Stat 341 – Homework 4

Gloria Grace

February 15, 2023

N1

```
fiji <- read_csv('https://sldr.netlify.app/data/fiji-filters.csv') |>
  mutate(household_annual_income = household_annual_income / 1000)
```

```
## Rows: 1006 Columns: 17
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): town, time_point, water_source, season, severe_diarrhea_adults, sev...
## dbl (9): household_id, n_adults, n_kids, n_total, household_annual_income, m...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Description

Based on the grid search model, the description of this model using the notation that I learn this week is:

$$\text{annual household income}_i \sim \text{Normal}(\mu, \sigma)$$

$$\mu \sim \text{Normal}(\text{mean}_1 = 57.7, \text{sd}_1 = 5)$$

$$\sigma \sim \text{Normal}(\text{mean}_2 = 41, \text{sd}_2 = 20)$$

### Logless

```
n_grid = 500
grid_income_model <-
  crossing(
    mu = seq(from = 50, to = 65, length.out = n_grid),
    sigma = seq(from = 80, to = 95, length.out = n_grid)
    ) |>
  mutate(
    # based on: http://www.salaryexplorer.com/salary-survey.php?loc=72&loctype=1
    prior_mu = dnorm(mu, mean = 57.7, sd = 5),
    prior_sigma = dnorm(sigma, mean = 41, sd = 20)
  ) |>
  rowwise() |>
  mutate(
    likelihood = dnorm(
      fiji$household_annual_income,
      mean = mu,
      sd = sigma,
      ) |>
  mutate(unstd_posterior = likelihood * (prior_mu + prior_sigma)) |>
  mutate(posterior = unstd_posterior / sum(unstd_posterior))
```

Where it goes wrong

```
n_grid = 500
grid_income_model <-
  crossing(
    mu = seq(from = 50, to = 65, length.out = n_grid),
    sigma = seq(from = 80, to = 95, length.out = n_grid)
    )

glimpse(grid_income_model)
```

Define the grid:

```
## Rows: 250,000
## Columns: 2
## $ mu    <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, ~
## $ sigma <dbl> 80.00000, 80.03006, 80.06012, 80.09018, 80.12024, 80.15030, 80.1~
```

```
grid_income_model <- grid_income_model |>
    mutate(
    prior_mu = dnorm(mu, mean = 57.7, sd = 5),
    prior_sigma = dnorm(sigma, mean = 41, sd = 20)
  )

glimpse(grid_income_model)
```

Define the priors

```
## Rows: 250,000
## Columns: 4
## $ mu          <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50~
## $ sigma       <dbl> 80.00000, 80.03006, 80.06012, 80.09018, 80.12024, 80.15030~
## $ prior_mu    <dbl> 0.02437551, 0.02437551, 0.02437551, 0.02437551, 0.02437551~
## $ prior_sigma <dbl> 0.002979735, 0.002971012, 0.002962307, 0.002953621, 0.0029~
```

I notice that the prior for sigma might be too small and think that the mistake might start from this point.

```
grid_income_model <- grid_income_model |>
  rowwise() |>
  mutate(
    logL = dnorm(
      fiji$household_annual_income,
      mean = mu,
      sd = sigma,
      log = TRUE
      ) |>
      sum()
    )|>
  ungroup()
```

```
glimpse(grid_income_model)
```

Compute the Likelihood

```
## Rows: 250,000
## Columns: 5
## $ mu          <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50~
## $ sigma       <dbl> 80.00000, 80.03006, 80.06012, 80.09018, 80.12024, 80.15030~
## $ prior_mu    <dbl> 0.02437551, 0.02437551, 0.02437551, 0.02437551, 0.02437551~
## $ prior_sigma <dbl> 0.002979735, 0.002971012, 0.002962307, 0.002953621, 0.0029~
## $ logL        <dbl> -5975.169, -5975.065, -5974.961, -5974.857, -5974.753, -59~
```

