Homework 4

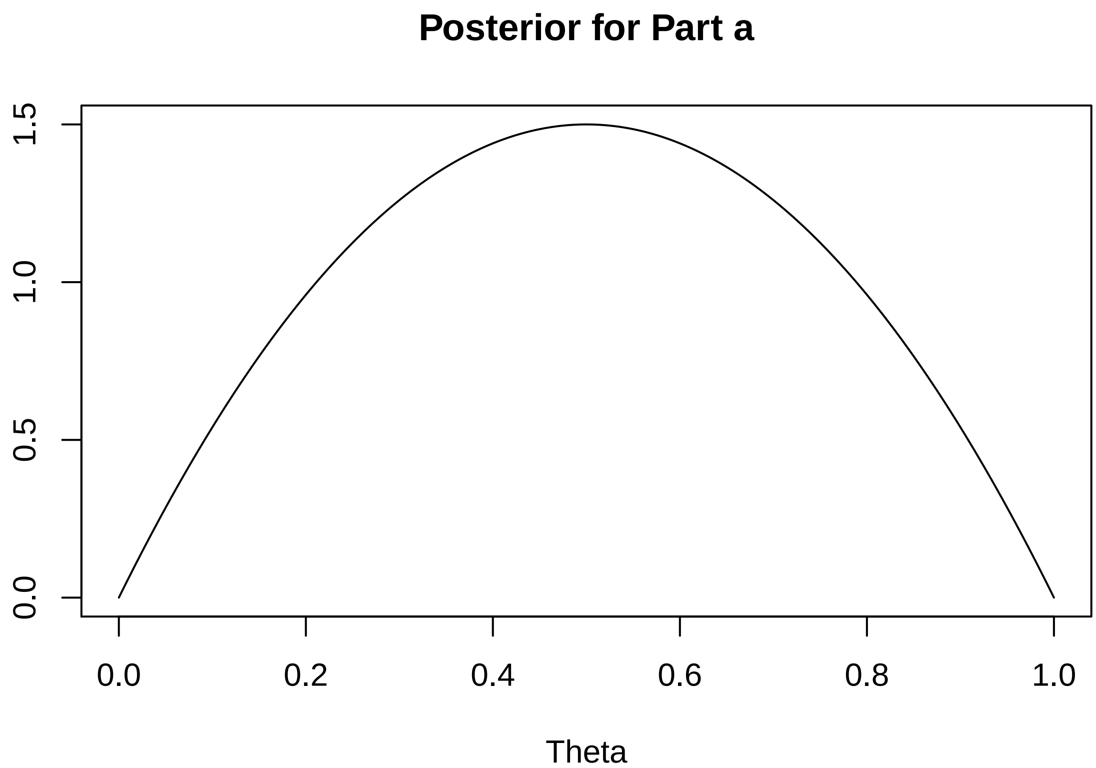
Austin Kao

Hoff, 3.10

a.

$$
\begin{aligned}
p_\theta(\theta) &= \frac{1}{\beta(a,b)}\theta^{a-1}(1-\theta)^{b-1}I(\theta>0)\\
\psi &= \log\left(\frac{\theta}{1-\theta}\right)\\
\theta = h(\psi) &= \frac{e^\psi}{1+e^\psi}\\
\frac{dh}{d\psi} &= \frac{e^\psi(1+e^\psi)-e^\psi(e^\psi)}{(1+e^\psi)^2} = \frac{e^\psi}{1+2e^\psi+e^{2\psi}}\\
e^\psi &= e^{\log\left(\frac{\theta}{1-\theta}\right)} = \frac{\theta}{1-\theta}\\
\frac{dh}{d\psi} &= \frac{\frac{\theta}{1-\theta}}{1+2\left(\frac{\theta}{1-\theta}\right)+\left(\frac{\theta}{1-\theta}\right)^2} = \frac{\theta(1-\theta)}{(1-\theta)^2+2\theta(1-\theta)+\theta^2} = \frac{\theta-\theta^2}{1-2\theta+\theta^2+2\theta-2\theta^2+\theta^2} = \theta(1-\theta)\\
p_\psi(\psi) &= p_\theta\left(h(\psi)\right)\times\left|\frac{dh}{d\psi}\right| = p_\theta(\theta)\times\left|\frac{dh}{d\psi}\right|\\
p_\psi(\psi) &= \frac{1}{\beta(a,b)}\theta^{a-1}(1-\theta)^{b-1}I(\theta>0)\times\left|\theta(1-\theta)\right| = \frac{1}{\beta(a,b)}\theta^{a-1}(1-\theta)^{b-1}\theta(1-\theta)I(\theta>0)\\
p_\psi(\psi) &\propto \theta^{a-1}(1-\theta)^{b-1}\theta(1-\theta)I(\theta>0) = \theta^a(1-\theta)^bI(\theta>0)\\
p_\psi(\psi) &\sim \text{Beta}(a+1,b+1)
\end{aligned}
$$

```r
### Plot p(psi) for a=1, b=1
### Define beta function
beta <- function (a, b, theta) {
  return (factorial(a+b-1)/(factorial(a-1)*factorial(b-1))*theta^(a-1)*(1-theta)^(b-1))
}
### Define range of theta values
theta = seq(0, 1, length.out = 1000)
### Plot posterior beta function for a=1, b=1
a = 1
b = 1
plot(theta, beta(a+1, b+1, theta),
     type = "l", ylab = "", xlab = "Theta", main = "Posterior for Part a")
```

