Class-4-Summary

Jenna G. Tichon

15/10/2019

4. Mixture Models

4.2 Finite mixtures

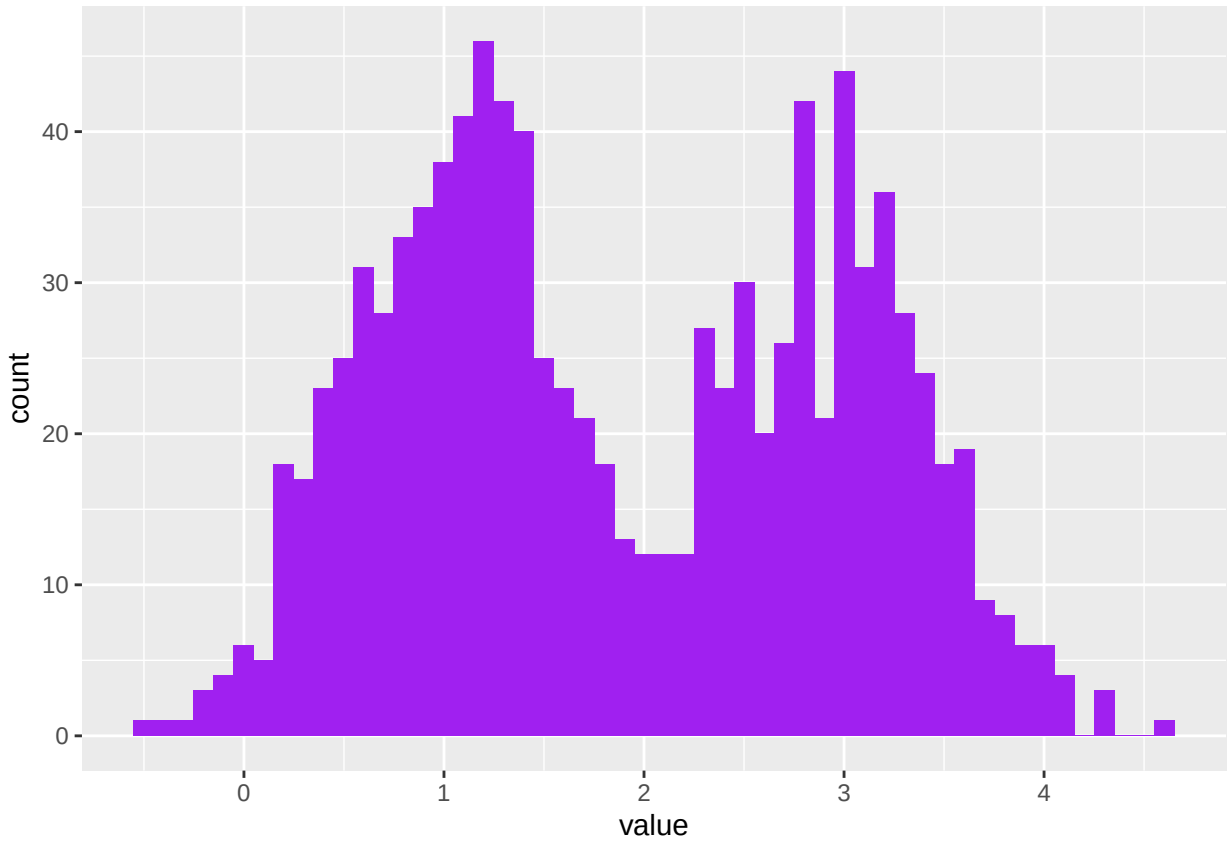
Coin flip experiment

```
# Flip a fair coin 10000 times and store as T or F
coinflips <- (runif(1000,0,1)>0.5)

# Make a summary table
table(coinflips)
```

```
## coinflips
## FALSE  TRUE
##   457   543
```

Coin flip followed by generating from either a $N(1, 0.5)$ or a $N(3, 0.5)$ distribution