```
grid_income_model <- grid_income_model |>
  mutate(
    unscaled_ln_post = logL + log(prior_mu) + log(prior_sigma)
  )

glimpse(grid_income_model)
```

Compute the posterior

```
## Rows: 250,000
## Columns: 6
## $ mu               <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 5~
## $ sigma            <dbl> 80.00000, 80.03006, 80.06012, 80.09018, 80.12024, 80.~
## $ prior_mu         <dbl> 0.02437551, 0.02437551, 0.02437551, 0.02437551, 0.024~
## $ prior_sigma      <dbl> 0.002979735, 0.002971012, 0.002962307, 0.002953621, 0~
## $ logL             <dbl> -5975.169, -5975.065, -5974.961, -5974.857, -5974.753~
## $ unscaled_ln_post <dbl> -5984.699, -5984.598, -5984.497, -5984.396, -5984.295~
```

```
grid_income_model <- grid_income_model |>
  mutate(
    scaled_posterior =
      exp(unscaled_ln_post - max(unscaled_ln_post)
        )
  )

glimpse(grid_income_model)
```

Scale the posterior

```
## Rows: 250,000
## Columns: 7
## $ mu               <dbl> 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 5~
## $ sigma            <dbl> 80.00000, 80.03006, 80.06012, 80.09018, 80.12024, 80.~
## $ prior_mu         <dbl> 0.02437551, 0.02437551, 0.02437551, 0.02437551, 0.024~
## $ prior_sigma      <dbl> 0.002979735, 0.002971012, 0.002962307, 0.002953621, 0~
## $ logL             <dbl> -5975.169, -5975.065, -5974.961, -5974.857, -5974.753~
## $ unscaled_ln_post <dbl> -5984.699, -5984.598, -5984.497, -5984.396, -5984.295~
## $ scaled_posterior <dbl> 6.513992e-08, 7.210785e-08, 7.978891e-08, 8.825260e-0~
```

