

Homework 4

Sara Shao

9/10/2021

1a.

$$\psi = \log\left(\frac{\theta}{1-\theta}\right) \rightarrow \theta = \frac{e^\psi}{1+e^\psi}$$
$$\left|\frac{dh}{d\psi}\right| = \frac{e^\psi}{(1+e^\psi)^2}$$

$$p_\psi(\psi) = \frac{1}{B(\alpha, \beta)} \left(\frac{e^\psi}{1+e^\psi}\right)^{\alpha-1} \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right)^{\beta-1} \times \frac{e^\psi}{(1+e^\psi)^2} = \frac{1}{B(\alpha, \beta)} \left(\frac{e^\psi}{1+e^\psi}\right)^\alpha \left(\frac{1+e^\psi}{e^\psi}\right) \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right)^{\beta-1} \times \frac{e^\psi}{(1+e^\psi)^2}$$
$$= \frac{1}{B(\alpha, \beta)} \left(\frac{e^\psi}{1+e^\psi}\right)^\alpha \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right)^\beta \left(\frac{1+e^\psi}{1+e^\psi} - \frac{e^\psi}{1+e^\psi}\right)^{-1} \times \frac{1}{1+e^\psi} = \frac{1}{B(\alpha, \beta)} \left(\frac{e^\psi}{1+e^\psi}\right)^\alpha \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right)^\beta$$

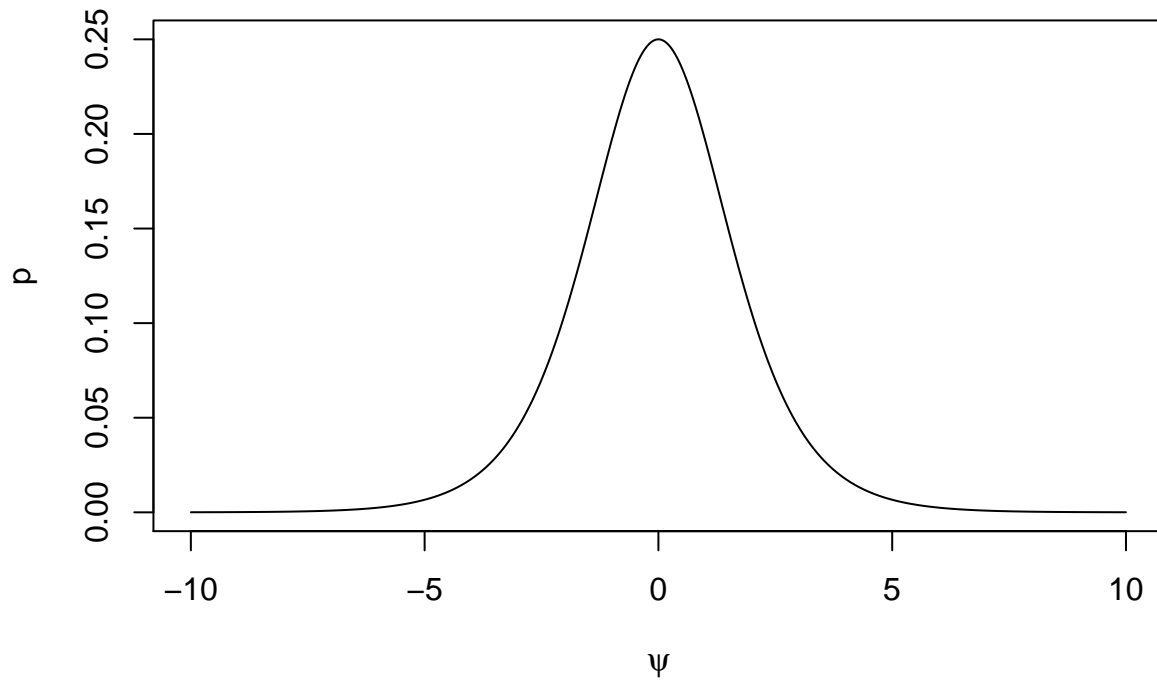
When $\alpha=\beta=1$:

$$\left(\frac{e^\psi}{1+e^\psi}\right)^\alpha \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right)^\beta = \left(\frac{e^\psi}{1+e^\psi}\right) \left(1 - \left(\frac{e^\psi}{1+e^\psi}\right)\right) = \left(\frac{e^\psi}{1+e^\psi}\right) \left(\frac{1}{1+e^\psi}\right) = \frac{e^\psi}{(1+e^\psi)^2}$$

```
psi.sim <- seq(-10, 10, length = 1000)

betaPsi = function(psi) {
  output <- exp(psi) / (1+exp(psi))^2
  return(output)
}

p <- apply(as.matrix(psi.sim), 1, betaPsi)
plot(psi.sim, p, type = "l", xlab = expression(psi))
```



