

Fitting Normal Distribution

데이터테크전공 20173204 곽명빈

2020-10-01

Data

From Stigler's

```
knitr::include_graphics("../pics/quetelet_soldiers.png")
```

MESURES de la POITRINE.	NOMBRE d'hommes.	NOMBRE PROPORTIONNEL.	PROBABILITÉ d'après L'OBSERVATION.	RANG dans LA TABLE.	RANG d'après le CALCUL.	PROBABILITÉ d'après LA TABLE.	NOMBRE d'OBSERVATIONS calculé.
Pouces.							
33	3	3	0,5000			0,5000	7
34	18	31	0,4905	52	50	0,4905	29
35	81	141	0,4964	42,5	42,5	0,4964	110
36	185	322	0,4825	35,5	34,5	0,4854	325
37	420	732	0,4501	26,0	26,5	0,4531	732
38	749	1305	0,3769	18,0	18,5	0,3799	1355
39	1075	1867	0,2464	10,5	10,5	0,2466	1838
			0,0597	2,5	2,5	0,0628	
40	1079	1882	0,1285	5,5	5,5	0,1359	1987
41	934	1628	0,2915	15	15,5	0,3034	1675
42	658	1148	0,4061	21	21,5	0,4150	1096
43	370	645	0,4706	30	29,5	0,4690	560
44	92	160	0,4866	35	37,5	0,4911	221
45	50	87	0,4955	41	45,5	0,4980	69
46	21	38	0,4991	49,5	55,5	0,4996	16
47	4	7	0,4998	50	61,8	0,4999	5
48	1	2	0,5000			0,5000	1
	5758	1,0000					1,0000

Frequency Table

- 케틀레가 작성한 스코틀랜드 군인 5738명의 가슴둘레(인치) 분포표를 옮기면

```
chest <- 33:48
freq <- c(3, 18, 81, 185, 420, 749, 1073, 1079, 934, 658, 370, 92, 50, 21, 4, 1)
data.frame(chest, freq)
```

```
##      chest freq
## 1      33    3
## 2      34   18
## 3      35   81
## 4      36  185
## 5      37  420
## 6      38  749
## 7      39 1073
## 8      40 1079
## 9      41  934
## 10     42  658
## 11     43  370
## 12     44   92
## 13     45   50
## 14     46   21
## 15     47    4
## 16     48    1
```

```
data.frame(Chest = chest, Freq = freq)
```

```
##      Chest Freq
## 1      33    3
## 2      34   18
## 3      35   81
## 4      36  185
## 5      37  420
## 6      38  749
## 7      39 1073
## 8      40 1079
## 9      41  934
## 10     42  658
## 11     43  370
## 12     44   92
## 13     45   50
## 14     46   21
## 15     47    4
## 16     48    1
```

```
chest_df <- data.frame(Chest = chest, Freq = freq)
chest_df
```

```
##      Chest Freq
## 1      33    3
## 2      34   18
## 3      35   81
## 4      36  185
## 5      37  420
## 6      38  749
## 7      39 1073
## 8      40 1079
## 9      41  934
## 10     42  658
## 11     43  370
## 12     44   92
## 13     45   50
## 14     46   21
## 15     47    4
## 16     48    1
```

```
str(chest_df)
```

```
## 'data.frame':    16 obs. of  2 variables:
##  $ Chest: int  33 34 35 36 37 38 39 40 41 42 ...
##  $ Freq : num  3 18 81 185 420 ...
```