N2

```r
set.seed(19)
phones <- read_csv('https://osf.io/download/r8s6n/') |>
  rename(participant_id = pp,
         program = faculty,
         phone_use = total_a20,
         percent_private_phone_use = privateUse,
         use = smartphoneUse,
         ) |>
  mutate(fomo_score = (fomo1 + fomo2 + fomo3)) |>
  select(participant_id,
         program,
         age,
         gender,
         fatigue,
         fomo_score,
         boredom,
         phone_use,
         percent_private_phone_use,
         use,) |>
  mutate(phone_frequency = case_when(use == 1 ~ 'never',
                               use ==2 ~ 'once daily',
                               use == 3 ~ 'several times a day',
                               use == 4 ~ 'once an hour',
                               use == 5 ~ 'several times an hour',
                               use == 6 ~ 'every few minutes')) |>
  select(-use) |>
  drop_na(phone_use) |>
  filter(phone_use > 0)
```

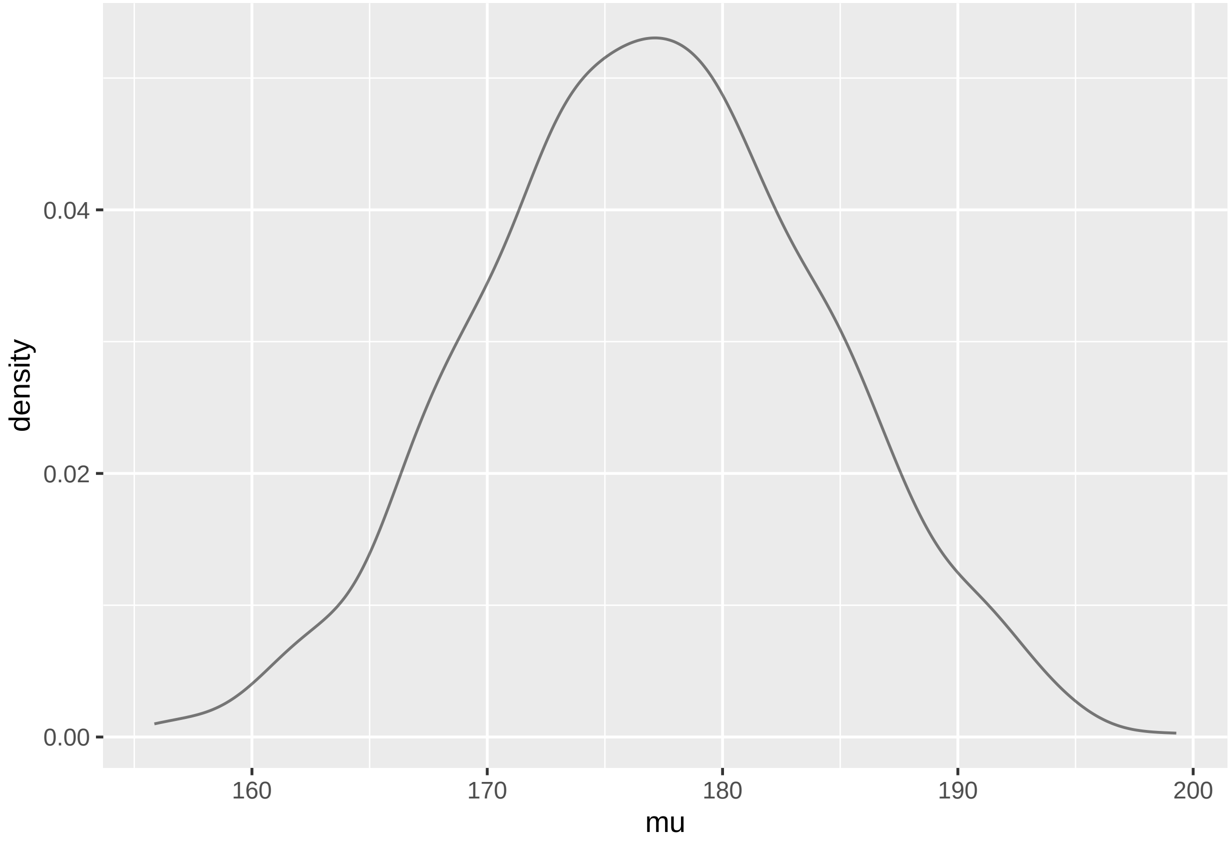
```
## Rows: 3234 Columns: 26
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (2): gender, faculty
## dbl  (19): pp, age, yearPhD, day, time, fatigue, boredom, total_b10, total_a...
## time  (5): startWork, endWork, startBreak, endBreak, exactTime
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Fit