1b.

$$\psi = \log(\theta) \rightarrow \theta = e^\psi$$

$$\left| \frac{dh}{d\psi} \right| = e^\psi$$

$$p_\psi(\psi) = \frac{\beta^\alpha}{\Gamma(\alpha)} (e^\psi)^{\alpha-1} e^{-\beta e^\psi} \times e^\psi = \frac{\beta^\alpha}{\Gamma(\alpha)} (e^\psi)^\alpha e^{-\beta e^\psi} = \frac{\beta^\alpha}{\Gamma(\alpha)} e^{\psi\alpha - \beta e^\psi}$$

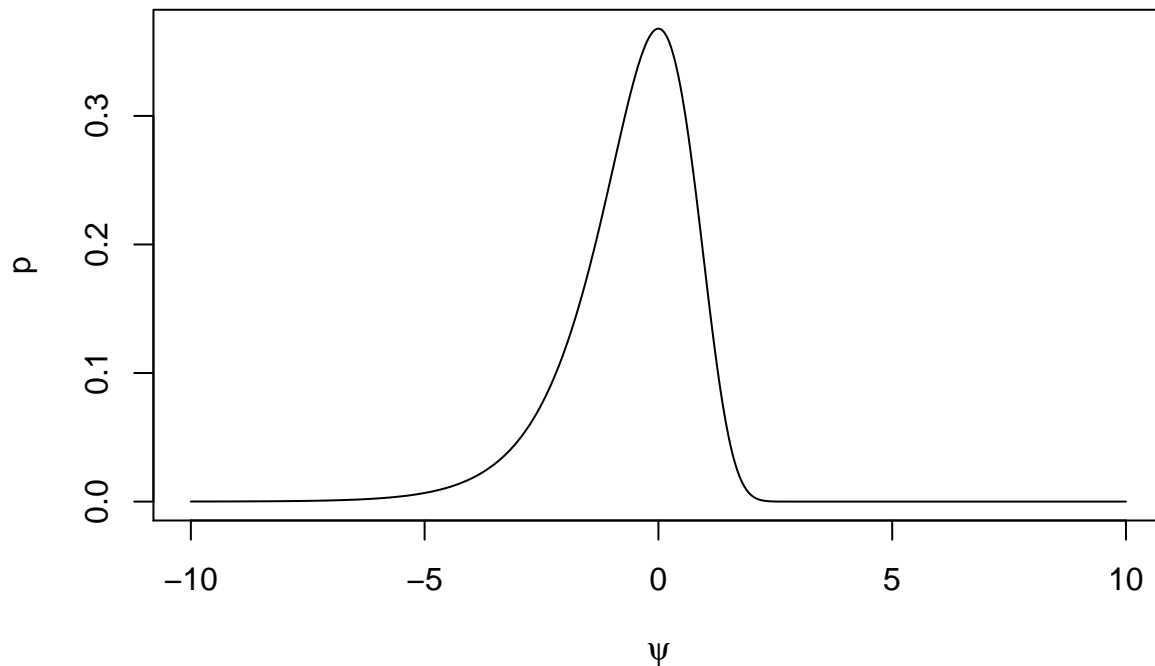
When $\alpha=\beta=1$:

$$e^{\psi\alpha - \beta e^\psi} = e^{\psi - e^\psi}$$

```
psi.sim <- seq(-10, 10, length = 1000)
```

```
gammaPsi = function(psi) {
  output <- exp(psi - exp(psi))
  return(output)
}
```

```
p <- apply(as.matrix(psi.sim), 1, gammaPsi)
plot(psi.sim, p, type = "l", xlab = expression(psi))
```