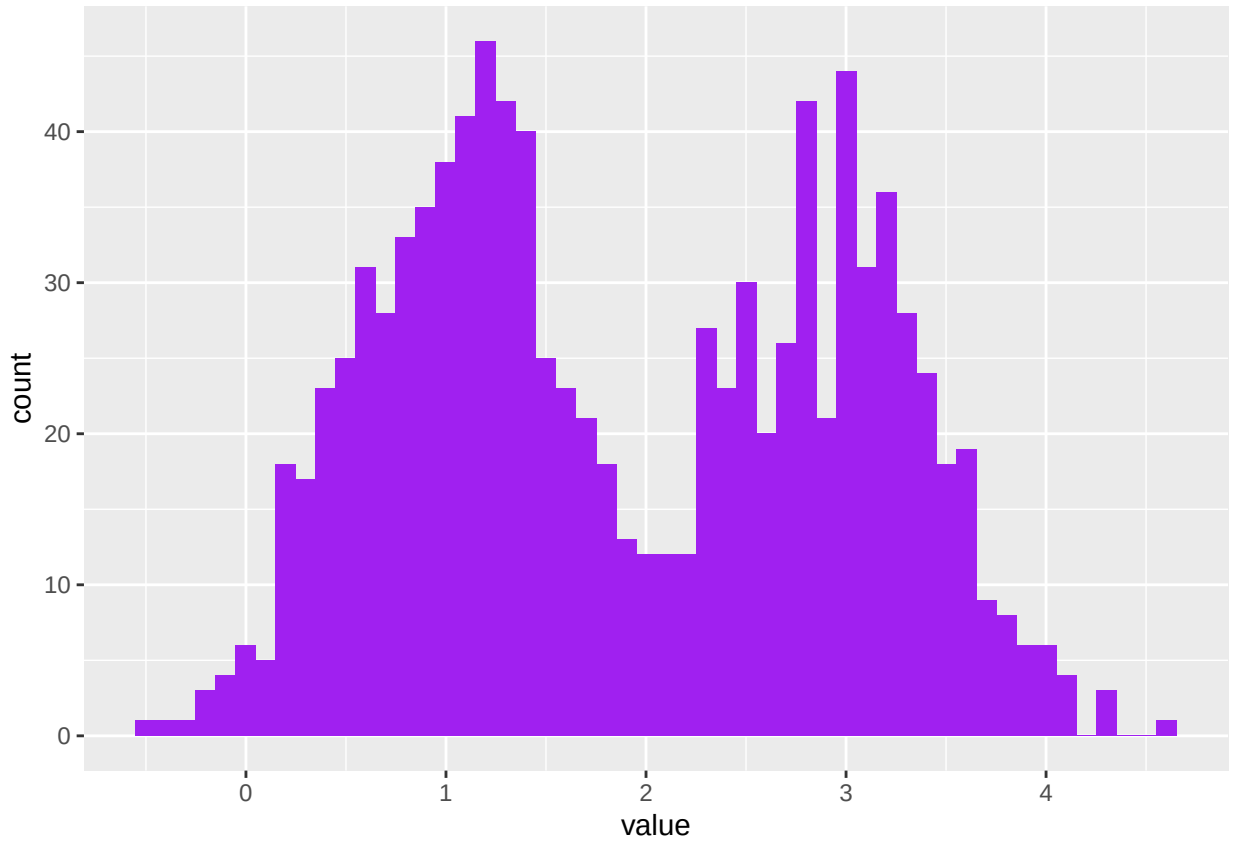
```
#Function to simulate one flip
oneFlip <- function(fl, mean1 = 1, mean2 = 3, sd1 = 0.5, sd2 = 0.5){
  #If heads use distribution 1, else use distribution 2
  if(fl){
    rnorm(1,mean1,sd1)
  } else {
    rnorm(1, mean2, sd2)
  }
}

#Make a histogram using the 10000 coinflips in coinflips
fairmix = vapply(coinflips, oneFlip, numeric(1))
ggplot(tibble(value = fairmix), aes(x = value)) +
  geom_histogram(fill = "purple", binwidth = 0.1)
```