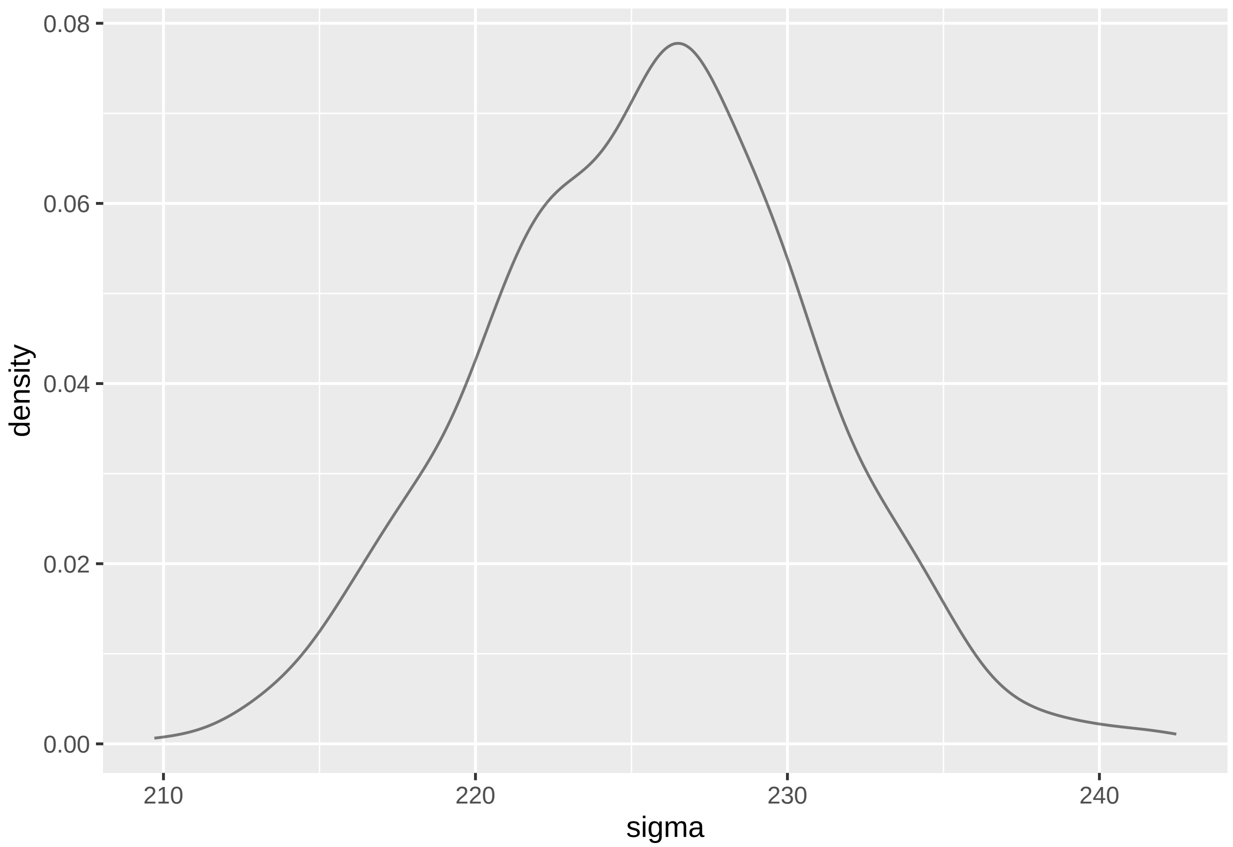
```r
model_descrip <- alist(
  # note variable name has to match actual data variable name
  phone_use ~ dnorm(mu, sigma),
  mu ~ dnorm(mean = 120, sd = 120),
  sigma ~ dnorm(mean = 300, sd = 100)
)

quap_phone_model <- quap(flist = model_descrip,
                          data = phones)
quap_phone_post_sample <- extract.samples(quap_phone_model, n = 1000)
```