Task 4 (Finish for Homework)

```
# spurters
x = c(18, 40, 15, 17, 20, 44, 38)
# control group
y = c(-4, 0, -19, 24, 19, 10, 5, 10,
      29, 13, -9, -8, 20, -1, 12, 21,
      -7, 14, 13, 20, 11, 16, 15, 27,
      23, 36, -33, 34, 13, 11, -19, 21,
      6, 25, 30, 22, -28, 15, 26, -1, -2,
      43, 23, 22, 25, 16, 10, 29)

prior = data.frame(m = 0, c = 1, a = 0.5, b = 50)
findParam = function(prior, data){
  postParam = NULL
  c = prior$c
  m = prior$m
  a = prior$a
  b = prior$b
  n = length(data)
  postParam = data.frame(m = (c*m + n*mean(data))/(c + n),
                          c = c + n,
                          a = a + n/2,
                          b = b + 0.5*(sum((data - mean(data))^2)) +
                            (n*c *(mean(data)- m)^2)/(2*(c+n)))
  return(postParam)
}
postS = findParam(prior, x)
postC = findParam(prior, y)
```

Now, we can answer our original question: “What is the posterior probability that $\mu_S > \mu_C$?”

The easiest way to do this is to take a bunch of samples from each of the posteriors, and see what fraction of times we have $\mu_S > \mu_C$. This is an example of a Monte Carlo approximation (much more to come on this in the future).

To do this, we draw $N = 10^6$ samples from each posterior:

```
# sampling from two posteriors

# Number of posterior simulations
sim = 1000000

# initialize vectors to store samples
mus = NULL
lambdas = NULL
muc = NULL
lambdac = NULL

# Following formula from the NormalGamma with
# the update parameters accounted for below

lambdas = rgamma(sim, shape = postS$a, rate = postS$b)
lambdac = rgamma(sim, shape = postC$a, rate = postC$b)

mus = sapply(sqrt(1/(postS$c*lambdas)),rnorm, n = 1, mean = postS$m)
muc = sapply(sqrt(1/(postC$c*lambdac)),rnorm, n = 1, mean = postC$m)

# Store simulations
sim_means <- data.frame(mus = mus, muc = muc, diff = mus - muc)
sim_means$mus_greater = case_when(sim_means$diff > 0 ~ "yes",
                                   sim_means$diff <= 0 ~ "no")

head(sim_means)

##           mus           muc           diff mus_greater
## 1 34.08041 11.90184 22.178568          yes
## 2 15.72703 12.42221  3.304816          yes
## 3 16.29141 12.36786  3.923546          yes
## 4 26.23529 15.46697 10.768323          yes
## 5 18.11225  9.90077  8.211480          yes
## 6 18.80054 11.49981  7.300734          yes

counts <- sim_means %>%
  count(mus_greater)

c_mus_greater <- counts %>%
  filter(mus_greater == "yes") %>%
  pull(n)

c_mus_greater / sim

## [1] 0.970673
```

The posterior probability that the mean spurter improvement is greater than the mean non-spurter improvement is 0.971. Therefore, it is highly likely that students labeled spurters improve more than students who are not labeled spurters.

Task 5 (Finish for Homework)

Let's return back to the prior assumptions. There are a few ways that you can check that the prior conforms with our prior beliefs. Let's go back and check these. Draw some samples from the prior and look at them—this is probably the best general strategy. See Figure ?? . It's also a good idea to look at sample hypothetical datasets $X_{1:n}$ drawn using these sampled parameter values.

