Welcome!

BAYESIAN REGRESSION MODELING WITH RSTANARM

Jake Thompson
Psychometrician, ATLAS, University of Kansas

Overview

1. Introduction to Bayesian regression
2. Customizing Bayesian regression models
3. Evaluating Bayesian regression models
4. Presenting and using Bayesian regression models

A review of frequentist regression

- Frequentist regression using ordinary least squares
- The `kidiq` data

```
kidiq
```

```
# A tibble: 434 x 4
    kid_score mom_hs mom_iq mom_age
        <int>  <int>  <dbl>   <int>
  1        65      1  121.       27
  2        98      1   89.4      25
  3        85      1  115.       27
  4        83      1   99.4      25
  5       115      1   92.7      27
 # ... with 430 more rows
```

- Predict child's IQ score from the mother's IQ score

```
lm_model <- lm(kid_score ~ mom_iq, data = kidiq)
summary(lm_model)
```

```
Call:
lm(formula = kid_score ~ mom_iq, data = kidiq)
Residuals:
    Min      1Q  Median      3Q     Max
-56.753 -12.074   2.217  11.710  47.691
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 25.79978    5.91741    4.36 1.63e-05 ***
mom_iq       0.60997    0.05852   10.42  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 18.27 on 432 degrees of freedom
Multiple R-squared:  0.201,  Adjusted R-squared:  0.1991
F-statistic: 108.6 on 1 and 432 DF,  p-value: < 2.2e-16
```

Examing model coefficients

- Use the **broom** package to focus just on the coefficients

```r
library(broom)


tidy(lm_model)
```

```
         term   estimate  std.error statistic      p.value
1 (Intercept) 25.7997778 5.91741208  4.359977 1.627847e-05
2      mom_iq  0.6099746 0.05852092 10.423188 7.661950e-23
```

- Be cautious about what the p-value actually represents

Comparing Frequentist and Bayesian probabilities

- What's the probability a woman has cancer, given positive mammogram?
  - $P(+M \mid C) = 0.9$
  - $P(C) = 0.004$
  - $P(+M) = (0.9 \times 0.004) + (0.1 \times 0.996) = 0.1$
- What is $P(C \mid M+)$?
  - 0.036

Spotify data

```
songs
```

```
# A tibble: 215 x 7
   track_name    artist_name song_age valence tempo popularity duration_ms
   <chr>         <chr>          <int>   <dbl> <dbl>      <int>       <int>
 1 Crazy In Love Beyoncé         5351   70.1   99.3         72      235933
 2 Naughty Girl  Beyoncé         5351   64.3  100.0         59      208600
 3 Baby Boy      Beyoncé         5351   77.4   91.0         57      244867
 4 Hip Hop Star  Beyoncé         5351   96.8  167.          39      222533
 5 Be With You   Beyoncé         5351   75.6   74.9         42      260160
 6 Me, Myself a… Beyoncé         5351   55.5   83.6         54      301173
 7 Yes           Beyoncé         5351   56.2  112.          43      259093
 8 Signs         Beyoncé         5351   39.8   74.3         41      298533
 9 Speechless    Beyoncé         5351    9.92 113.          41      360440
# ... with 206 more rows
```

Let's practice!

BAYESIAN REGRESSION MODELING WITH RSTANARM

Bayesian Linear Regression

BAYESIAN REGRESSION MODELING WITH RSTANARM

Jake Thompson
Psychometrician, ATLAS, University of Kansas

Why use Bayesian methods?

- P-values make inferences about the probability of data, not parameter values
- Posterior distribution: combination of likelihood and prior
  - Sample the posterior distribution
  - Summarize the sample
  - Use the summary to make inferences about parameter values

The rstanarm package

- Interface to the *Stan* probabilistic programming language
- **rstanarm** provides high level access to *Stan*
- Allows for custom model definitions

```r
library(rstanarm)
stan_model <- stan_glm(kid_score ~ mom_iq, data = kidiq)
```

```
SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
 Gradient evaluation took 0.000408 seconds
 1000 transitions using 10 leapfrog steps per transition would take
 4.08 seconds.
 Adjust your expectations accordingly!

 Iteration:    1 / 2000 [  0%]  (Warmup)
 Iteration:  200 / 2000 [ 10%]  (Warmup)
 Iteration:  400 / 2000 [ 20%]  (Warmup)
 Iteration:  600 / 2000 [ 30%]  (Warmup)
 Iteration:  800 / 2000 [ 40%]  (Warmup)
 Iteration: 1000 / 2000 [ 50%]  (Warmup)
 Iteration: 1001 / 2000 [ 50%]  (Sampling)
 Iteration: 1200 / 2000 [ 60%]  (Sampling)
 Iteration: 1400 / 2000 [ 70%]  (Sampling)
 Iteration: 1600 / 2000 [ 80%]  (Sampling)
```

```
summary(stan_model)
```

```
Model Info:
 function:     stan_glm
 family:       gaussian [identity]
 formula:      kid_score ~ mom_iq
 algorithm:    sampling
 priors:       see help('prior_summary')
 sample:       4000 (posterior sample size)
 observations: 434
 predictors:   2

Estimates:
                mean    sd      2.5%    25%     50%     75%     97.5%
(Intercept)      25.7     6.0    13.8    21.6    25.7    30.0    37.0
mom_iq            0.6     0.1     0.5     0.6     0.6     0.7     0.7
sigma            18.3     0.6    17.1    17.9    18.3    18.7    19.5
mean_PPD         86.8     1.2    84.3    85.9    86.8    87.6    89.2
log-posterior -1885.4     1.2 -1888.5 -1886.0 -1885.1 -1884.5 -1884.0

Diagnostics:
              mcse Rhat n_eff
(Intercept)   0.1  1.0  4000
mom_iq        0.0  1.0  4000
sigma         0.0  1.0  3827
```

