

Lab 5

Name here

Suppose you have been hired as a statistical consultant by a NBA expansion team that will be moving to Big Sky, MT. Your goal is to help identify basketball players to bring to Montana.

Specifically, the team has one more spot to fill and is looking to add a free throw shooting specialist. Here are two options:

- Bugs Bunny. Bugs Bunny is a shooting guard that was 2 for 7 on free throws last year.
- Tasmanian Devil. Tasmanian Devil is a center that was 25 for 40 on free throws last year.

1.

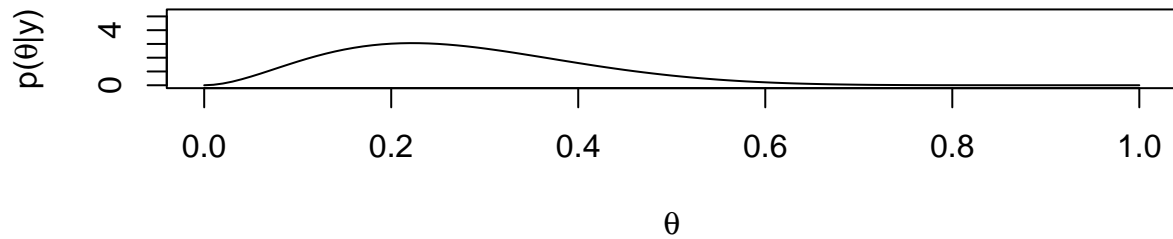
Use a binomial model, along with uniform prior of the probability parameter, to model the free throw shooting for Bugs Bunny and Tasmanian Devil.

a (4 points)

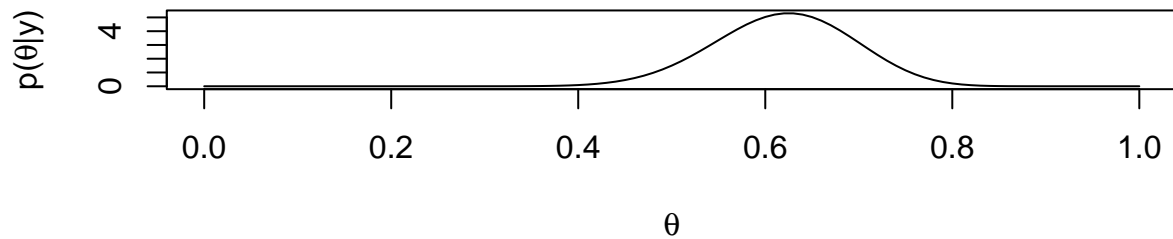
Plot and summarize the posterior distribution for each player.

```
par(mfcol=c(2,1))
x <- seq(0,1, by=.01)
y.max <- max(dbeta(x,26,16))
plot(x,dbeta(x,3,8), main='Bugs Bunny Posterior', type='l', ylim=c(0,y.max),ylab=expression(paste('p(',
plot(x,dbeta(x,26,16), main='Tasmanian Devel Posterior', type='l', ylim=c(0,y.max),ylab=expression(paste('p('
```

Bugs Bunny Posterior



Tasmanian Devil Posterior



b (4 points)

Using the posteriors above, compute the probability that Bugs Bunny has a higher shooting percentage (θ parameter in the binary setting).

```
num.sims <- 5000
bugs <- rbeta(num.sims, 3, 8)
td <- rbeta(num.sims, 26, 16)
bugs.better <- mean(bugs > td)
```

In this setting, the probability that Bugs Bunny has a higher shooting percentage is 0.017

c (2 points) If both players had 25 free throws, who do you think will make more free throws? Why?

I'd think that Tasmanian Devil would have a better outcome, but with 25 free throws it would be possible that Bugs Bunny does better, but highly unlikely.

2.

Now assume that we know that shooting guards, like Bugs Bunny, tend to make about 80 percent of free throws and centers, like Tasmanian Devil, tend to make about 60 percent of free throws. This information can be imparted by more informative priors: Use Beta(40,10) for Bugs Bunny and Beta(30,20) for Tasmanian Devil as the prior distributions.