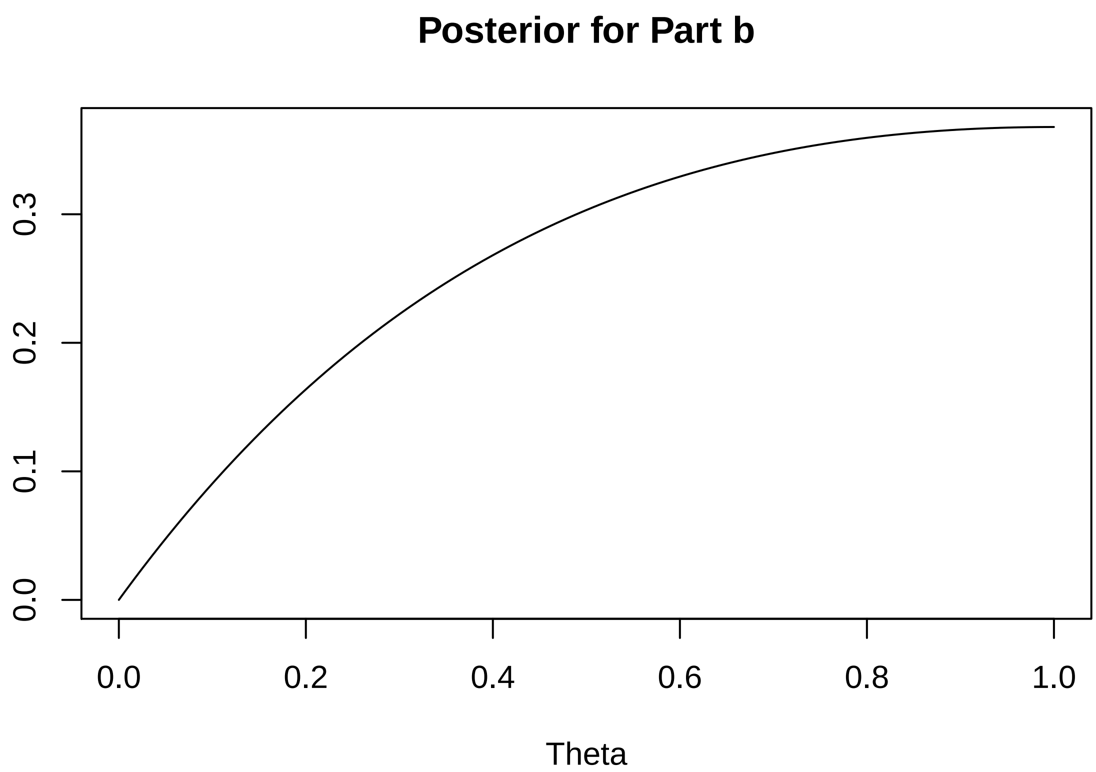


b.

$$
\begin{aligned}
p_\theta(\theta) &= \frac{b^a}{\Gamma(a)}\theta^{a-1}e^{-b\theta}I(\theta > 0) \\
\psi &= \log(\theta) \\
\theta = h(\psi) &= e^\psi \\
\frac{dh}{d\psi} &= e^\psi = e^{\log(\theta)} = \theta \\
p_\psi(\psi) &= p_\theta\left(h(\psi)\right) \times \left|\frac{dh}{d\psi}\right| = p_\theta(\theta) \times \left|\frac{dh}{d\psi}\right| \\
p_\psi(\psi) &= \frac{b^a}{\Gamma(a)}\theta^{a-1}e^{-b\theta}I(\theta > 0) \times |\theta| = \frac{b^a}{\Gamma(a)}\theta^{a-1}e^{-b\theta}\theta I(\theta > 0) \\
p_\psi(\psi) &\propto \theta^a e^{-b\theta} I(\theta > 0) \\
p_\psi(\psi) &\sim \text{Gamma}(a+1, b)
\end{aligned}
$$

```r
### Plot p(psi) for a=1, b=1
### Define gamma function
gamma <- function (a, b, theta) {
  return (b^a/(factorial(a-1))*theta^(a-1)*exp(-b*theta))
}
### Define range of theta values
theta = seq(0, 1, length.out = 1000)
### Plot posterior gamma function for a=1, b=1
a = 1
b = 1
```

```r
plot(theta, gamma(a+1, b, theta),
     type = "l", ylab = "", xlab = "Theta", main = "Posterior for Part b")
```



Lab Component

```r
### Set random seed to get reproducible results
set.seed(12345)
# input data
# 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)
# store data in data frame
iqData = data.frame(Treatment = c(rep("Spurters", length(x)),
                                  rep("Controls", length(y))),
                                  Gain = c(x, y))
```

Task 1

```
### Code provided for lab 4, task 1
xLimits = seq(min(iqData$Gain) - (min(iqData$Gain) %% 5),
              max(iqData$Gain) + (max(iqData$Gain) %% 5),
              by = 5)

ggplot(data = iqData, aes(x = Gain, fill = Treatment, colour = I("black"))) +
  geom_histogram(position = "dodge", alpha = 0.5, breaks = xLimits, closed = "left")+
  scale_x_continuous(breaks = xLimits,
                     expand = c(0,0))+
  scale_y_continuous(expand = c(0,0),
breaks = seq(0, 10, by = 1))+
  ggtitle("Histogram of Change in IQ Scores") + labs(x = "Change in IQ Score",
                                fill = "Group") +
                          theme(plot.title = element_text(hjust = 0.5))
```

Task 2

```
### Code provided for lab 4, task 2
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)
```

% latex table generated in R 4.0.2 by xtable 1.8-4 package % Fri Sep 11 15:32:43 2020