Extract Parts of an Object

```
chest_df$Freq
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
chest_df %>%
  .$Freq
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
str(chest_df$Freq)
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df[, 2]
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
chest_df %>%
  `[`, 2)
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
str(chest_df[, 2])
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df[, "Freq"]
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
chest_df %>%
  `[`(, "Freq")
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
str(chest_df[, "Freq"])
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df["Freq"]
```

```
##      Freq
## 1      3
## 2     18
## 3     81
## 4    185
## 5    420
## 6    749
## 7   1073
## 8   1079
## 9    934
## 10   658
## 11   370
## 12    92
## 13    50
## 14    21
## 15     4
## 16     1
```

```
chest_df %>%
  `[`("Freq")
```

```
##      Freq
## 1      3
## 2     18
## 3     81
## 4    185
## 5    420
## 6    749
## 7   1073
## 8   1079
## 9    934
## 10   658
## 11   370
## 12    92
## 13    50
## 14    21
## 15     4
## 16     1
```

```
str(chest_df["Freq"])
```

```
## 'data.frame':   16 obs. of  1 variable:
##  $ Freq: num  3 18 81 185 420 ...
```

```
chest_df["Freq"]$Freq
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
chest_df %>%
  `[("Freq") %>%
  .$Freq
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
str(chest_df["Freq"]$Freq)
```

```
##  num [1:16] 3 18 81 185 420 ...
```

```
chest_df["Freq"][[1]]
```

```
## [1]  3  18  81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16]  1
```

```
chest_df %>%
  `[("Freq") %>%
  `[[` (1)
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
# `[`(, 1)
# `[`(1)
str(chest_df["Freq"][[1]])
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df[2]
```

```
##      Freq
## 1      3
## 2     18
## 3     81
## 4    185
## 5    420
## 6    749
## 7   1073
## 8   1079
## 9    934
## 10   658
## 11   370
## 12    92
## 13    50
## 14    21
## 15     4
## 16     1
```

```
chest_df %>%
  `[`(2)
```

```
##      Freq
## 1      3
## 2     18
## 3     81
## 4    185
## 5    420
## 6    749
## 7   1073
## 8   1079
## 9    934
## 10   658
## 11   370
## 12    92
## 13    50
## 14    21
## 15     4
## 16     1
```

```
str(chest_df[2])
```

```
## 'data.frame': 16 obs. of 1 variable:
## $ Freq: num 3 18 81 185 420 ...
```

```
chest_df[2]$Freq
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
chest_df %>%
  `[`(2) %>%
  .$Freq
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
str(chest_df[2]$Freq)
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df[2][[1]]
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
chest_df %>%
  `[`(2) %>%
  `[`(1)
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
str(chest_df[2][[1]])
```

```
## num [1:16] 3 18 81 185 420 ...
```

```
chest_df[[2]]
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
chest_df %>%
  `[`(2)
```

```
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
```

```
str(chest_df[[2]])
```

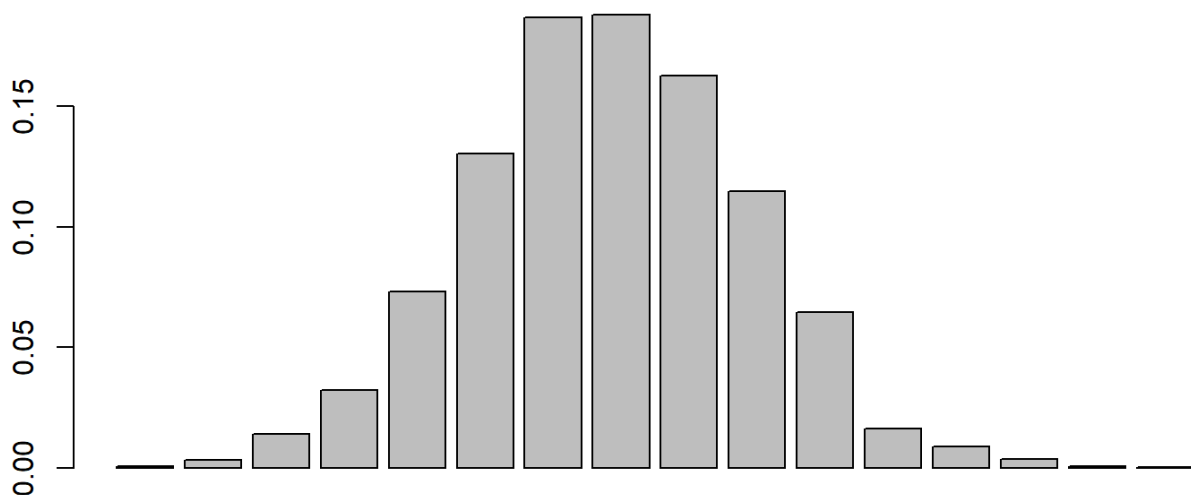
```
## num [1:16] 3 18 81 185 420 ...
```

- 33인치인 사람이 3명, 34인치인 사람이 18명 등으로 기록되어 있으나 이는 구간의 가운데로 이해하여야 함.