Predictive and Critical Pt1

```
gf_dens(~mu, data = quap_phone_post_sample)
```



```
gf_dens(~sigma, data = quap_phone_post_sample)
```



Based on the prior predictive distribution, for the mu prior, it seems like the estimation is a bit off since based on the distribution the mean for mu lies around 175 to 180. For the sigma prior, the mean for sigma lies around 225 to 228 which I think it might still be acceptable.

Predictive and Critical Pt2

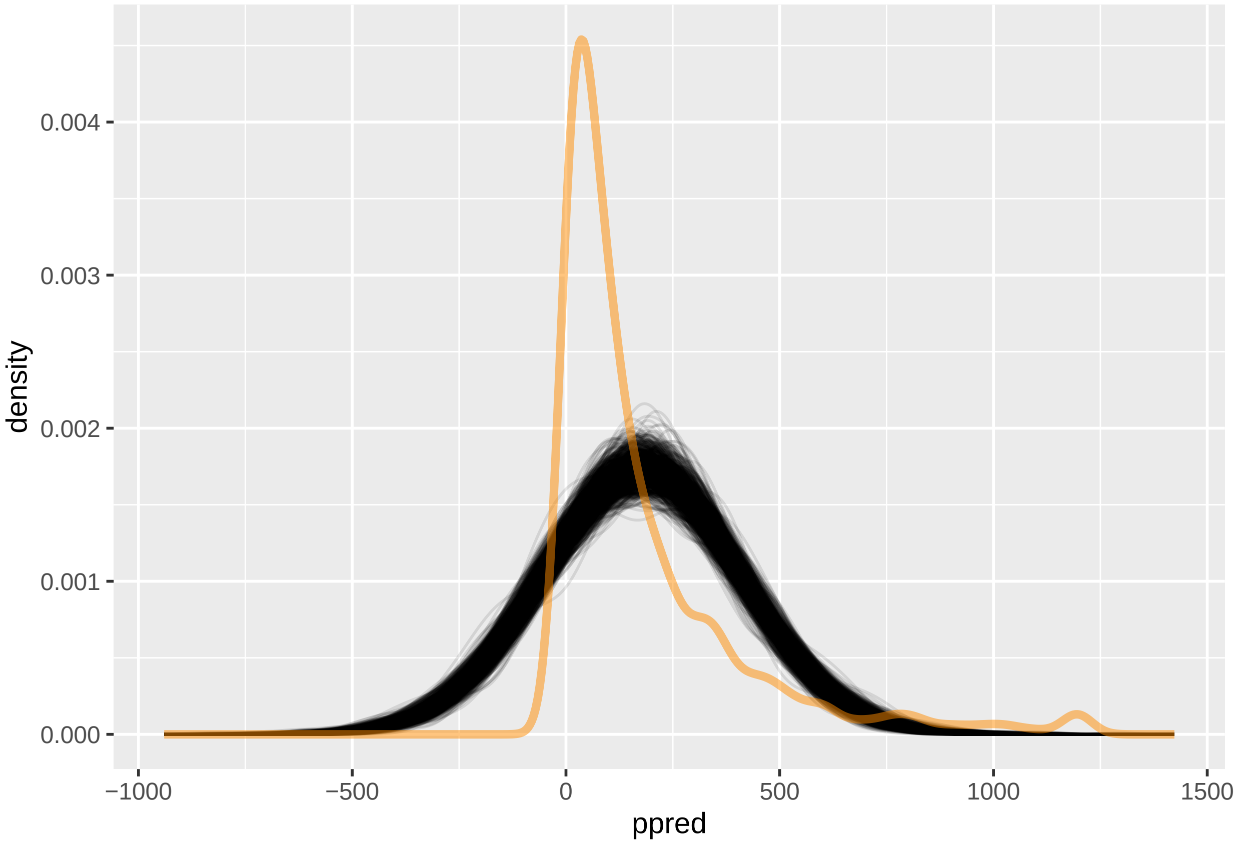
```
phone_ppred <- quap_phone_post_sample |>
  # add row numbers to "label" each sampled combo of mu & sigma
  mutate(row_num = c(1:n())) |>
  # work one row (one mu, sigma combination) at a time
  rowwise() |>
  # simulate a dataset for each row (= each mu, sigma combo)
  mutate(ppred = list(rnorm(nrow(phones),
                            mean = mu,
                            sd = sigma))) |>
  # keep only the row-ids and the simulated data
  select(row_num, ppred)

# note the structure: each row's ppred is a LIST of incomes nrow(phone) long
glimpse(phone_ppred)
```

```
## Rows: 1,000
## Columns: 2
## Rowwise:
## $ row_num <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,~
## $ ppred   <list> <13.5988927, 582.1345240, -113.9706633, 281.2752625, 290.8958~
```

```
# "unnest" results so each observed income is in its own row instead of a list. Now there are nrow(phon
phone_ppred <- phone_ppred |>
  unnest(cols = ppred)

# peek at the final result
glimpse(phone_ppred)
```

```
## Rows: 933,000
## Columns: 2
## $ row_num <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ ppred   <dbl> 13.598893, 582.134524, -113.970663, 281.275263, 290.895897, 43~
```

```
# graph simulated datasets - one line for each mu,sigma combo
# this will take a while unless you reduce n above in extract.samples()!
gf_dens(~ppred, group = ~row_num,
        data = phone_ppred,
        alpha = 0.1) |>
  # overlay actual data
  gf_dens(~phone_use,
          data = phones,
          inherit = FALSE,
          color = 'darkorange',
          linewidth = 1.5)
```



Based on what I see, the posterior predictive distribution is underestimated because of the model fit.

N3

Prior

Based on my web search, the rough estimate was 270 seconds per 20 minutes on average.

For sd, I estimate about 150.

Source: https://techjury.net/blog/how-much-time-does-the-average-american-spend-on-their-phone/#gref

Description

```
gamma_params(mean = 270, sd = 130)
```

```
## # A tibble: 1 x 6
##   shape   rate scale  mode  mean    sd
##   <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  4.31 0.0160  62.6  207.   270   130
```