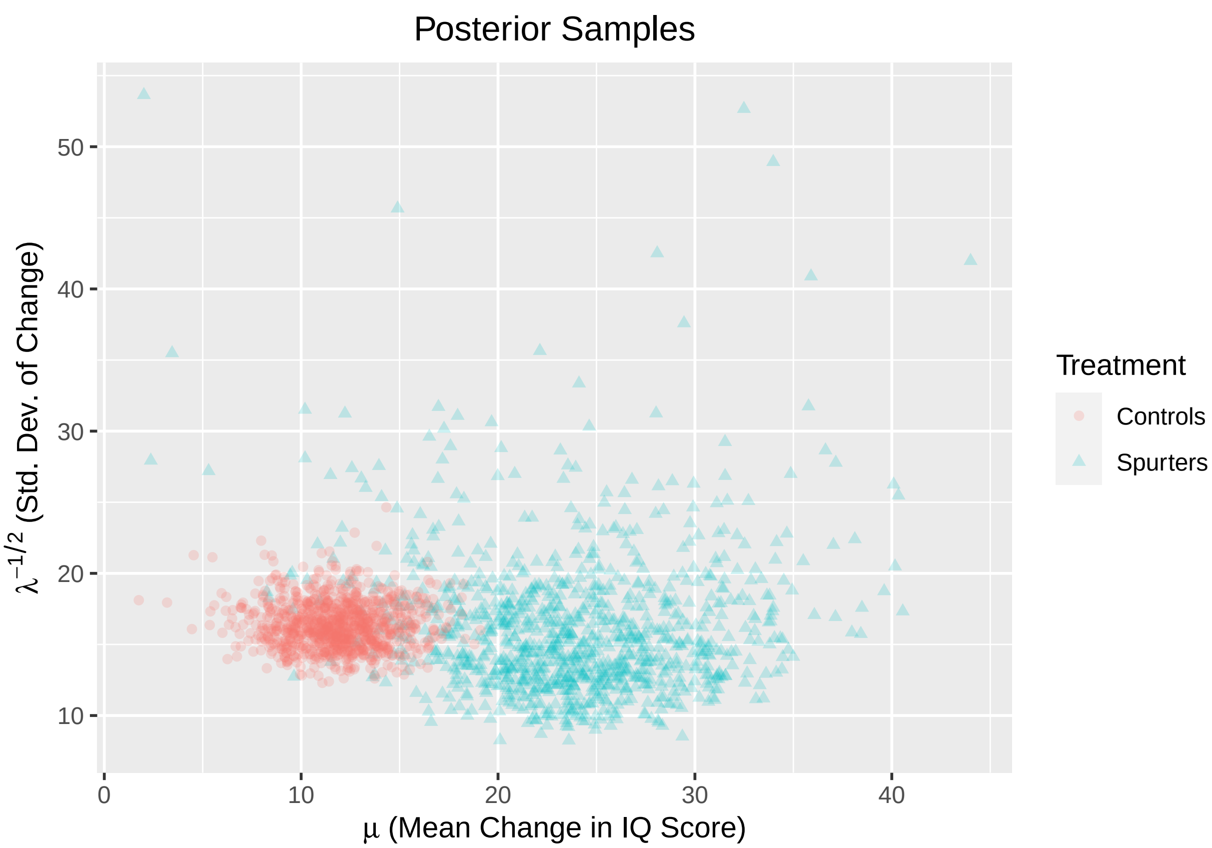
	m	c	a	b
prior	0.00	1.00	0.50	50.00
Spurters Posterior	24.00	8.00	4.00	855.00
Controls Posterior	11.80	49.00	24.50	6343.98

Table 1: Parameters

Task 3

```
### Code provided for lab 4, task 3
# sampling from two posteriors

# Number of posterior simulations
sim = 1000

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

# Following formula from the NormalGamma with
# the update paramaters accounted 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
simDF = data.frame(lambda = c(lambdas, lambdac),
                   mu = c(mus, muc),
```

```r
                      Treatment = rep(c("Spurters", "Controls"),
                                      each = sim))

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

# Plot the simulations
ggplot(data = simDF, aes(x = mu, y = lambda, colour = Treatment, shape = Treatment)) +
  geom_point(alpha = 0.2) +
  labs(x = expression(paste(mu, " (Mean Change in IQ Score)")),
       y = expression(paste(lambda^{-1/2}, " (Std. Dev. of Change)")))  +
  ggtitle("Posterior Samples")+
  theme(plot.title = element_text(hjust = 0.5))
```



Task 4

```r
### Define number of samples
N = 1000000

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

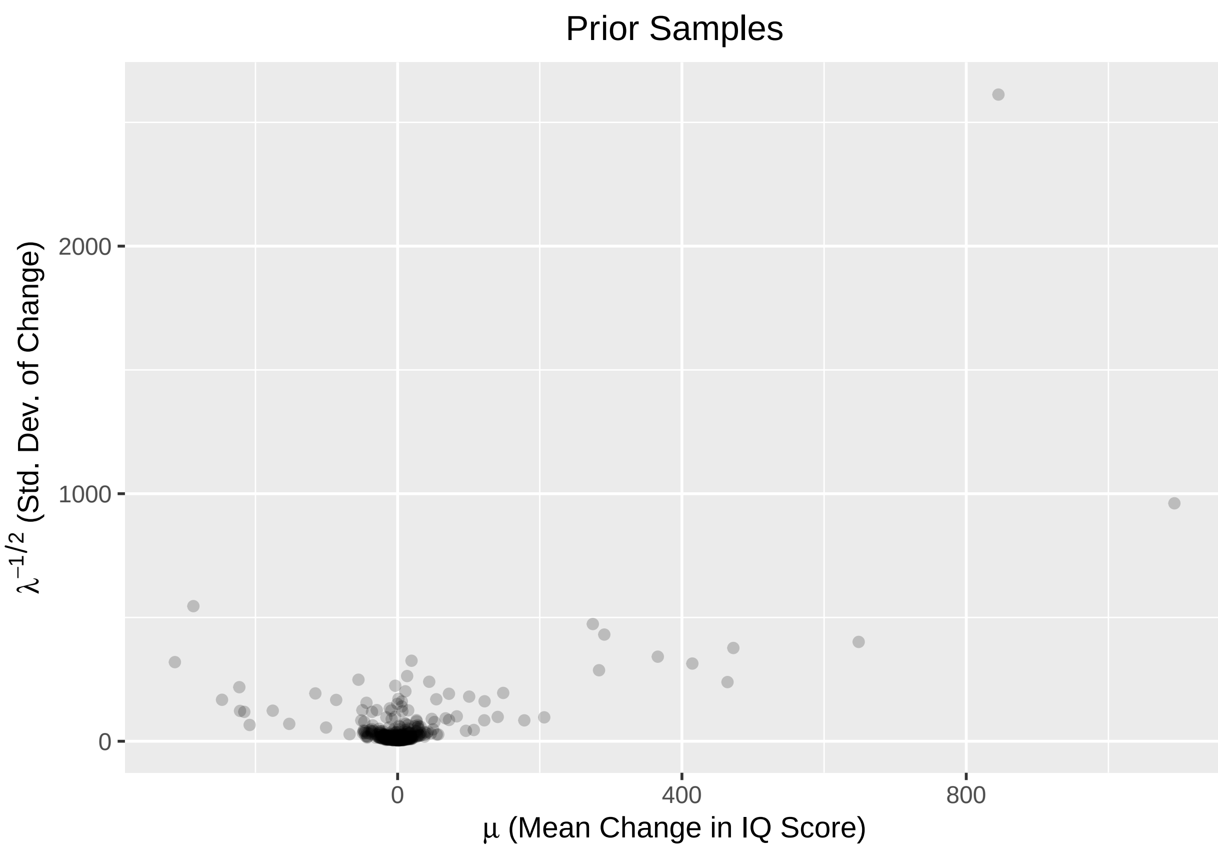
```r
# Following formula from the NormalGamma with
# the update paramaters accounted accounted for below

