Final Project - MCMC Chain + Autocorrelation

Matthew Beebe

2024-12-09

```
setwd("C:\\Users\\Matt\\Downloads")
winedata <- read.csv("winemag-data-130k-v2.csv")
winedata <- winedata %>% filter(!is.na(points),
  !is.na(variety)) %>% mutate(variety_code = as.numeric(factor(variety)))
### Parameters: variety (discrete) and points (continuous)
edited_data <- list(n = nrow(winedata), v = 709, variety = winedata$variety_code + 1,
  points = winedata$points)

theoretical_posterior <- sum(winedata$points) / 129971 ### Theoretical posterior for points
theoretical_posterior
```

```
## [1] 88.44714
```

```
model <- "
  data{
    int<lower = 1> n;
    real points[n];
  }
  parameters{
    real mu;
    real <lower = 0> sigma;
  }
  model{
    mu ~ normal(90, 5);
    sigma ~ cauchy(0, 5);
    points ~ normal(mu, sigma);
  }"

stan_model <- stan(model_code = model, data = edited_data, iter = 6*1000, chains = 4)
```

```
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0.001035 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 10.35 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 6000 [  0%]  (Warmup)
## Chain 1: Iteration:  600 / 6000 [ 10%]  (Warmup)
## Chain 1: Iteration: 1200 / 6000 [ 20%]  (Warmup)
## Chain 1: Iteration: 1800 / 6000 [ 30%]  (Warmup)
## Chain 1: Iteration: 2400 / 6000 [ 40%]  (Warmup)
## Chain 1: Iteration: 3000 / 6000 [ 50%]  (Warmup)
## Chain 1: Iteration: 3001 / 6000 [ 50%]  (Sampling)
## Chain 1: Iteration: 3600 / 6000 [ 60%]  (Sampling)
## Chain 1: Iteration: 4200 / 6000 [ 70%]  (Sampling)
## Chain 1: Iteration: 4800 / 6000 [ 80%]  (Sampling)
## Chain 1: Iteration: 5400 / 6000 [ 90%]  (Sampling)
## Chain 1: Iteration: 6000 / 6000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 5.244 seconds (Warm-up)
## Chain 1:                5.404 seconds (Sampling)
## Chain 1:                10.648 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL ’anon_model’ NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0.000461 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 4.61 seconds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 6000 [  0%]  (Warmup)
## Chain 2: Iteration:  600 / 6000 [ 10%]  (Warmup)
## Chain 2: Iteration: 1200 / 6000 [ 20%]  (Warmup)
## Chain 2: Iteration: 1800 / 6000 [ 30%]  (Warmup)
## Chain 2: Iteration: 2400 / 6000 [ 40%]  (Warmup)
## Chain 2: Iteration: 3000 / 6000 [ 50%]  (Warmup)
## Chain 2: Iteration: 3001 / 6000 [ 50%]  (Sampling)
## Chain 2: Iteration: 3600 / 6000 [ 60%]  (Sampling)
## Chain 2: Iteration: 4200 / 6000 [ 70%]  (Sampling)
## Chain 2: Iteration: 4800 / 6000 [ 80%]  (Sampling)
## Chain 2: Iteration: 5400 / 6000 [ 90%]  (Sampling)
## Chain 2: Iteration: 6000 / 6000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 5.602 seconds (Warm-up)
## Chain 2:                5.008 seconds (Sampling)
## Chain 2:                10.61 seconds (Total)
## Chain 2:
##
## SAMPLING FOR MODEL ’anon_model’ NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0.000419 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 4.19 seconds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 6000 [  0%]  (Warmup)
## Chain 3: Iteration:  600 / 6000 [ 10%]  (Warmup)
## Chain 3: Iteration: 1200 / 6000 [ 20%]  (Warmup)
## Chain 3: Iteration: 1800 / 6000 [ 30%]  (Warmup)
## Chain 3: Iteration: 2400 / 6000 [ 40%]  (Warmup)
## Chain 3: Iteration: 3000 / 6000 [ 50%]  (Warmup)
## Chain 3: Iteration: 3001 / 6000 [ 50%]  (Sampling)
## Chain 3: Iteration: 3600 / 6000 [ 60%]  (Sampling)
## Chain 3: Iteration: 4200 / 6000 [ 70%]  (Sampling)
## Chain 3: Iteration: 4800 / 6000 [ 80%]  (Sampling)
## Chain 3: Iteration: 5400 / 6000 [ 90%]  (Sampling)
## Chain 3: Iteration: 6000 / 6000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 5.451 seconds (Warm-up)
## Chain 3:                5.625 seconds (Sampling)
## Chain 3:                11.076 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'anon_model' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0.000383 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 3.83 seconds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 6000 [  0%]  (Warmup)
## Chain 4: Iteration:  600 / 6000 [ 10%]  (Warmup)
## Chain 4: Iteration: 1200 / 6000 [ 20%]  (Warmup)
## Chain 4: Iteration: 1800 / 6000 [ 30%]  (Warmup)
## Chain 4: Iteration: 2400 / 6000 [ 40%]  (Warmup)
## Chain 4: Iteration: 3000 / 6000 [ 50%]  (Warmup)
## Chain 4: Iteration: 3001 / 6000 [ 50%]  (Sampling)
## Chain 4: Iteration: 3600 / 6000 [ 60%]  (Sampling)
## Chain 4: Iteration: 4200 / 6000 [ 70%]  (Sampling)
## Chain 4: Iteration: 4800 / 6000 [ 80%]  (Sampling)
## Chain 4: Iteration: 5400 / 6000 [ 90%]  (Sampling)
## Chain 4: Iteration: 6000 / 6000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 5.05 seconds (Warm-up)
## Chain 4:                4.788 seconds (Sampling)
## Chain 4:                9.838 seconds (Total)
## Chain 4:
```