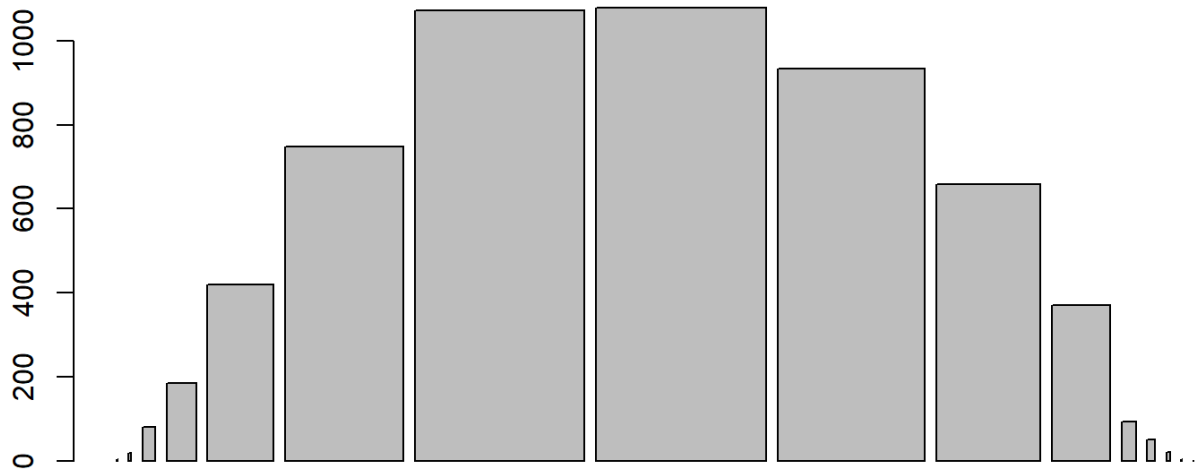
Probability Histogram

- `barplot(height, ...)` 은 기본적으로 `height` 만 주어진다면 그릴 수 있음. 확률 히스토그램의 기둥 면적의 합은 1이므로, 각 기둥의 높이는 각 계급의 dot수를 전체 dot수, 5738명으로 나눠준 값임.

```
total <- sum(chest_df$Freq)
barplot(chest_df$Freq / total)
chest_df$Freq %>%
  `/(., sum(.))` %>%
  barplot
```

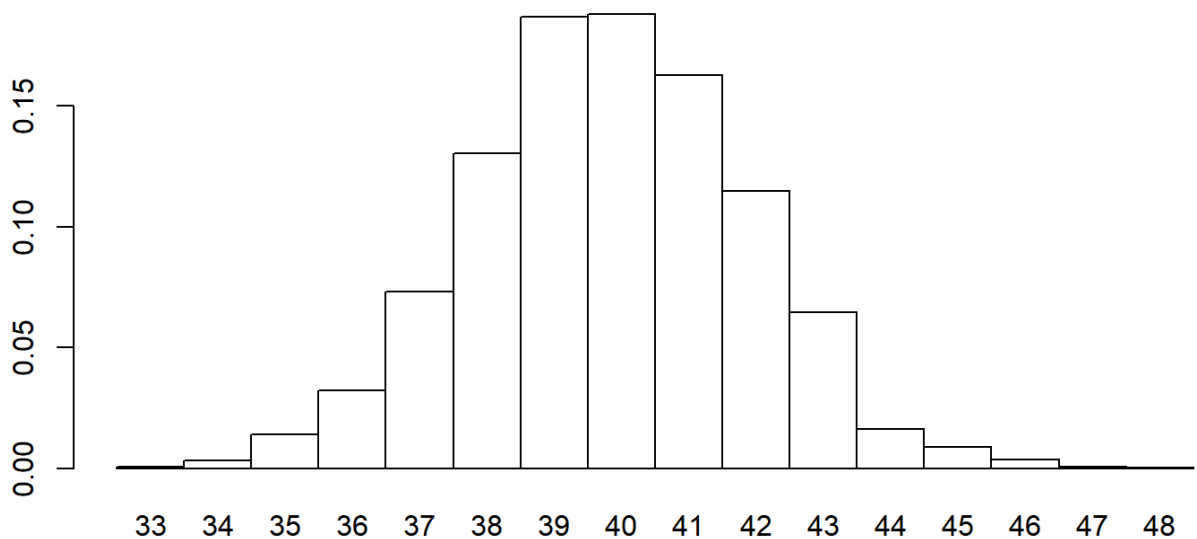


```
chest_df$Freq %>%
  prop.table %>%
  #> R 4.0.0 부터는 proportions 사용 가능
  # proportions %>%
  barplot
#> 조심! 다음 두 표현은 원하는 그림이 나오지 않음.
chest_df$Freq %>%
  barplot(. / sum(.))
chest_df$Freq %>%
  barplot(`/(., sum(.))`)
```