Please replicate a plot similar to Figure ?? and report your findings.

```
prior = data.frame(m = 0, c = 1, a = 0.5, b = 50)

# Number of prior simulations
sim2 = 500

# initialize vectors to store samples
mu_prior = NULL
lambda_prior = NULL

# Following formula from the NormalGamma

lambda_prior = rgamma(sim2, shape = prior$a, rate = prior$b)

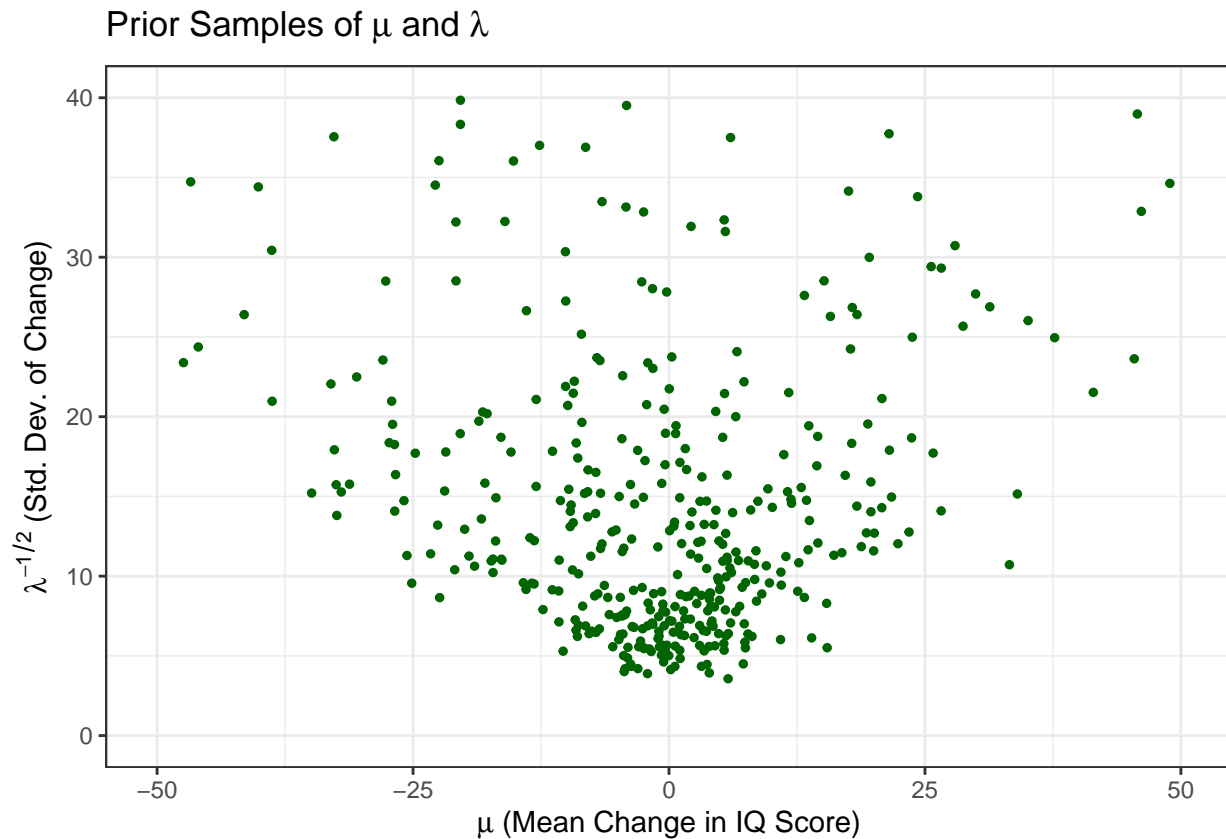
mu_prior = sapply(sqrt(1/(prior$c*lambda_prior)), rnorm, n = 1, mean = prior$m)

# Store simulations
prior_sim = data.frame(lambda = lambda_prior, mu = mu_prior)

prior_sim$lambda = prior_sim$lambda^{-0.5}

# Plot the simulations
ggplot(data = prior_sim, aes(x = mu, y = lambda)) +
  geom_point(size = 1, color = "dark green") +
  xlim(-50, 50) +
  ylim(0, 40) +
  labs(x = expression(paste(mu, " (Mean Change in IQ Score)")),
       y = expression(paste(lambda^{-1/2}, " (Std. Dev. of Change)"))) +
  ggtitle(expression(paste("Prior Samples of ", mu, " and ", lambda))) +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_bw()

## Warning: Removed 101 rows containing missing values (geom_point).
```



This graph shows that when the standard deviation of change is close to 0, the observed mean changes tend to be close to 0 as well. When the standard deviation of change is large, the observed mean changes tend to vary more. In this graph, we can also see that the observed standard deviations never reach below a 3. These findings from the graph are consistent with our prior beliefs that we don't know if students will improve or not on average (mean change centered around 0), and we wouldn't expect the standard deviation of change to be very small (very unlikely that the standard deviation is less than 3).