{\beta}}_{ls}$, is given by

$$\underset{\boldsymbol{\beta}}{\arg\min} \sum_{i=1}^{n} \left(Y_i - \mathbf{X}_{i\cdot}\boldsymbol{\beta}\right)^2$$

Previously we derived,

$$\hat{\boldsymbol{\beta}}_{ls} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\,\mathbf{Y}$$

Frequentist Fit

```
l = lm(Y ~ X1 + X2 + X3, data=d)
l$coefficients
```

```
## (Intercept)          X1          X2          X3
##   0.6566561   1.4657537  -2.2807109   0.1629704
```

```
(beta_hat = solve(t(X) %*% X, t(X)) %*% d$Y)
```

```
##                   [,1]
## (Intercept)  0.6566561
## X1           1.4657537
## X2          -2.2807109
## X3           0.1629704
```

Bayesian model specification (JAGS)

```
model =
"model{
  # Likelihood
  for(i in 1:length(Y)){
    Y[i] ~ dnorm(mu[i], tau)
    mu[i] = beta[1] + beta[2]*X1[i] + beta[3]*X2[i] + beta[4]*X3[i]
  }

  # Prior for beta
  for(j in 1:4){
    beta[j] ~ dnorm(0,1/100)
  }

  # Prior for sigma / tau2
  tau ~ dgamma(1, 1)
  sigma2 = 1/tau
}"
```

Compiling

```r
m = rjags::jags.model(
  textConnection(model),
  data = d, n.chains = 2
)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 100
##    Unobserved stochastic nodes: 5
##    Total graph size: 810
##
## Initializing model
```

Sampling

```
# Burnin
update(m, n.iter=1000, progress.bar="none")

# Draw samples
samp = rjags::coda.samples(
  m, variable.names=c("beta","sigma2"),
  n.iter=5000, progress.bar="none"
)
```

Results

```
str(samp)
```

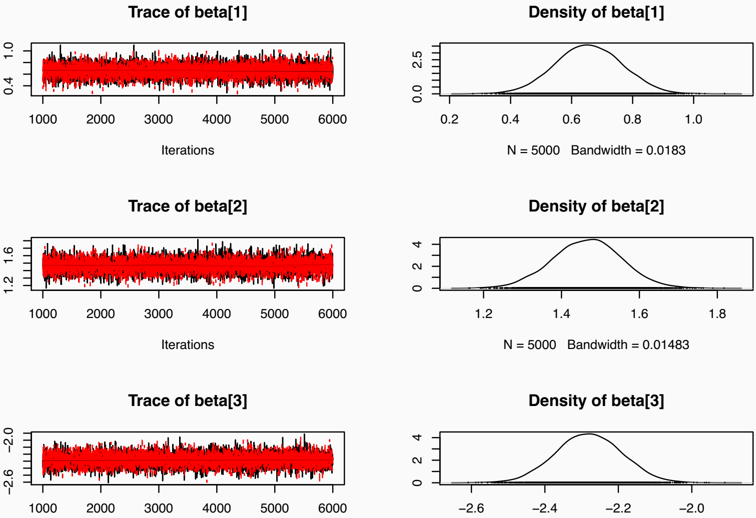
```
## List of 2
##  $ : 'mcmc' num [1:5000, 1:5] 0.477 0.533 0.513 0.608 0.47 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:5] "beta[1]" "beta[2]" "beta[3]" "beta[4]" ...
##   ..- attr(*, "mcpar")= num [1:3] 1001 6000 1
##  $ : 'mcmc' num [1:5000, 1:5] 0.729 0.351 0.576 0.687 0.622 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:5] "beta[1]" "beta[2]" "beta[3]" "beta[4]" ...
##   ..- attr(*, "mcpar")= num [1:3] 1001 6000 1
##  - attr(*, "class")= chr "mcmc.list"
```

```
head(samp[[1]])
```

```
## Markov Chain Monte Carlo (MCMC) output:
## Start = 1001
## End = 1007
## Thinning interval = 1
##        beta[1]  beta[2]   beta[3]    beta[4]    sigma2
## [1,] 0.4772517 1.419778 -2.199144 0.28649330 1.0809728
## [2,] 0.5328665 1.521434 -2.391340 0.10170084 1.2145136
## [3,] 0.5127307 1.569942 -2.288688 0.09279966 1.2330530
## [4,] 0.6082786 1.457989 -2.301124 0.11507748 0.7760024
## [5,] 0.4696911 1.497949 -2.427537 0.15902657 1.1581875
## [6,] 0.7505117 1.397314 -2.231748 0.04573775 0.9684167
## [7,] 0.6361385 1.625917 -2.208012 0.21898668 1.3867957
```