Q 4.1

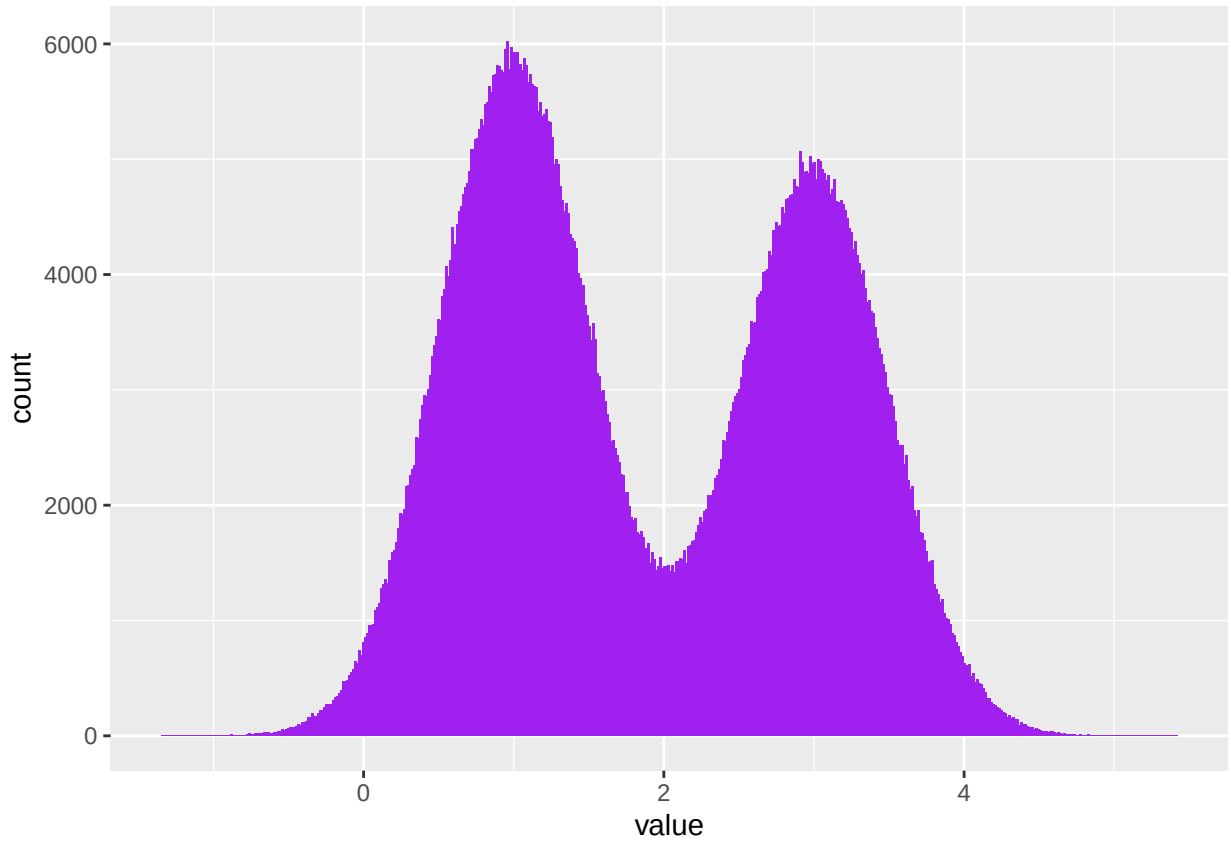
! Typo ! The standard deviation suddenly changes

```
means = c(1, 3)
sds   = c(0.5, 0.5)
values = rnorm(length(coinflips),
          mean = ifelse(coinflips, means[1], means[2]),
          sd   = ifelse(coinflips, sds[1],   sds[2]))
ggplot(tibble(value = fairmix), aes(x = value)) +
  geom_histogram(fill = "purple", binwidth = 0.1)
```