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

### Calculate and print the probability
probability = sum(mus > muc) / N
probability
```

```
## [1] 0.970841
```

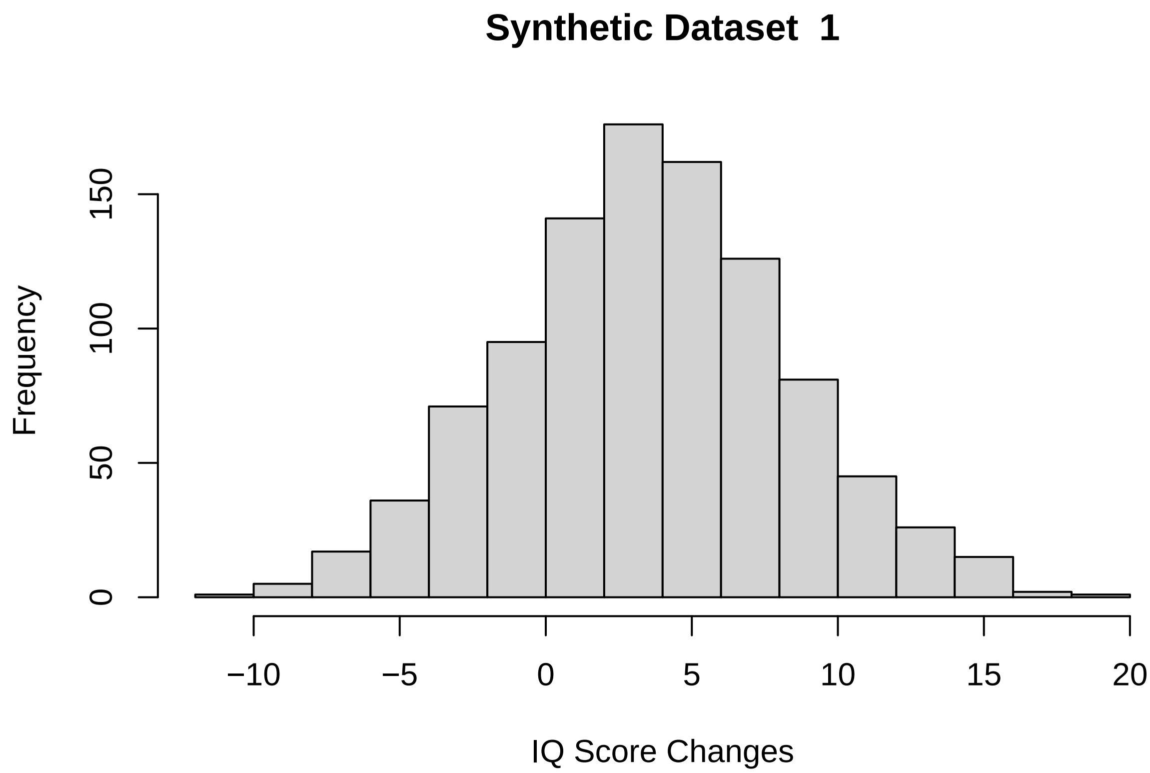
The probability that the mean IQ change of the spurters is greater than the mean IQ change of the controls given the data points we used for the Monte Carlo simulation is 0.970841. This should a good estimate of the true probability that the mean IQ change of the spurters is greater than the mean IQ change of the controls given the original data.

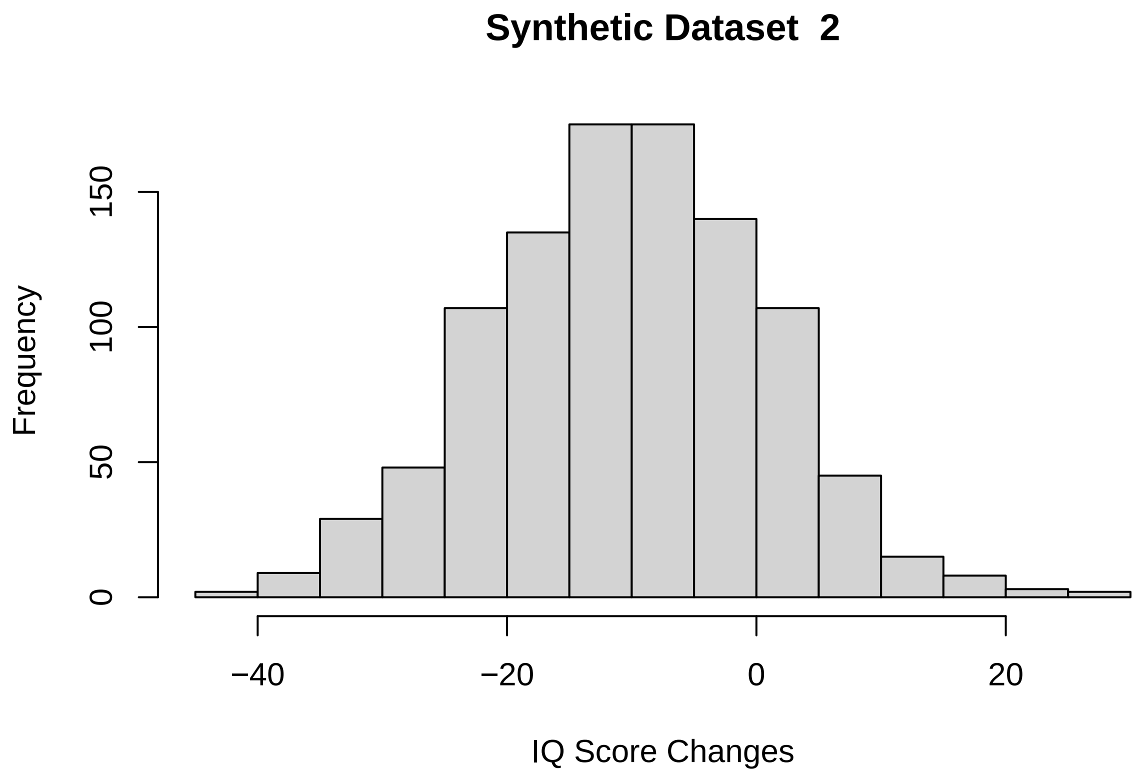
Task 5

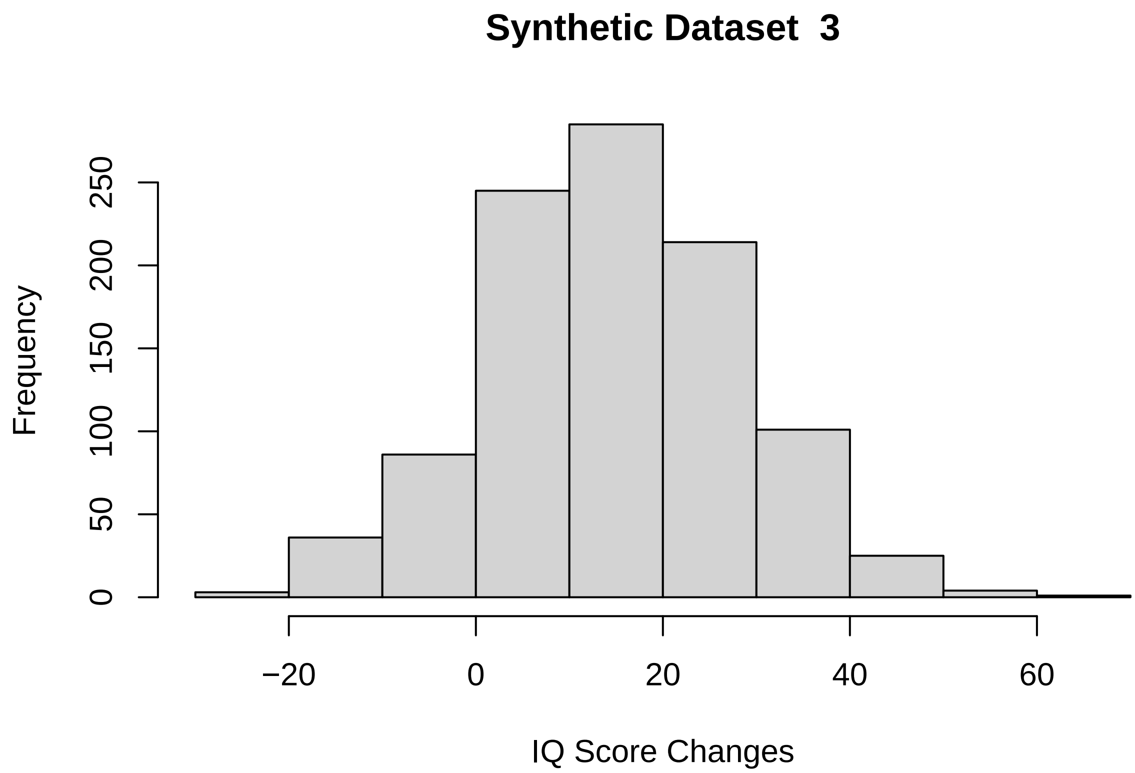
```r
### Based on sample code from OH
### Set a seed for reproducibility
set.seed(45678)
### Sample mu and lambda from the prior
### Define simulation size
sim <- 500
### Initialize vectors
prior_mu <- NULL
prior_lambda <- NULL
### Calculate simulated values
prior_lambda <- rgamma(n = sim, shape = prior$a, rate = prior$b)
prior_mu <- sapply(sqrt(1/(prior$c*prior_lambda)),rnorm, n = 1, mean = prior$m)