CODA & mcmc objects - plotting

```
plot(samp)
```



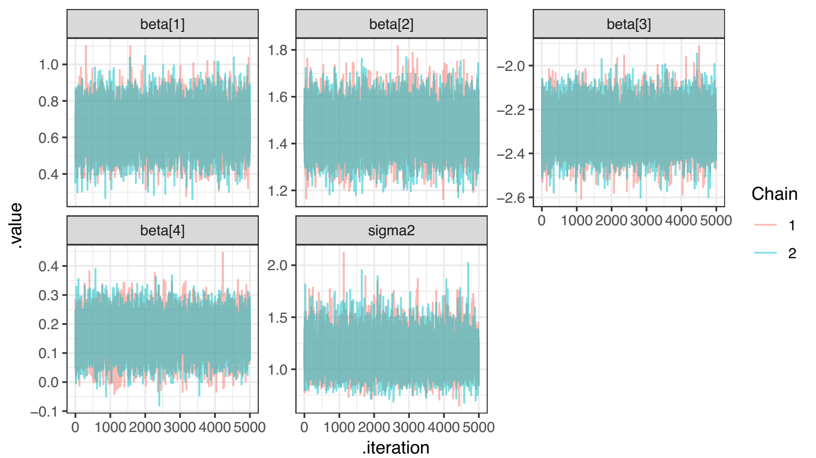
```
df = samp %>%
  tidybayes::gather_draws(beta[i], sigma2) %>%
  mutate(param = ifelse(is.na(i), .variable, paste0(.variable,"[",i,"]")))

df
```

```
## # A tibble: 50,000 x 7
## # Groups:   i, .variable [5]
##    .chain .iteration .draw     i .variable .value param
##     <int>      <int> <int> <int> <chr>      <dbl> <chr>
##  1      1          1     1     1 beta       0.477 beta[1]
##  2      1          1     1     2 beta       1.42  beta[2]
##  3      1          1     1     3 beta      -2.20  beta[3]
##  4      1          1     1     4 beta       0.286 beta[4]
##  5      1          2     2     1 beta       0.533 beta[1]
##  6      1          2     2     2 beta       1.52  beta[2]
##  7      1          2     2     3 beta      -2.39  beta[3]
##  8      1          2     2     4 beta       0.102 beta[4]
##  9      1          3     3     1 beta       0.513 beta[1]
## 10      1          3     3     2 beta       1.57  beta[2]
## # ... with 49,990 more rows
```

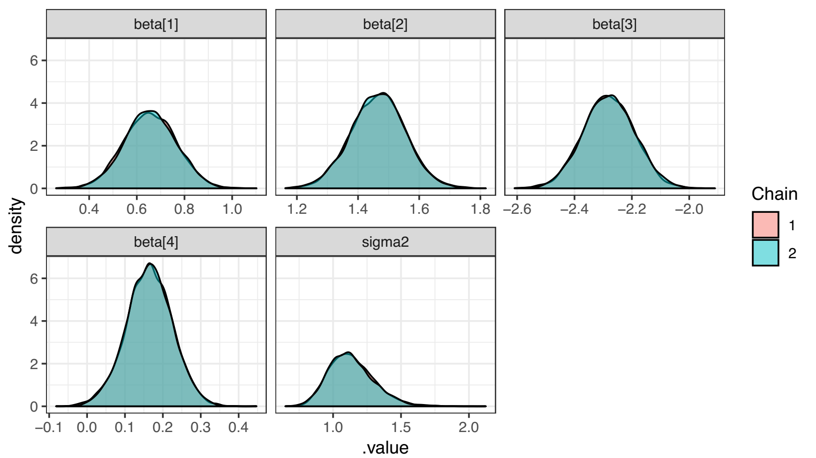
Tidy Bayes + ggplot - Traceplot

```
ggplot(df, aes(x=.iteration, y=.value, color=as.character(.chain))) +
  geom_line(alpha=0.5) +
  facet_wrap(~param, scale="free_y") +
  labs(color="Chain")
```



Tidy Bayes + ggplot - Density plot

```
ggplot(df, aes(x=.value, fill=as.character(.chain))) +
  geom_density(alpha=0.5) +
  facet_wrap(~param, scale="free_x") +
  labs(fill="Chain")
```



Comparing Approaches

```
pt_est = df %>%
  filter(.chain == 1) %>%
  group_by(param) %>%
  summarize(post_mean = mean(.value)) %>%
  ungroup() %>%
  mutate(
    truth = c(0.7, 1.5, -2.2, 0.1, 1),
    ols   = c(l$coefficients, var(l$residuals))
  ) %>%
  select(param, truth, ols, post_mean)

pt_est
```

```
## # A tibble: 5 x 4
##   param   truth    ols post_mean
##   <chr>   <dbl>  <dbl>     <dbl>
## 1 beta[1]   0.7  0.657     0.656
## 2 beta[2]   1.5  1.47      1.47
## 3 beta[3]  -2.2 -2.28     -2.28
## 4 beta[4]   0.1  0.163     0.164
## 5 sigma2    1    1.08      1.14
```

Comparing Approaches - code

```
ggplot(df, aes(x=.value, fill=as.character(.chain))) +
  geom_density(alpha=0.5) +
  facet_wrap(~param, scale="free_x") +
  geom_vline(
    data = tidyr::gather(pt_est, pt_est, value, -param),
    aes(xintercept = value, color=pt_est)
  ) +
  labs(fill="Chain")
```

Comparing Approaches - plot