Q 4.2

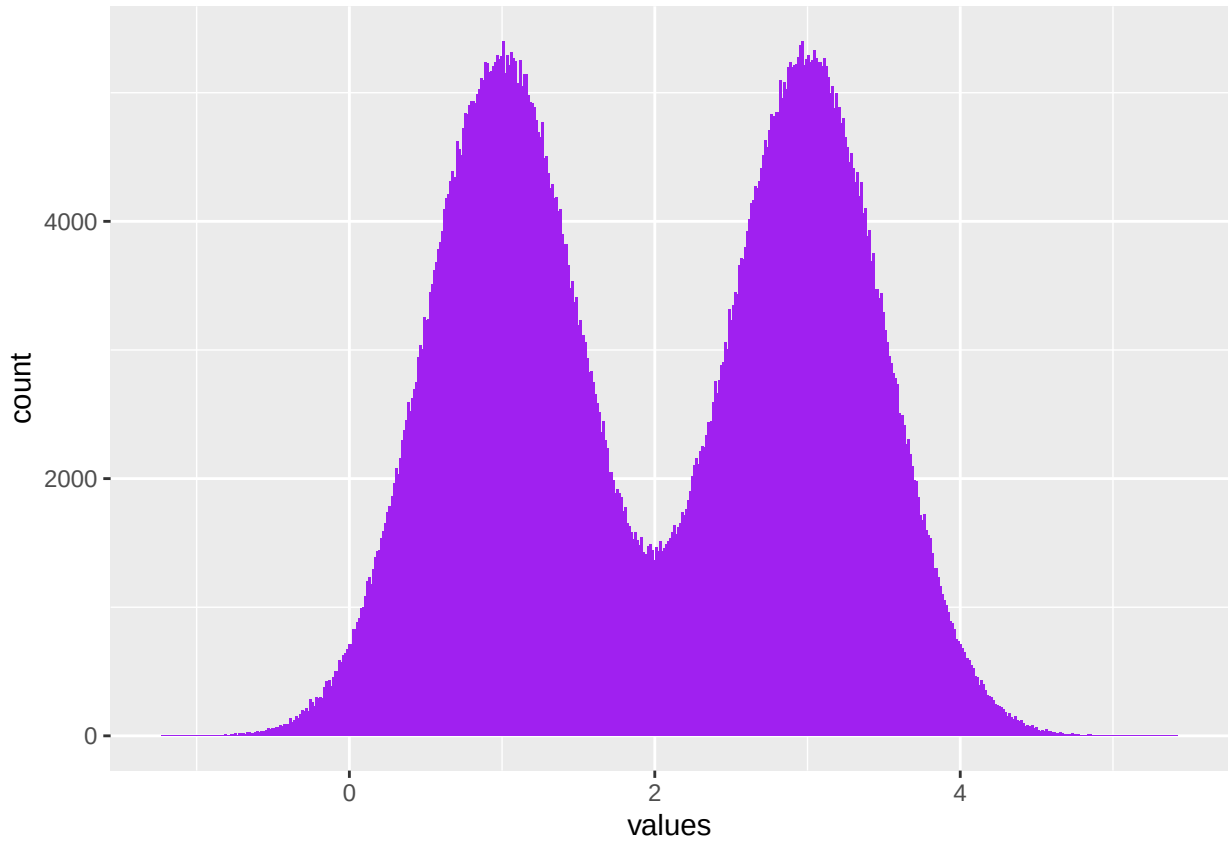
```
means = c(1, 3)
sds   = c(0.5, 0.5)
values = rnorm(1000000,
          mean = ifelse(coinflips, means[1], means[2]),
          sd   = ifelse(coinflips, sds[1],   sds[2]))
ggplot(tibble(value = values), aes(x = value)) +
  geom_histogram(fill = "purple", bins = 500)
```