### Store into a data frame
priorSim <- data.frame(lambda = c(prior_lambda), mu = c(prior_mu))
priorSim$lambda <- priorSim$lambda^{-0.5}

### Plot prior samples
ggplot(data = priorSim, aes(x = mu, y = lambda)) +
  geom_point(alpha = 0.2) +
  labs(x = expression(paste(mu, " (Mean Change in IQ Score)")),
       y = expression(paste(lambda^{-1/2}, " (Std. Dev. of Change)")))  +
  ggtitle("Prior Samples")+
  theme(plot.title = element_text(hjust = 0.5))
```

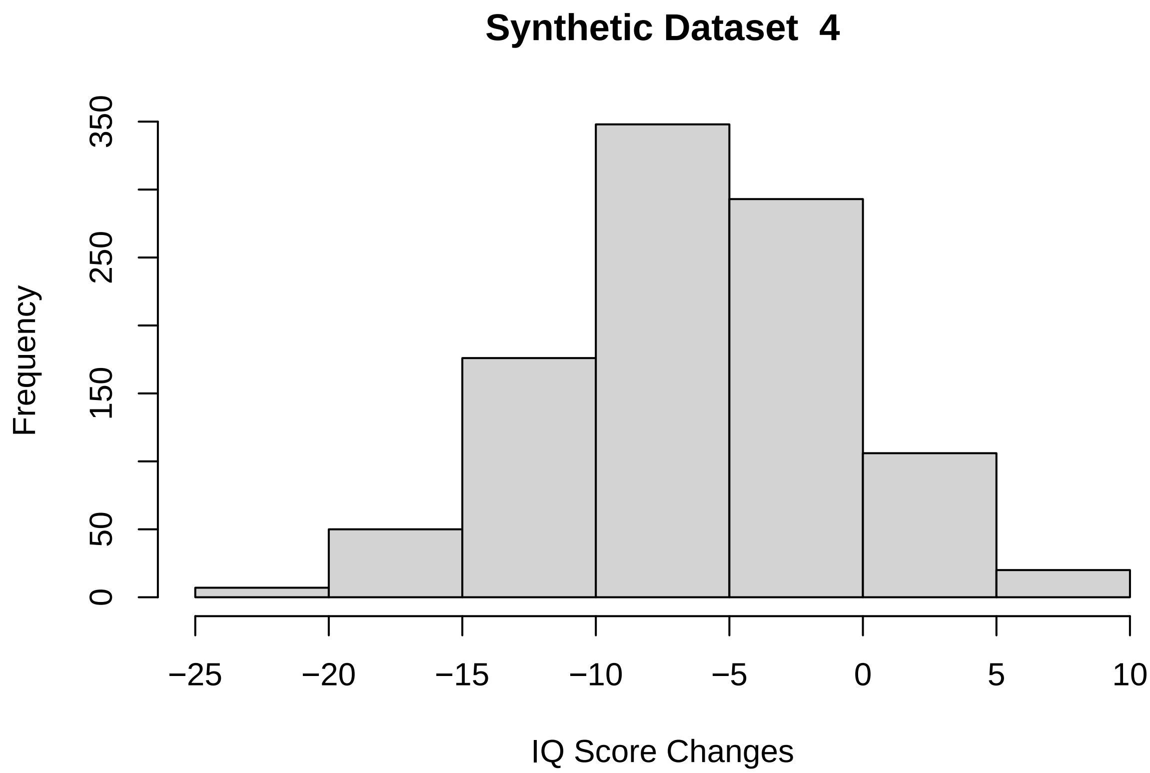


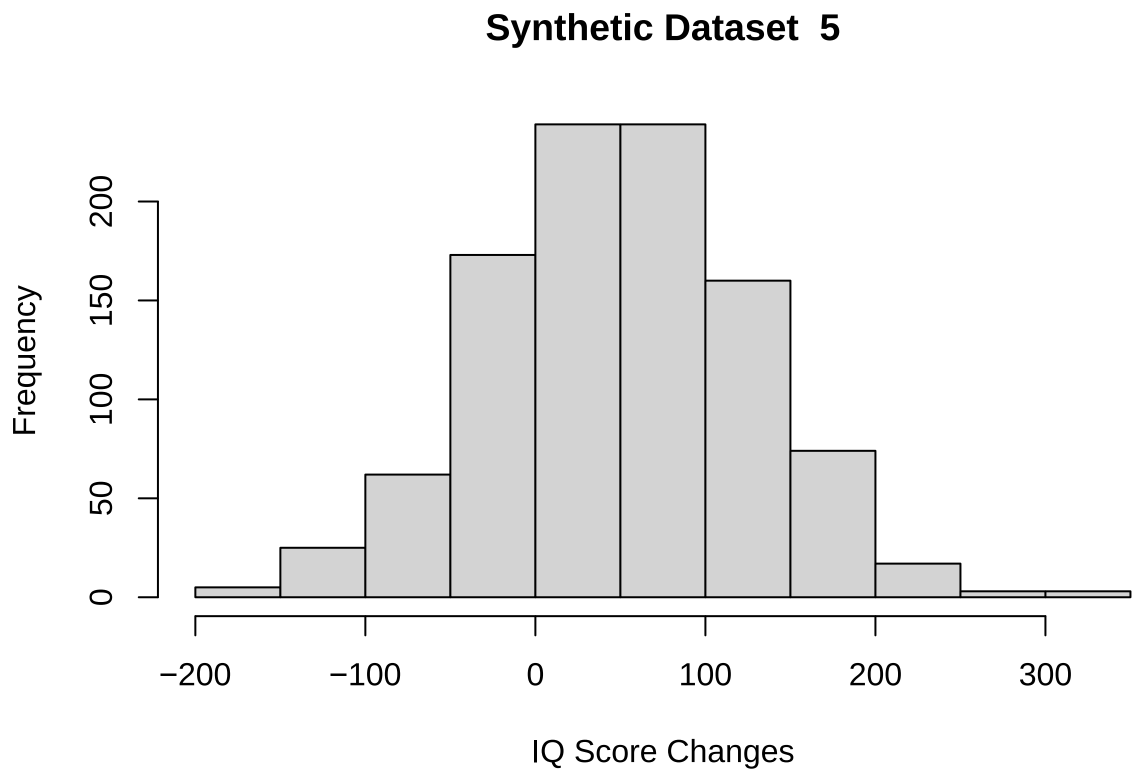