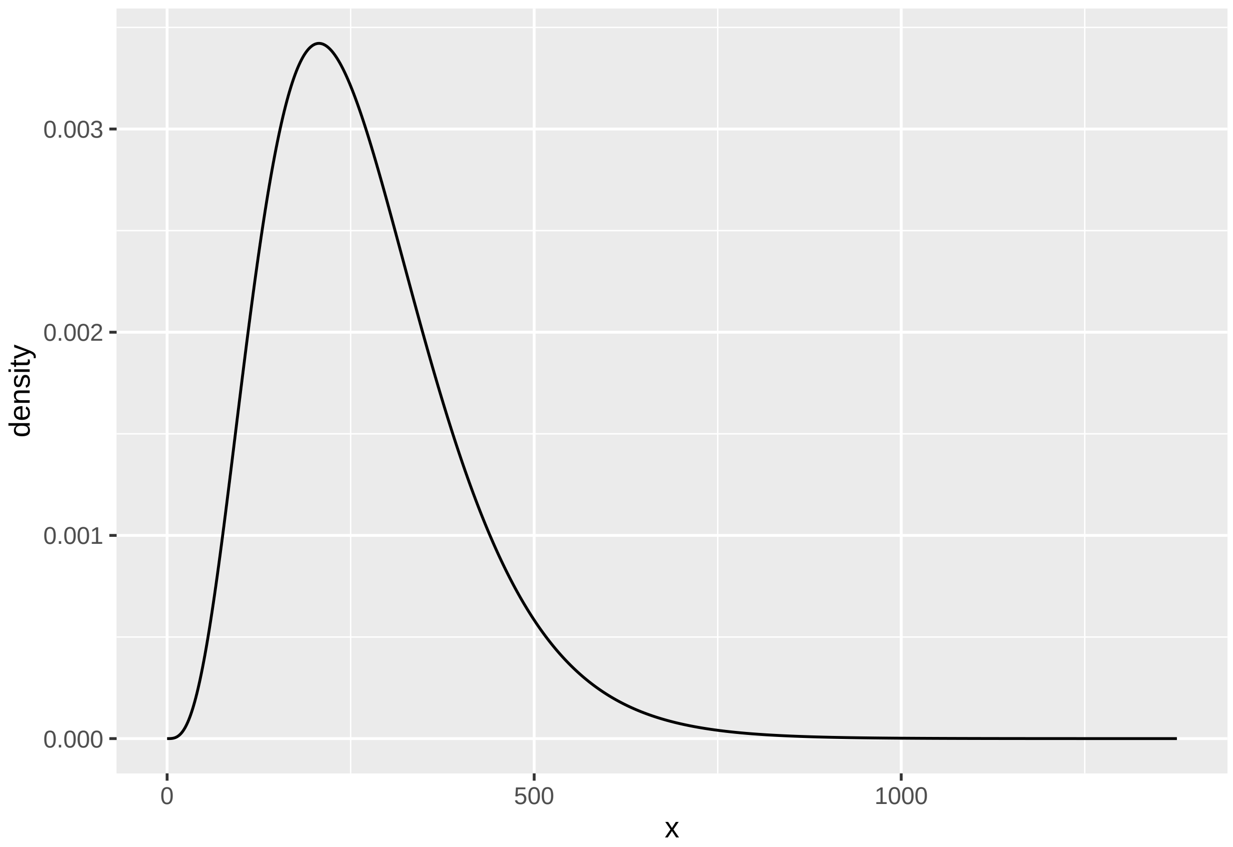
```
gamma_params(mean = 100, sd = 20)
```

```
## # A tibble: 1 x 6
##   shape  rate scale  mode  mean    sd
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    25  0.25     4    96   100    20
```