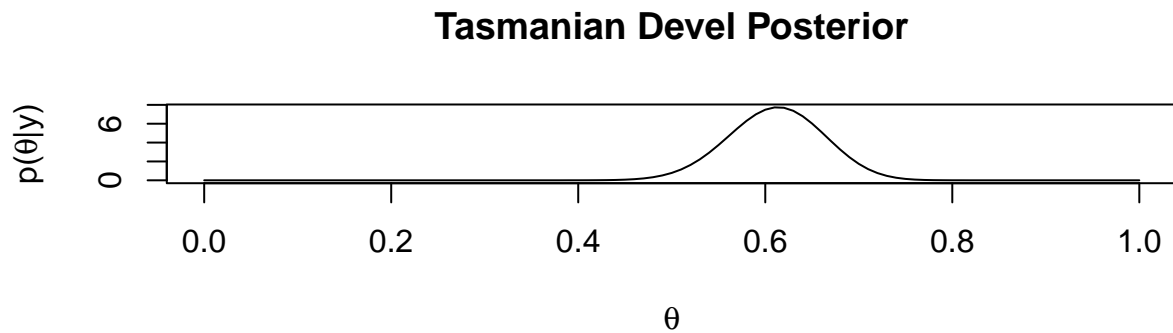
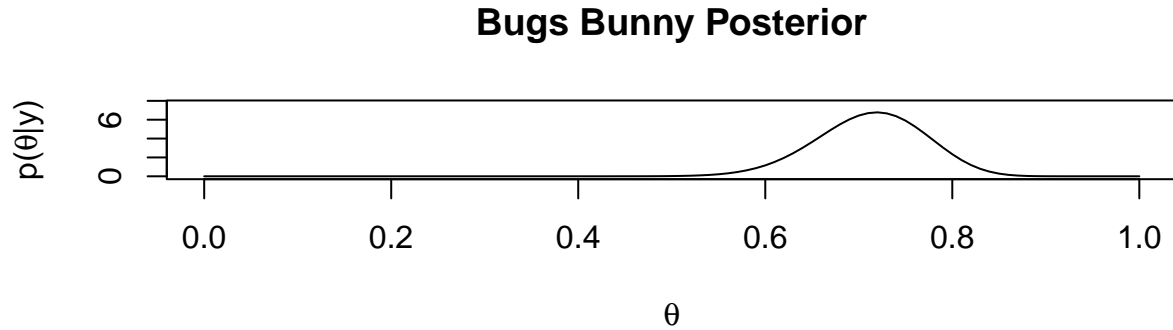
a (4 points)

Plot and summarize the posterior distribution for each player.

```

par(mfcol=c(2,1))
x <- seq(0,1, by=.01)
y.max <- max(c(dbeta(x,42,17),dbeta(x,55, 35)))
plot(x,dbeta(x,42,17), main='Bugs Bunny Posterior', type='l', ylim=c(0,y.max),ylab=expression(paste('p(
plot(x,dbeta(x,55,35), main='Tasmanian Devel Posterior', type='l', ylim=c(0,y.max),ylab=expression(past

```



b (4 points)

Using the posteriors above, compute the probability that Bugs Bunny has a higher shooting percentage (θ parameter in the binary setting).

```

num.sims <- 5000
bugs <- rbeta(num.sims, 42, 17)
td <- rbeta(num.sims, 55, 35)
bugs.better <- mean(bugs > td)

```

In this case, the probability that bugs bunny has a higher shooting percentage is 0.895

c (2 points)

If both players had 25 free throws, who do you think will make more free throws? Why?

Now I'd bet on Bugs Bunny making more free throws.

3. (4 points)

Reflect on the results from question a and question b. Which analysis, and associated prior, do you prefer, why?

The prior distribution can be highly influential for the posterior and that is not necessarily good or bad. Rather it depends on how reasonable the prior belief is and how much data is available to change or update the prior beliefs.

4. Bayesian Analysis on a continuous outcome

Let's model housing prices using our King County, WA dataset.

A normal distribution seems like a reasonable starting point, although we will see some potential issues.