```r
### Second part of the problem
### Set a seed for reproducibility
set.seed(123)
#### This will create 10 synthetic data sets (of sample size 1000)
Simdata <- mapply(rnorm, n=1000, mean = tail(priorSim$mu, n=10),
        sd = tail(priorSim$lambda, n=10))
### Plot the corresponding histograms of these 10 synthetic data sets
### and note the behavior
columns = seq(1, 10)
for (i in columns) {
  hist(Simdata[,i], main = paste("Synthetic Dataset ", i), xlab = "IQ Score Changes")
}
```


Synthetic Dataset 1
Frequency
0
50
100
150
−10
−5
0
5
10
15
20
IQ Score Changes

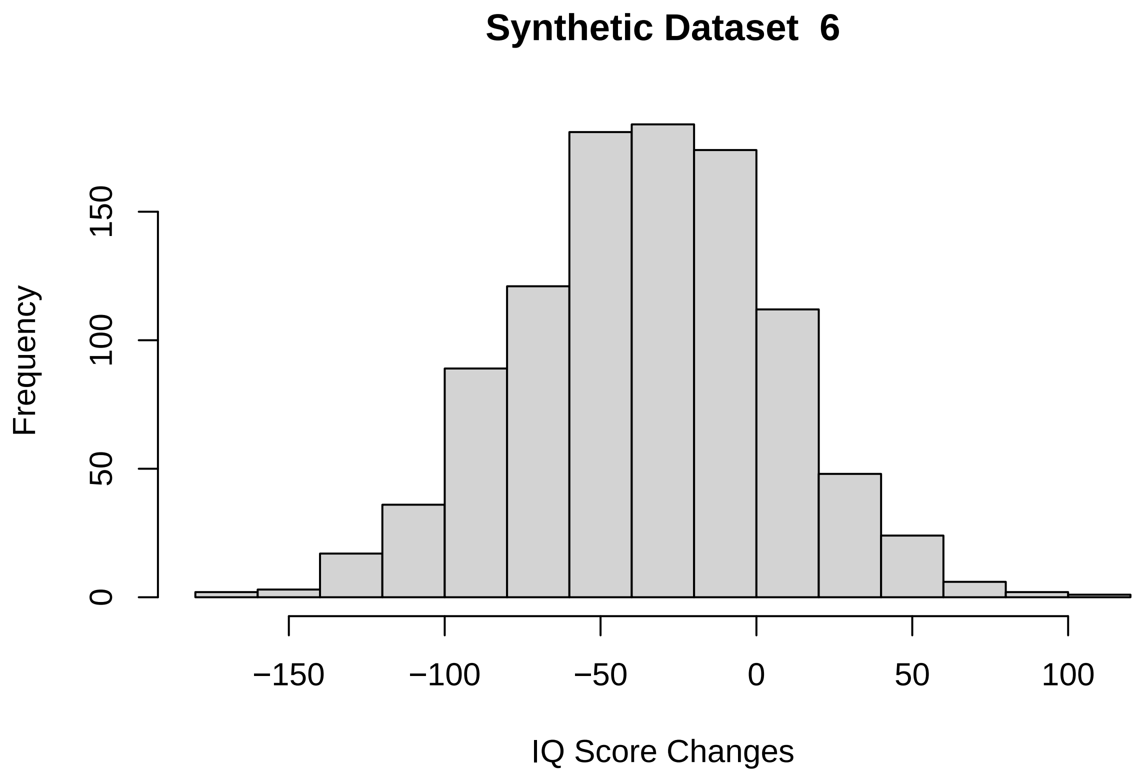

Synthetic Dataset 2
Frequency
0
50
100
150
−40
−20
0
20
IQ Score Changes

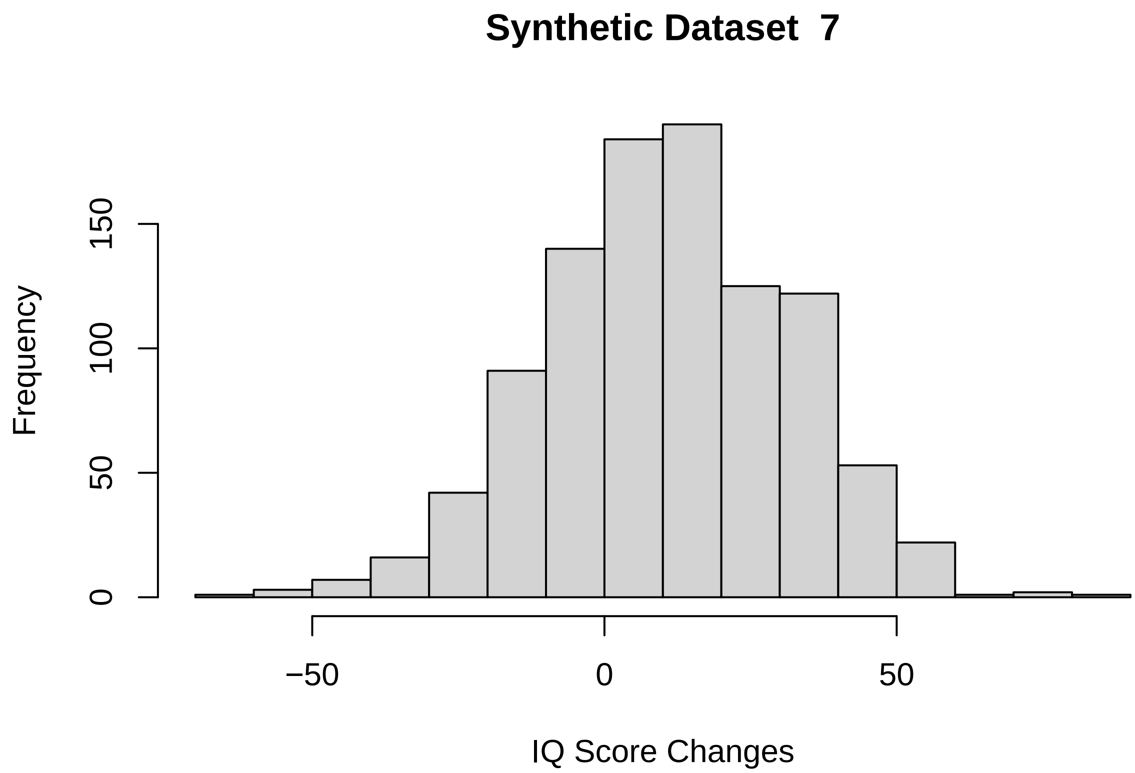

Synthetic Dataset 3
Frequency
0
50
100
150
200
250
−20
0
20
40
60
IQ Score Changes


Synthetic Dataset 4
Frequency
0
50
150
250
350
−25
−20
−15
−10
−5
0
5
10
IQ Score Changes

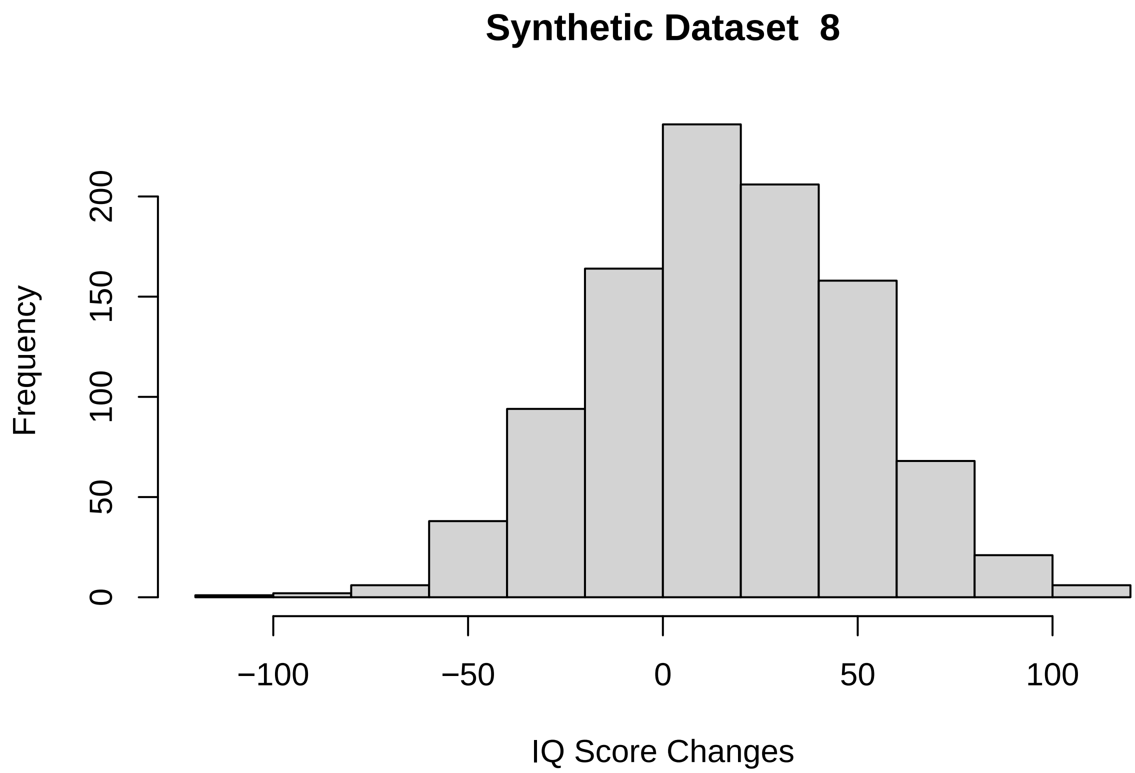


Synthetic Dataset 5
Frequency
0
50
100
150
200
−200
−100
0
100
200
300