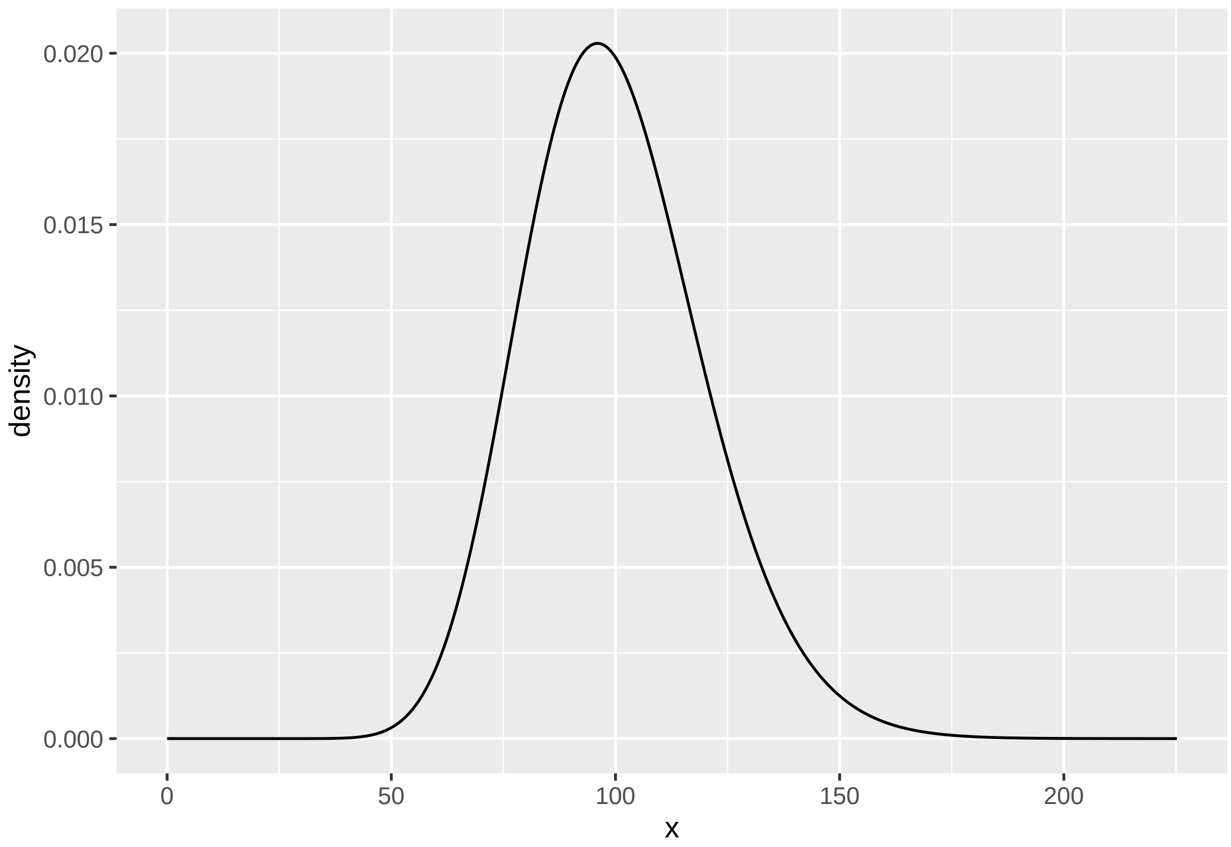
$$\text{phone use} \sim \text{Gamma}(\alpha, \lambda)$$
$$\alpha \sim \text{Gamma}(\text{shape}_1 = 4.31, \text{rate}_1 = 0.016)$$
$$\lambda \sim \text{Gamma}(\text{shape}_2 = 25, \text{rate}_2 = 0.25)$$

Predictive and Critical Pt1

```
gf_dist('gamma', shape = 4.31, rate = 0.016)
```



```
gf_dist('gamma', shape = 25, rate = 0.25)
```



For both, I was successfully estimated so that the prior is greater than 0.

Fit

```
model_descrip2 <- alist(
  # note variable name has to match actual data variable name
  phone_use ~ dgamma(alpha, lambda),
  alpha ~ dgamma(shape = 4.31, rate = 0.016),
  lambda ~ dgamma(shape = 25, rate = 0.25)
)

quap_phone_model2 <- quap(flist = model_descrip2,
                          data = phones)
quap_phone_post_sample2 <- extract.samples(quap_phone_model2, n = 1000)
```