$$price = \beta_0 + \epsilon \quad \epsilon \sim N(0, \sigma^2)$$

So we have two parameters in this model β_0 and σ^2

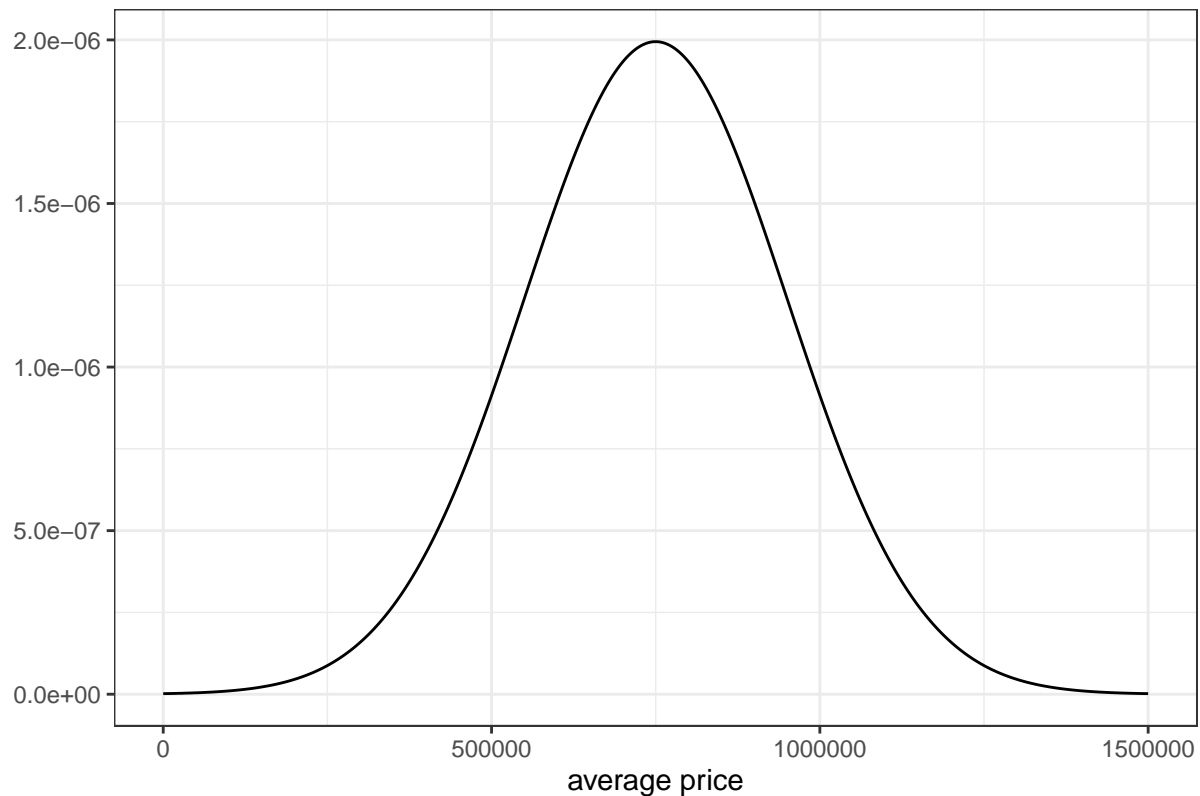
a. Specify and plot a prior distributions for β_0

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr  1.0.1
## v tibble  3.1.8      v dplyr  1.1.0
## v tidyr   1.3.0      v stringr 1.5.0
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

temp_seq <- seq(0, 1500000, length.out = 500)
tibble(vals = c(dnorm(temp_seq, 750000, 200000)),
       temperature = temp_seq) %>%
  ggplot(aes(x = temperature, y = vals)) +
  geom_line() + theme_bw() +
  ggtitle(paste('Prior distribution for mean price: N($750,000,$200,000^2)')) +
  xlab('average price') + ylab('')
```

Prior distribution for mean price: $N(\$750,000, \$200,000^2)$



b. (4 points)

Using your prior from part a, along with a prior such that $\sigma \sim \text{Gamma}(.00001, .00001)$, model the mean housing price in the Seattle dataset.

```
seattle <- read_csv('http://www.math.montana.edu/ahoegh/teaching/stat408/datasets/SeattleHousing.csv')

## Rows: 869 Columns: 14
## -- Column specification -----
## Delimiter: ","
## dbl (14): price, bedrooms, bathrooms, sqft_living, sqft_lot, floors, waterfr...
##
## 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.
y <- seattle$price
n <- nrow(seattle)
```

For reference, you should expect to get similar values to `summary(lm(price ~ 1, data=seattle))`

```
model_normal<- "model{
  # Likelihood
  for (i in 1:n){
    y[i] ~ dnorm(beta0, 1/sigma^2)
  }

  # Prior
  beta0 ~ dnorm(750000, 1/200000^2)
```

```

    sigma ~ dgamma(.00001, .00001)
  }"

library(rjags)

## Loading required package: coda
## Linked to JAGS 4.3.0
## Loaded modules: basemod,bugs
dataList = list(y = y, n = n)

model <- jags.model(file = textConnection(model_normal), data = dataList)

## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 869
##   Unobserved stochastic nodes: 2
##   Total graph size: 881
##
## Initializing model
update(model, n.iter = 5000) # warmup

# take samples
posterior_sample <- coda.samples(model,
                                variable.names = c("beta0", 'sigma'),
                                n.iter = 10000)

summary(posterior_sample)

##
## Iterations = 6001:16000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean      SD Naive SE Time-series SE
## beta0 633617 21341    213.4      213.4
## sigma 632688 15110    151.1      219.0
##
## 2. Quantiles for each variable:
##
##           2.5%    25%    50%    75%  97.5%
## beta0 592337 619026 633523 648038 675675
## sigma 603988 622508 632323 642815 663118

```