IQ Score Changes

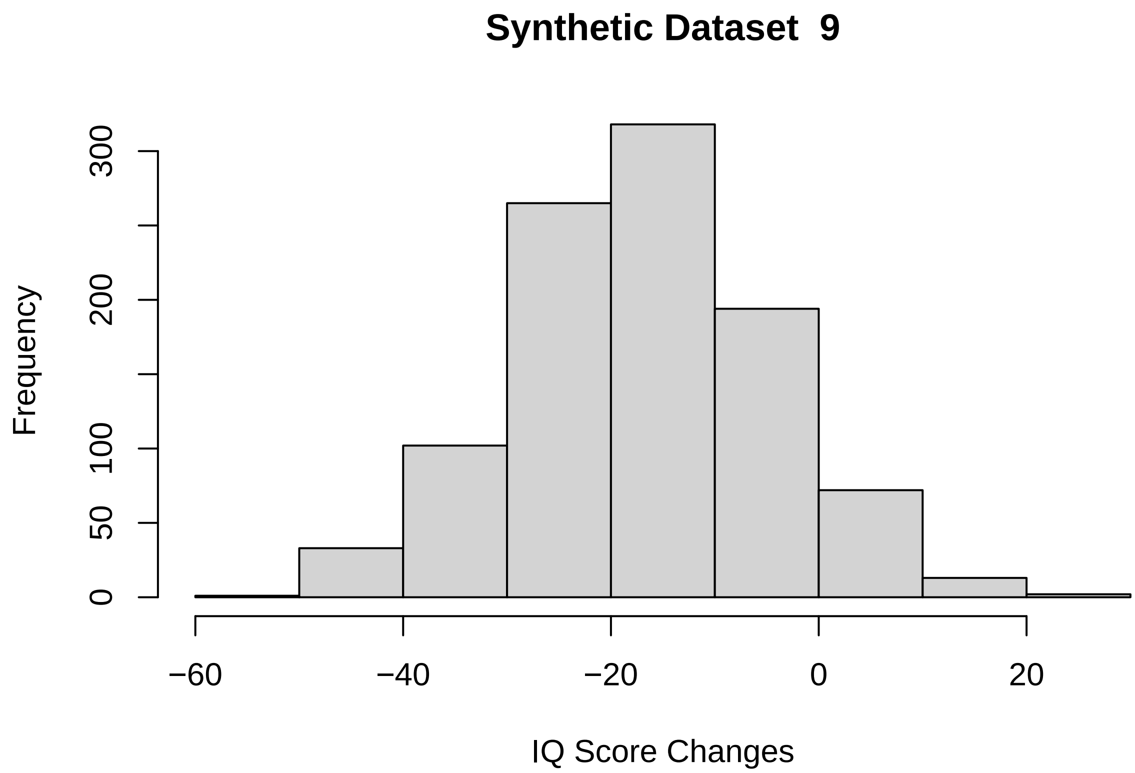

Synthetic Dataset 6
Frequency
0
50
100
150
−150
−100
−50
0
50
100
IQ Score Changes

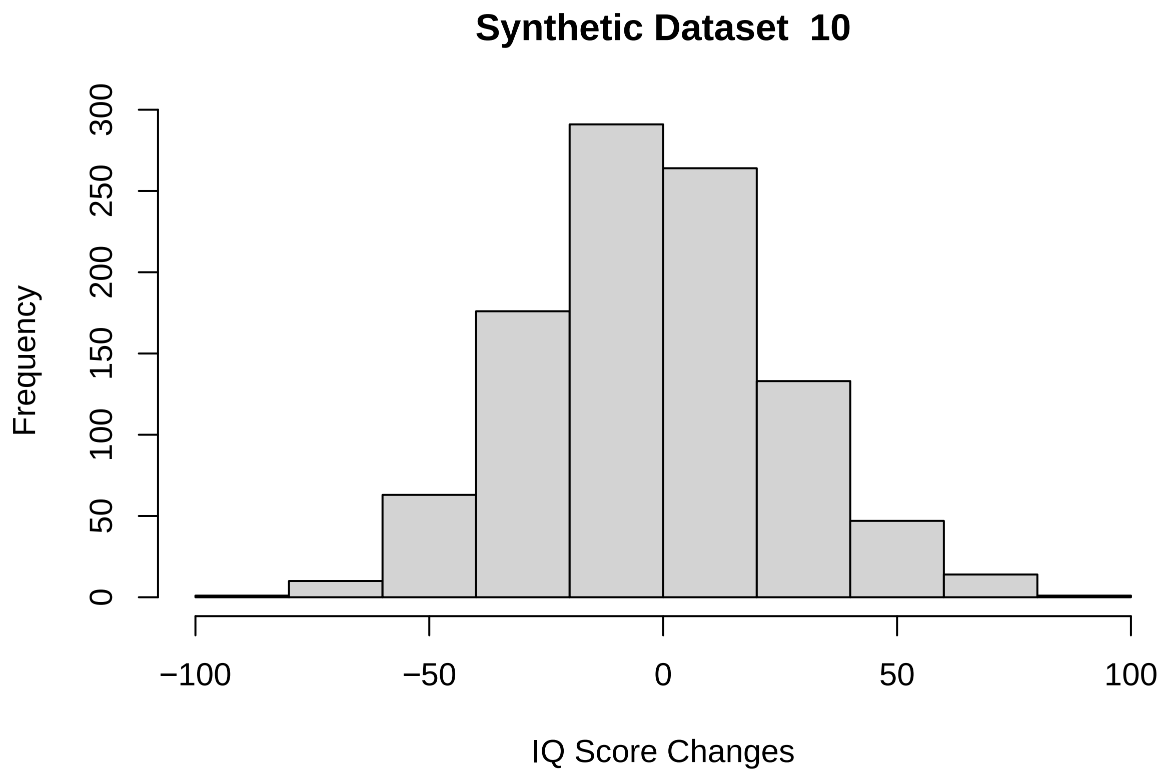


Synthetic Dataset 7
Frequency
0
50
100
150
−50
0
50
IQ Score Changes



Synthetic Dataset 8
Frequency
0
50
100
150
200
−100
−50
0
50
100
IQ Score Changes


Synthetic Dataset 9
Frequency
0
50
100
200
300
−60
−40
−20
0
20
IQ Score Changes



Synthetic Dataset 10
Frequency
0
50
100
150
200
250
300
−100
−50
0
50
100
IQ Score Changes