Predictive and Critical Pt2

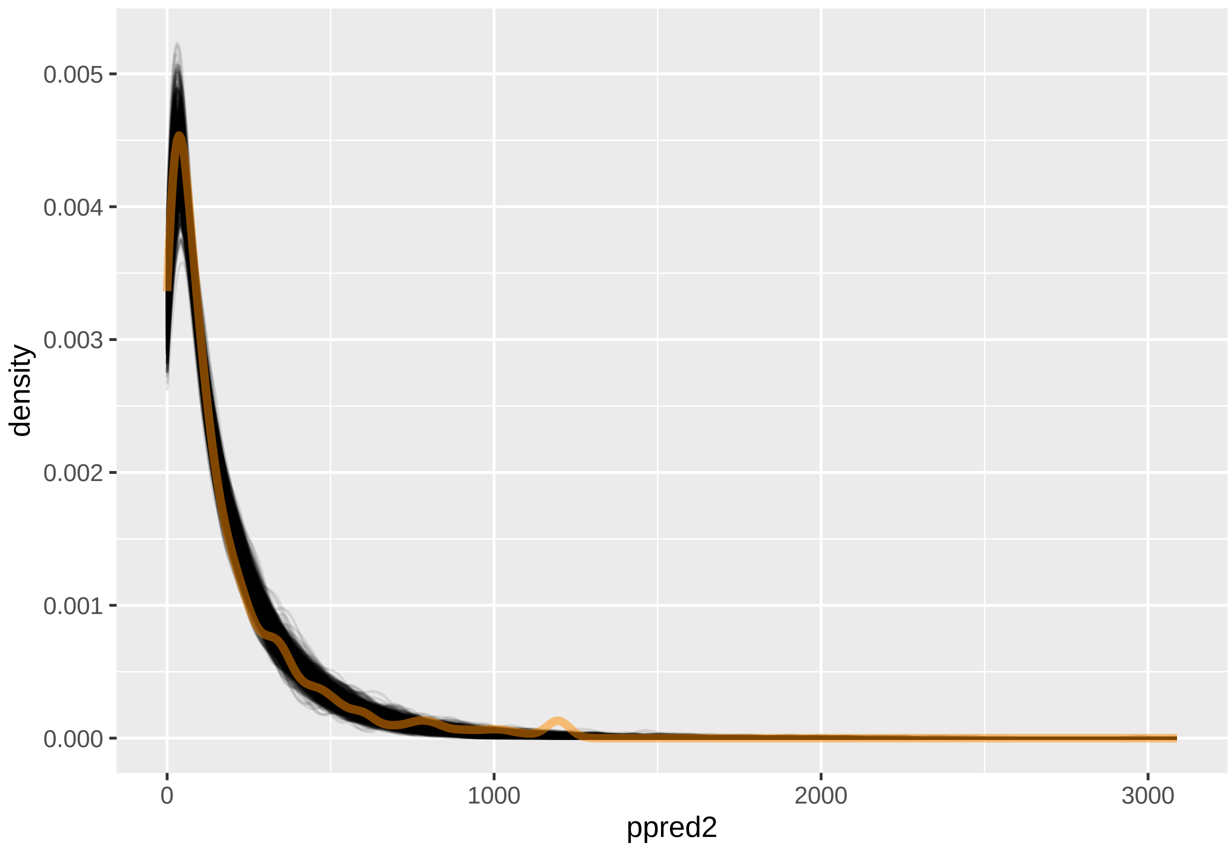
```
phone_ppred2 <- quap_phone_post_sample2 |>
  # add row numbers to "label" each sampled combo of mu & sigma
  mutate(row_num = c(1:n())) |>
  # work one row (one mu, sigma combination) at a time
  rowwise() |>
  # simulate a dataset for each row (= each mu, sigma combo)
  mutate(ppred2 = list(rgamma(nrow(phones),
                              shape = alpha,
                            rate = lambda))) |>
  # keep only the row-ids and the simulated data
  select(row_num, ppred2)

# note the structure: each row's ppred is a LIST of incomes nrow(phone) long
glimpse(phone_ppred2)
```

```
## Rows: 1,000
## Columns: 2
## Rowwise: 
## $ row_num <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,~
## $ ppred2  <list> <1.471821e+02, 2.101764e+02, 2.414893e+01, 4.445335e+02, 1.04~
```

```
# "unnest" results so each observed income is in its own row instead of a list. Now there are nrow(phon
phone_ppred2 <- phone_ppred2 |>
  unnest(cols = ppred2)

# peek at the final result
glimpse(phone_ppred2)
```

```
## Rows: 933,000
## Columns: 2
## $ row_num <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ ppred2  <dbl> 147.1821330, 210.1763605, 24.1489295, 444.5334975, 104.4721828~
```

```
# graph simulated datasets - one line for each mu,sigma combo
# this will take a while unless you reduce n above in extract.samples()!
gf_dens(~ppred2, group = ~row_num,
        data = phone_ppred2,
        alpha = 0.1) |>
  # overlay actual data
  gf_dens(~phone_use,
          data = phones,
          inherit = FALSE,
          color = 'darkorange',
          linewidth = 1.5)
```



Based on the model, my Gamma model seems to predict more close and similar to the actual model compared to using the normal distribution from before. Looking at both posterior predictive distribution plot, I would prefer using the second one to analyze the data because of its' similar accuracy.