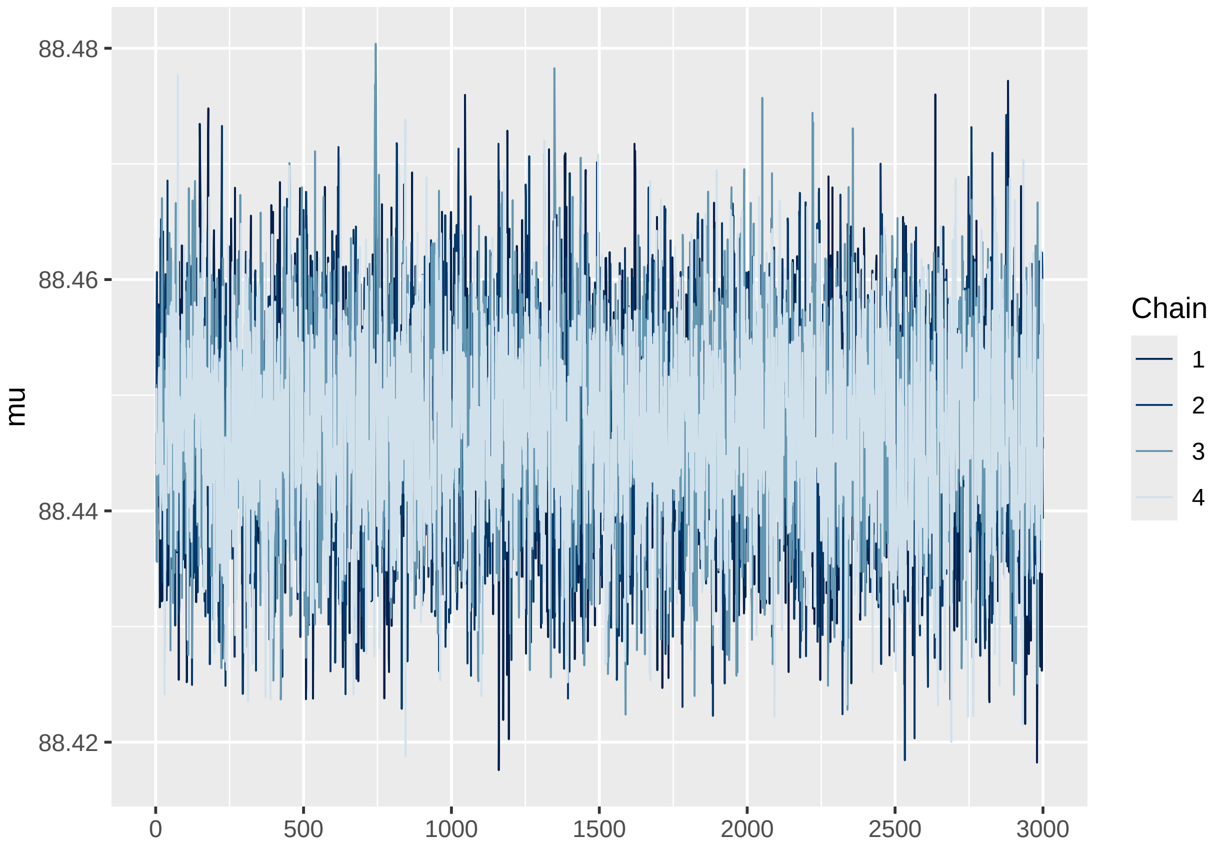
```r
summary(stan_model)
```

```
## $summary
##                 mean      se_mean          sd          2.5%           25%
## mu     8.844719e+01 1.100771e-04 0.008573750      88.43017  8.844148e+01
## sigma  3.039745e+00 5.771322e-05 0.006023031       3.02774  3.035741e+00
## lp__  -2.094830e+05 1.436245e-02 1.032029365 -209485.74719 -2.094834e+05
##                  50%           75%         97.5%     n_eff     Rhat
## mu     8.844726e+01  8.845316e+01  8.846383e+01  6066.632 1.000959
## sigma  3.039782e+00  3.043753e+00  3.051706e+00 10891.295 1.000040
## lp__  -2.094827e+05 -2.094822e+05 -2.094820e+05  5163.296 1.000496
##
## $c_summary
## , , chains = chain:1
##
##          stats
## parameter          mean          sd         2.5%           25%           50%
##     mu     8.844702e+01 0.008419683  8.843044e+01      88.44133      88.44708
##     sigma  3.039648e+00 0.006149391  3.027372e+00       3.03564       3.03957
##     lp__  -2.094830e+05 1.048076797 -2.094859e+05 -209483.36350 -209482.63716
##          stats
## parameter           75%         97.5%
##     mu     8.845286e+01  8.846321e+01
##     sigma  3.043781e+00  3.051894e+00
##     lp__  -2.094822e+05 -2.094820e+05
## 
## , , chains = chain:2
## 
##          stats
## parameter          mean          sd          2.5%           25%           50%
##     mu         88.44712 0.008856996  8.842909e+01  8.844113e+01  8.844742e+01
##     sigma       3.03980 0.006032372  3.027541e+00  