- 각 막대의 이름은 계급을 나타내는 가슴둘레 값으로 표현할 수 있고, 막대 간의 사이를 띄우지 않으며, 디폴트 값으로 주어진 회색 보다는 차라리 백색이 나으므로 이를 설정해 주면,

```
barplot(chest_df$Freq/total,
        names.arg = 33:48,
        space = 0,
        col = "white")
chest_df$Freq %>%
  `/\`(. , sum(.)) %>%
  barplot(names.arg = 33:48,
          space = 0,
          col = "white")
```



- 확률 히스토그램의 정의에 따라 이 막대들의 면적을 합하면 1이 됨에 유의.

Summary statistics and SD

- 33인치가 3명, 34인치가 18명 등을 한 줄의 긴 벡터로 나타내어야 평균과 표준편차를 쉽게 계산할 수 있으므로 long format으로 바꾸면,

```
chest_vec <- rep(chest_df$Chest, chest_df$Freq)
chest_vec <- chest_df %$%
  rep(.$Chest, .$Freq)
str(chest_vec)
```

```
## int [1:5738] 33 33 33 34 34 34 34 34 34 34 ...
```

```
rep()
```

```
rep(1:3, times = 3)
```

```
## [1] 1 2 3 1 2 3 1 2 3
```

```
rep(1:3, each = 3)
```

```
## [1] 1 1 1 2 2 2 3 3 3
```

```
rep(1:3, 1:3)
```

```
## [1] 1 2 2 3 3 3
```

- chest_vec 을 이용하여 기초통계와 표준편차를 계산하면,

```
summary(chest_vec)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    33.00   38.00   40.00   39.83   41.00   48.00
```

```
sd(chest_vec)
```

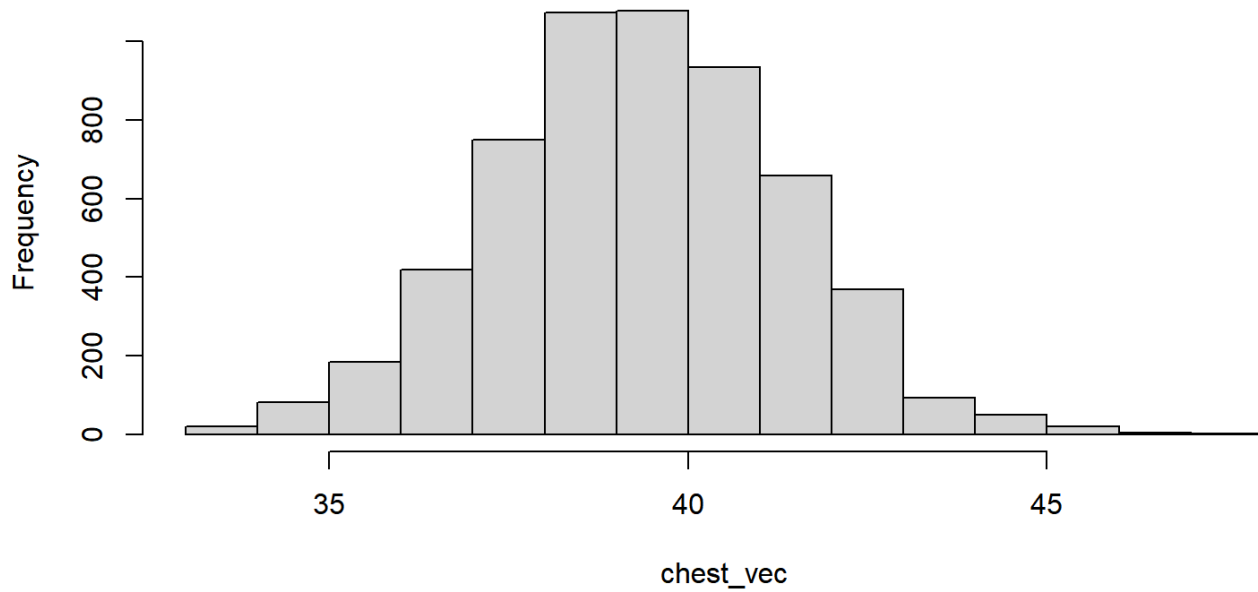
```
## [1] 2.049616
```

Histogram

- 히스토그램을 직관적으로 그려보면 y 축은 dots가 기본값임을 알 수 있음.