Data becomes less sparse and looks more continuous

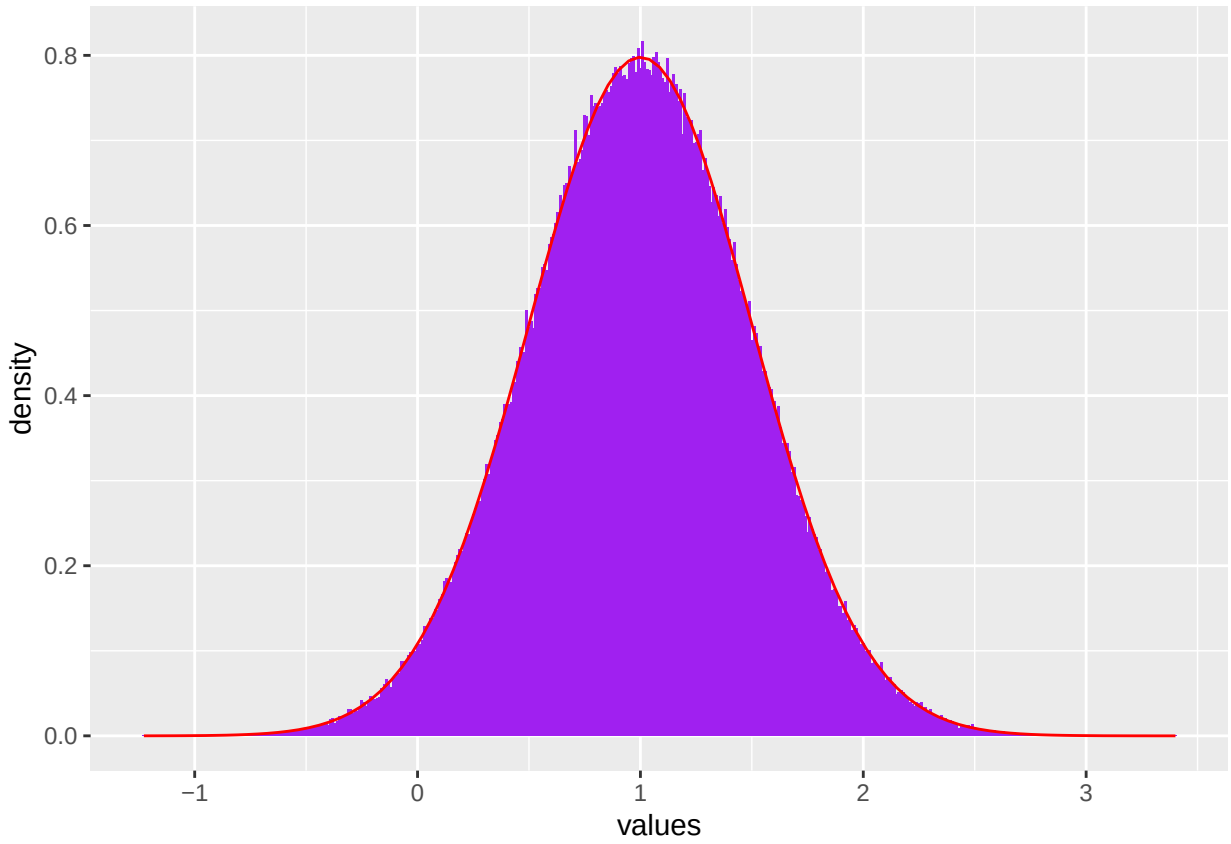
Text solution:

```
fair = tibble(
  coinflips = (runif(1e6) > 0.5),
  values = rnorm(length(coinflips),
                 mean = ifelse(coinflips, means[1], means[2]),
                 sd   = ifelse(coinflips, sds[1],   sds[2])))
ggplot(fair, aes(x = values)) +
     geom_histogram(fill = "purple", bins = 500)
```



Q 4.3

```
#Take the tibble coinflips and sort out just those marked as fair = TRUE
#Make a histogram of those coin flips and overlay the density of normal for the fair coin flips
ggplot(dplyr::filter(fair, coinflips), aes(x = values)) +
  geom_histogram(aes( y = ..density..), fill = "purple",
                 binwidth = 0.01) +
  stat_function(fun = dnorm,
                args = list(mean = means[1], sd = sds[1]), color = "red")
```