3.035846e+00  3.039929e+00
##     lp__  -209483.01205 1.035715883 -2.094858e+05 -2.094835e+05 -2.094827e+05
##          stats
## parameter           75%         97.5%
##     mu     8.845319e+01  8.846434e+01
##     sigma  3.043907e+00  3.051673e+00
##     lp__  -2.094823e+05 -2.094820e+05
## 
## , , chains = chain:3
## 
##          stats
## parameter          mean          sd          2.5%           25%           50%
##     mu     8.844759e+01 0.008587772  8.843057e+01  8.844196e+01  8.844762e+01
##     sigma  3.039739e+00 0.005912366  3.028679e+00  3.035672e+00  3.039697e+00
##     lp__  -2.094830e+05 1.036288940 -2.094856e+05 -2.094833e+05 -2.094826e+05
##          stats
## parameter           75%         97.5%
##     mu     8.845344e+01  8.846389e+01
##     sigma  3.043532e+00  3.051646e+00
##     lp__  -2.094822e+05 -2.094820e+05
## 
## , , chains = chain:4
## 
##          stats
## parameter          mean          sd          2.5%           25%           50%
##     mu     8.844701e+01 0.008414139  8.843053e+01      88.44130  8.844692e+01
##     sigma  3.039791e+00 0.005997398  3.027961e+00       3.03583  3.039901e+00
##     lp__  -2.094830e+05 1.007087913 -2.094857e+05 -209483.33344 -2.094826e+05
##          stats
## parameter           75%         97.5%
##     mu     8.845313e+01  8.846310e+01
##     sigma  3.043758e+00  3.051485e+00
##     lp__  -2.094822e+05 -2.094820e+05
```

```r
post_mu <- rstan::extract(stan_model)$mu
posterior_total_mean <- mean(post_mu)
posterior_total_mean
```