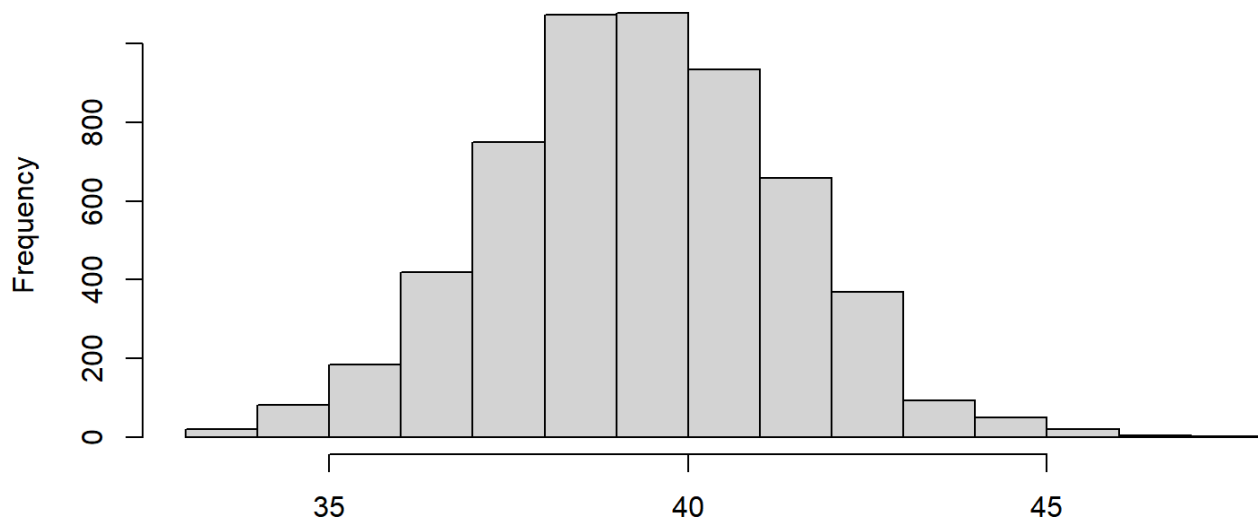
```
hist(chest_vec)
```

Histogram of chest_vec



```
chest_vec %>%
  hist
```

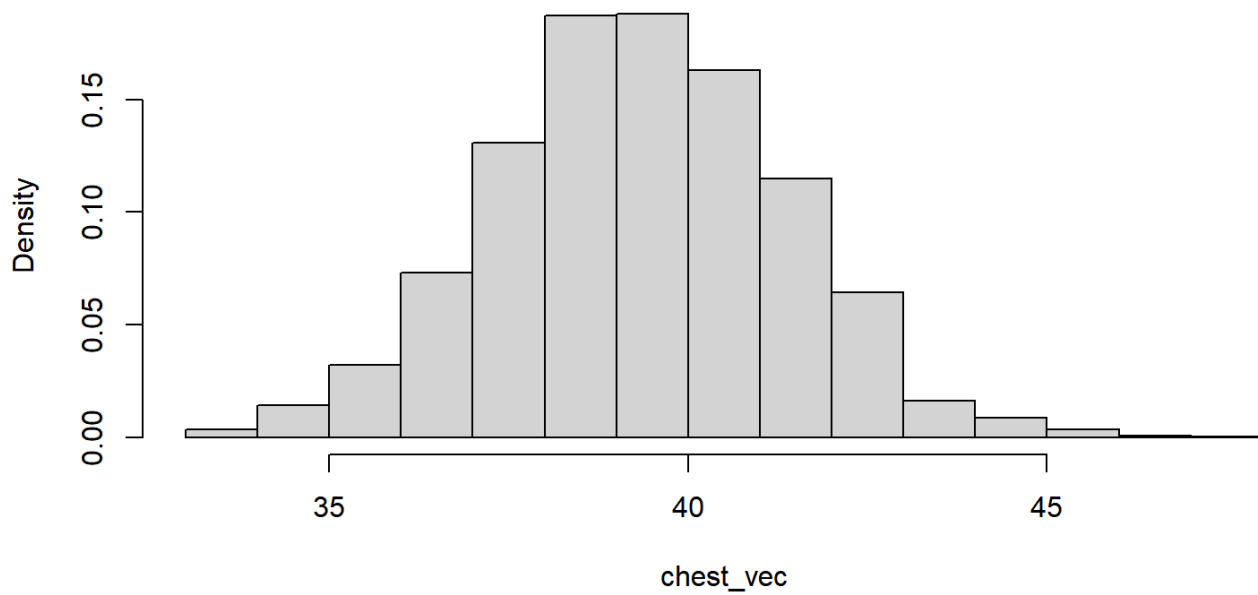
Histogram of .



