Lab 1 code

Christian Kammerer(chrka821), Jakob Lindner (jakli758)

```r
##1.1

library(ggplot2)
n <- 78
f <- 35
s <- n - f
alpha <- 7
beta <- 7
drawsize <- 10000

posterior_sample <- function(n, alpha, beta, f, s){
  return(rbeta(n, alpha + s, beta + f))
}

n_draws <- posterior_sample(drawsize, alpha, beta, f, s)
n_means <- numeric(drawsize)
n_sds <- numeric(drawsize)

for(i in 1:drawsize){
  n_means[[i]] <- mean(n_draws[1:i])
  n_sds[[i]] <- sd(n_draws[1:i])
}

n_sds[[1]] <- 0

a <- alpha + s
b <- beta + f

# Calculate mean and standard deviation according to beta distribution
true_mean <- a / (a + b)
true_sd <- sqrt((a * b) / ((a + b)^2 * (a + b + 1)))

# Create a data frame for horizontal line to allow for color mapping
hline_data <- data.frame(x = c(1, 10000), y = true_mean)

p <- ggplot() +
  aes(x = (1:10000), y = n_means, color = "Sample Mean") +
  geom_line() +
  xlab("Number of Draws") +
  ylab("Mean across the first i draws") +
  ggtitle("Change in sample mean for increasing sample size") +
  geom_line(data = hline_data, aes(x = x, y = y, color = "True Mean"), linetype = "dashed") +
  scale_color_manual(values = c("Sample Mean" = "red", "True Mean" = "limegreen")) +
  theme(legend.position = "bottom")
```

```
print(p)
hline_data <- data.frame(x = c(1, 10000), y = true_sd)
p <- ggplot() +
  aes(x = (1:10000), y = n_sds, color = "Sample SD") +
  geom_line() +
  xlab("Number of Draws") +
  ylab("Mean across the first i draws") +
  ggtitle("Change in sample SD for increasing sample size") +
  geom_line(data = hline_data, aes(x = x, y = y, color = "True SD"), linetype = "dashed") +
  scale_color_manual(values = c("Sample SD" = "blue", "True SD" = "limegreen")) +
  theme(legend.position = "bottom")

print(p)


##1.2
prob <- mean(n_draws > 0.5)
actual_prob <- 1 - pbeta(q = 0.5, shape1 = alpha + s, shape2 = beta + f)
cat("Posterior probability: ", prob, ", exact value: ", actual_prob)


## 1.3
phis <- n_draws / (1 - n_draws)
density_phi <- density(phis)

# Plot histogram
hist(phis, main = expression(paste("Distribution of ", phi)),
     xlab = expression(phi), freq = FALSE, breaks=20)

# Add density line
lines(density_phi, col = "red")

##2.1

library(asbio)
library(bayestestR)

y_vec <- c(22, 33, 31, 49, 65, 78, 17, 24)
mu <- 3.65

calc_tau_sq <- function(y_vec, mu){
  return(sum((log(y_vec) - mu)^2)/length(y_vec))
}

tau_sq <- calc_tau_sq(y_vec, mu)
sample_var <- rinvchisq(n = 10000, df = length(y_vec) - 1) * tau_sq

density_sample_var <- density(sample_var)

# Plot histogram
hist(sample_var, main = expression(paste("Distribution of ", sigma^2)),
     xlab = expression(sigma^2), breaks = 50, freq = FALSE)
```

```
# Add density line
lines(density_sample_var, col = "red")

##2.2

ginis <- sapply(sample_var, function(sigma_sq) 2 * pnorm(q = sqrt(sigma_sq)/sqrt(2)) - 1)

density_ginis <- density(ginis)

# Plot histogram
hist(ginis, main = "Distribution of Gini Coefficient",
     xlab = "Gini coefficient", breaks = 50, freq = FALSE)

lines(density_ginis, col = "blue")

##2.3
ci <- quantile(ginis, probs = c(0.025, 0.975))

##2.4
hpdi <- hdi(ginis, ci = 0.95)
cat("CI: ", ci)
print(hpdi)

##3.1
lambda_grid <- seq(0, 20, length.out = 1000)
unnorm_posterior <- lambda_grid^24 * exp(-7 * lambda_grid - lambda_grid^2 / 50)
posterior <- unnorm_posterior / sum(unnorm_posterior * diff(lambda_grid)[1])

p <- ggplot() +
  aes(x=lambda_grid, y=posterior, col = "red") +
  geom_line() +
  ggtitle(expression(paste("Normalized Posterior Distribution of ", lambda))) +
  xlab(expression(lambda)) +
  ylab("Probability Density") +
  theme(legend.position="none")
print(p)

##3.2
mode <- lambda_grid[which.max(posterior)]
cat("mode: ", mode)
```