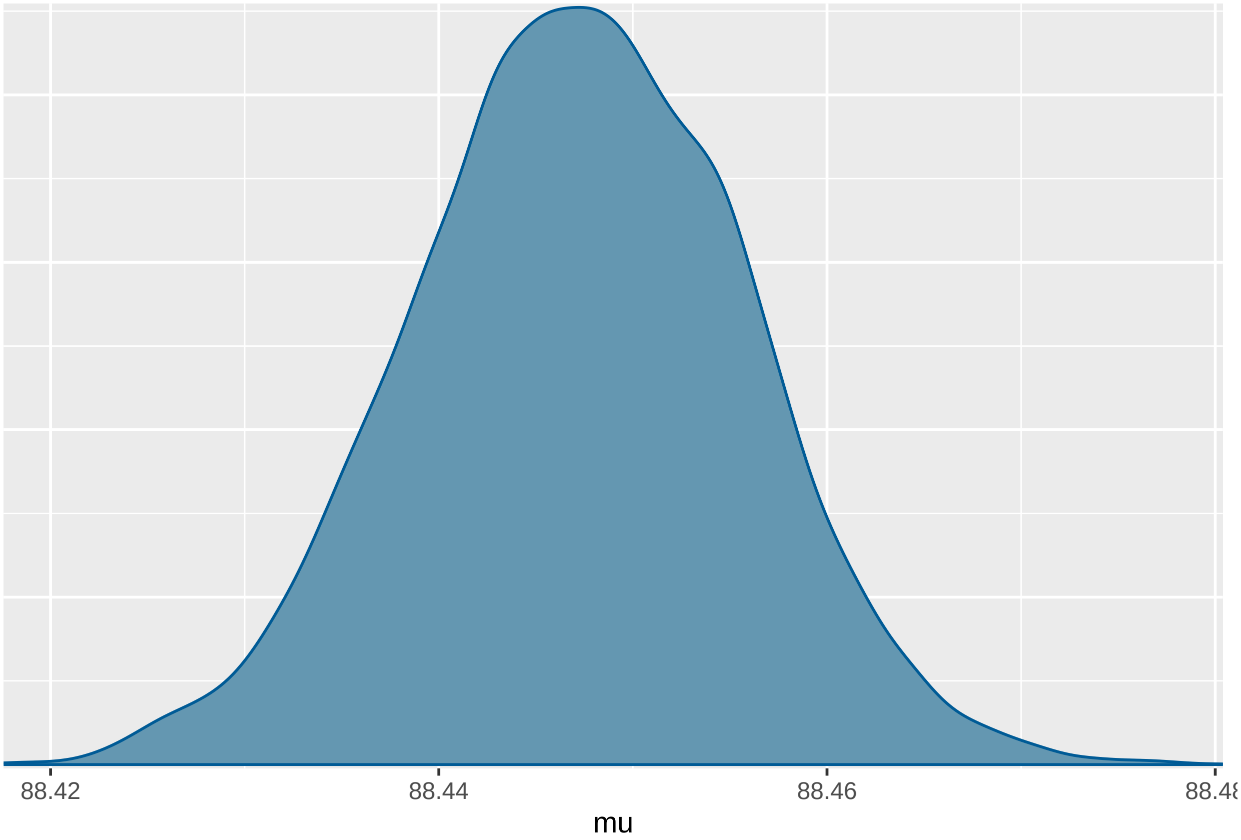
```
## [1] 88.44719
```

```r
mcmc_trace(stan_model, pars = "mu")
```