rstanarm summary: Estimates

```
Estimates:
                 mean      sd      2.5%     25%      50%      75%      97.5%
(Intercept)       25.7     6.0     13.8     21.6     25.7     30.0     37.0
mom_iq             0.6     0.1      0.5      0.6      0.6      0.7      0.7
sigma             18.3     0.6     17.1     17.9     18.3     18.7     19.5
mean_PPD          86.8     1.2     84.3     85.9     86.8     87.6     89.2
log-posterior -1885.4     1.2  -1888.5  -1886.0  -1885.1  -1884.5  -1884.0
```

- sigma: Standard deviation of errors
- mean_PPD: mean of posterior predictive samples
- log-posterior: analogous to a likelihood

rstanarm summary: Diagnostics

```
Diagnostics:
              mcse Rhat n_eff
(Intercept)   0.1  1.0  4000
mom_iq        0.0  1.0  4000
sigma         0.0  1.0  3827
mean_PPD      0.0  1.0  4000
log-posterior 0.0  1.0  1896

For each parameter, mcse is Monte Carlo standard error,
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).
```

- Rhat: a measure of within chain variance compared to across chain variance
- Values less than 1.1 indicate convergence

Let's practice!

BAYESIAN REGRESSION MODELING WITH RSTANARM

Comparing Bayesian and Frequentist Approaches

BAYESIAN REGRESSION MODELING WITH RSTANARM

Jake Thompson
Psychometrician, ATLAS, University of Kansas

The same parameters!

```
tidy(lm_model)
```

```
         term   estimate  std.error statistic      p.value
1 (Intercept) 25.7997778 5.91741208  4.359977 1.627847e-05
2      mom_iq  0.6099746 0.05852092 10.423188 7.661950e-23
```

```
tidy(stan_model)
```

```
         term   estimate  std.error
1 (Intercept) 25.7257965 6.01262625
2      mom_iq  0.6110254 0.05917996
```

Frequentist vs. Bayesian

- Frequentist: parameters are fixed, data is random
- Bayesian: parameters are random, data is fixed
- What's a p-value?
  - Probability of test statistic, given null hypothesis
- So what do Bayesians want?
  - Probability of parameter values, given the observed data

Evaluating Bayesian parameters

- Confidence interval: Probability that a range contains the true value
  - There is a 90% probability that range contains the true value
- Credible interval: Probability that the true value is within a range
  - There is a 90% probability that the true value falls within this range
- Probability of parameter values vs. probability of range boundaries

Creating credible intervals

```
posterior_interval(stan_model)
```

```
                    5%        95%
(Intercept) 16.1396617 35.6015948
mom_iq       0.5131289  0.7042666
sigma       17.2868651 19.3411104
```

```
posterior_interval(stan_model, prob = 0.95)
```

```
                  2.5%      97.5%
(Intercept) 14.5472824 37.2505664
mom_iq       0.4963677  0.7215823
sigma       17.1197930 19.5359616
```

```
posterior_interval(stan_model, prob = 0.5)
```

```
                   25%        75%
(Intercept) 21.7634032 29.6542886
mom_iq       0.5714405  0.6496865
sigma       17.8776965 18.7218373
```

Confidence vs. Credible intervals

```r
confint(lm_model, parm = "mom_iq", level = 0.95)
```

```
             2.5 %    97.5 %
mom_iq 0.4949534 0.7249957
```

```r
stan_model <- stan_glm(kid_score ~ mom_iq,
                       data = kidiq)
posterior_interval(stan_model,
                   pars = "mom_iq",
                   prob = 0.95)
```

```
              2.5%     97.5%
mom_iq 0.4963677  0.7215823
```

```r
posterior <- spread_draws(stan_model, mom_iq)
mean(between(posterior_mom_iq, 0.60, 0.65))
```

```
0.31475
```

Let's practice!

BAYESIAN REGRESSION MODELING WITH RSTANARM