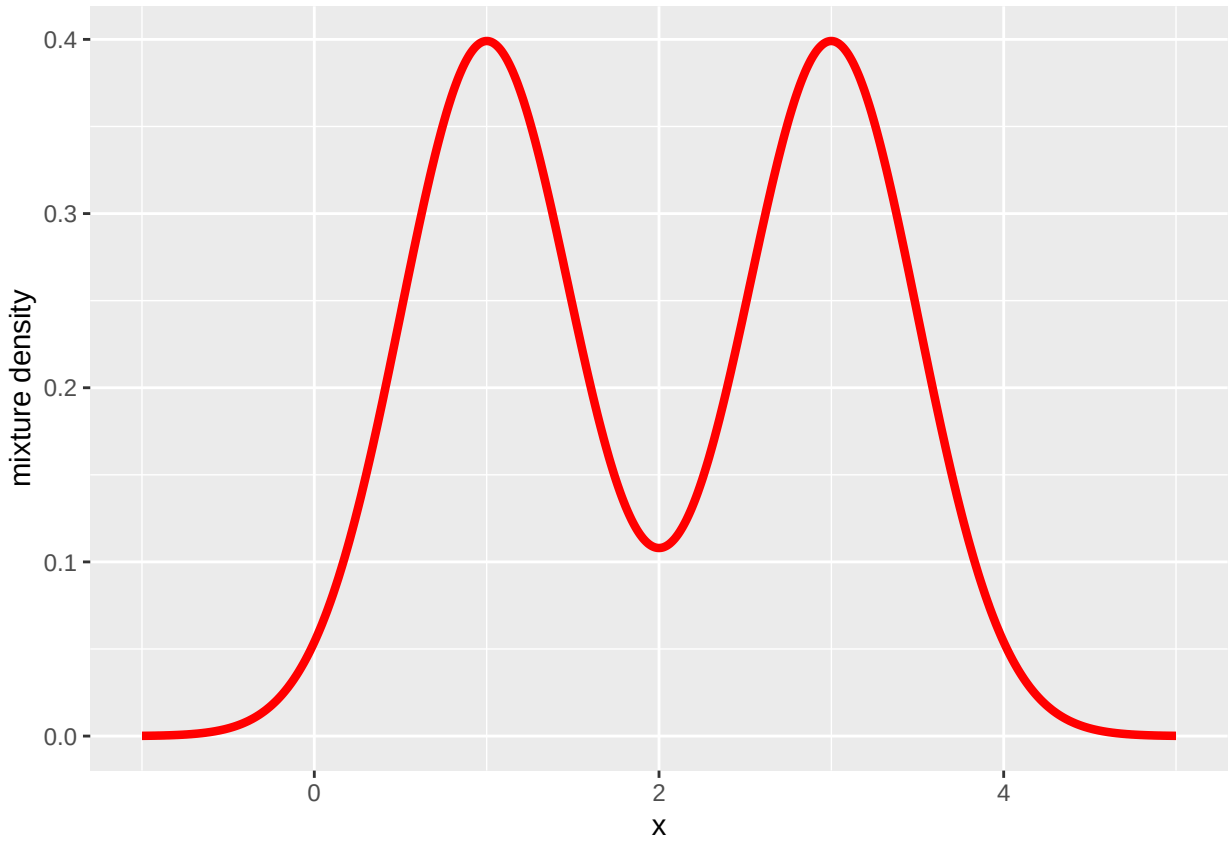


The density curve is

$$f(x) = \frac{1}{2}\phi_1(x) + \frac{1}{2}\phi_2(x)$$

Plotting the density:

```
fairtheory = tibble(
  x = seq(-1, 5, length.out = 1000),
  f = 0.5 * dnorm(x, mean = means[1], sd = sds[1]) +
      0.5 * dnorm(x, mean = means[2], sd = sds[2]))
ggplot(fairtheory, aes(x = x, y = f)) +
  geom_line(color = "red", size = 1.5) + ylab("mixture density")
```



4.2.2 Discovering the hidden class labels

u is unobserved group label. y is observed data from two unknown groups. The joint density of y and u is

$$f_\theta(y, u) = f_\theta(y|u) f_\theta(u)$$

In this last example $\theta = (\mu_1, \mu_2, \sigma_1, \sigma_2, \lambda)$ where λ is the mixture fraction $\lambda = 0.5$.

Experiment: - With prob π flip coin 1 with $p_1 = 0.125$, with probability $1 - \pi$ flip coin 2 with $p_2 = 0.25$ - Toss coin twice - Record number of heads K

```
#Function to simulate the experiment once
kflips<-function(p1=0.125, p2=0.25, pi=(1/8)){
  coin<-rbinom(1,1,pi)
  if(coin == 1){
    rbinom(1,2,p1)
  } else{
    rbinom(1,2,p2)
  }
}

#Apply this 100 times and make a contingency table
k<-replicate(100, kflips())
table(k)
```

```
## k
##  0  1  2
```

```
## 63 32  5
```

Redo with pi=0.25

```
k<-replicate(100, kflips(pi=0.25))
table(k)
```

```
## k
##  0  1  2
## 57 41  2
```