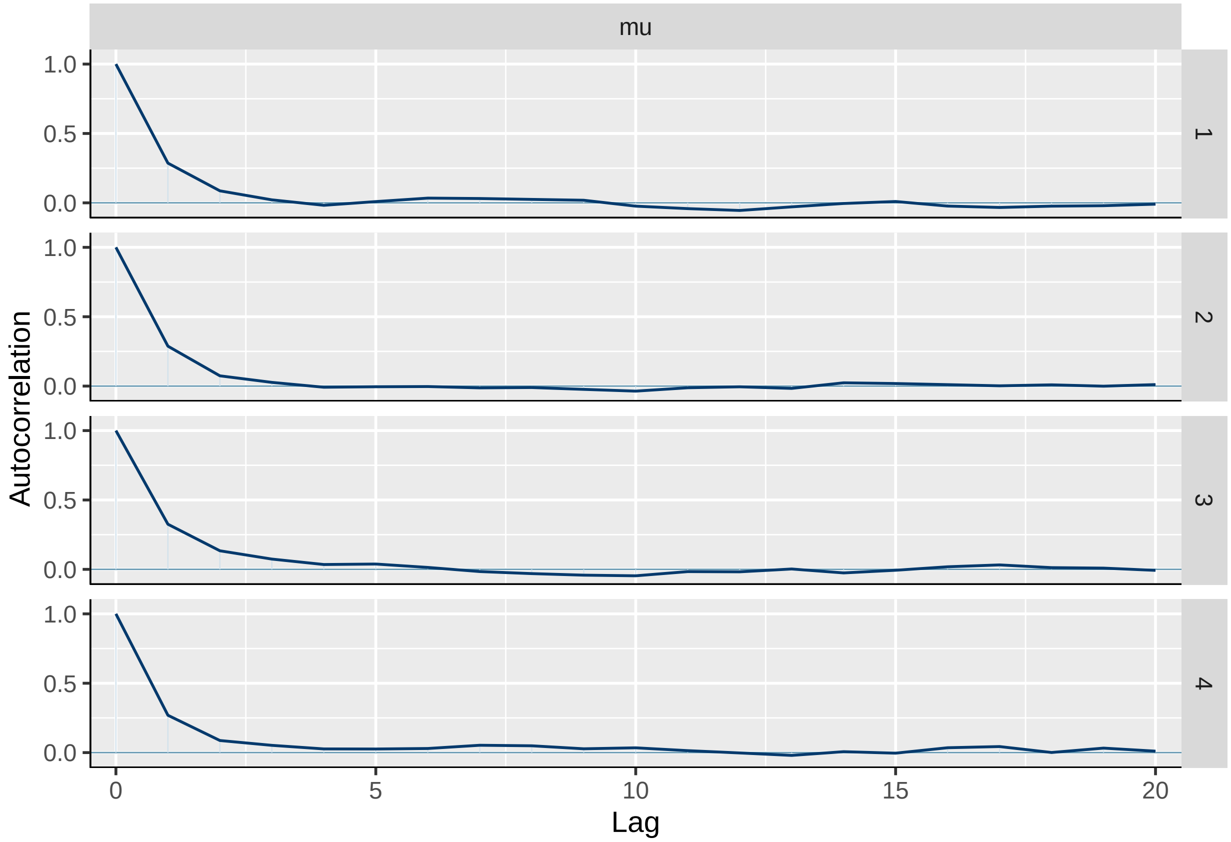


```r
mcmc_dens(stan_model, pars = "mu")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## i The deprecated feature was likely used in the bayesplot package.
##   Please report the issue at <https://github.com/stan-dev/bayesplot/issues/>.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



```
mcmc_acf(stan_model, pars = "mu")
```



```
neff_ratio(stan_model, pars = "mu")
```

```
## [1] 0.5055526
```

```
rhat(stan_model, pars = "mu")
```

```
## [1] 1.000959
```