- 정규분포와 비교하기 위해서 y 축을 확률로 나타내려면

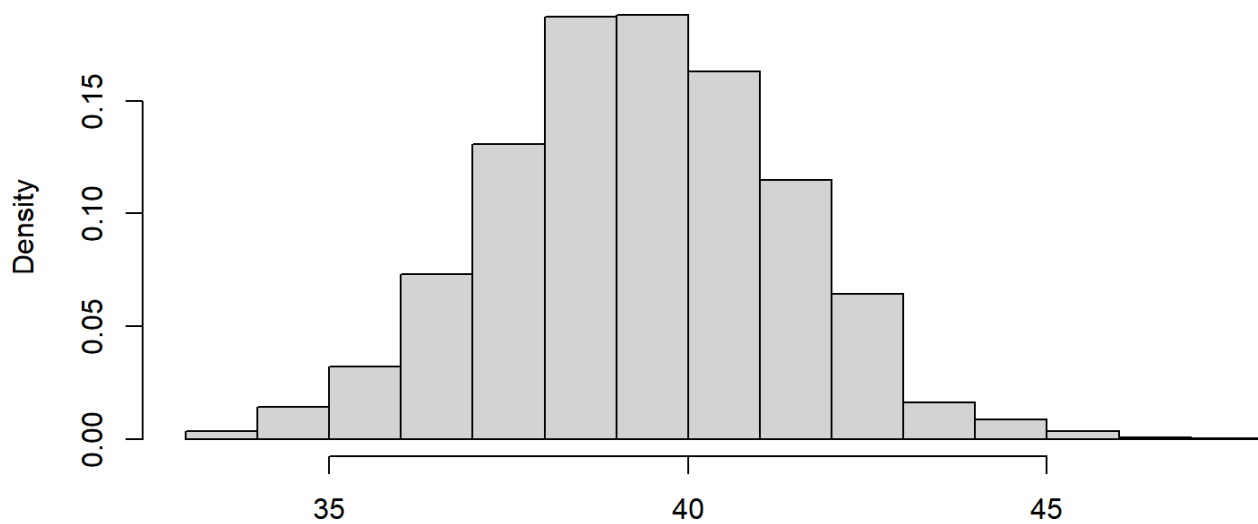
```
hist(chest_vec,
     probability = TRUE)
```

Histogram of chest_vec



```
chest_vec %>%
  hist(probability = TRUE)
```

Histogram of .



Inside the histogram

- 실제로 이 히스토그램을 그리는 데 계산된 값들은?

```
(h_chest <- hist(chest_vec, plot = FALSE))
```

```
## $breaks
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##
## $counts
## [1] 21 81 185 420 749 1073 1079 934 658 370 92 50 21 4 1
##
## $density
## [1] 0.0036598118 0.0141164169 0.0322411990 0.0731962356 0.1305332869
## [6] 0.1869989543 0.1880446148 0.1627744859 0.1146741025 0.0644823980
## [11] 0.0160334611 0.0087138376 0.0036598118 0.0006971070 0.0001742768
##
## $mids
## [1] 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5 47.5
##
## $xname
## [1] "chest_vec"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

```
list(breaks = h_chest$breaks,
     counts = h_chest$counts,
     density = h_chest$density,
     mids = h_chest$mids)
```

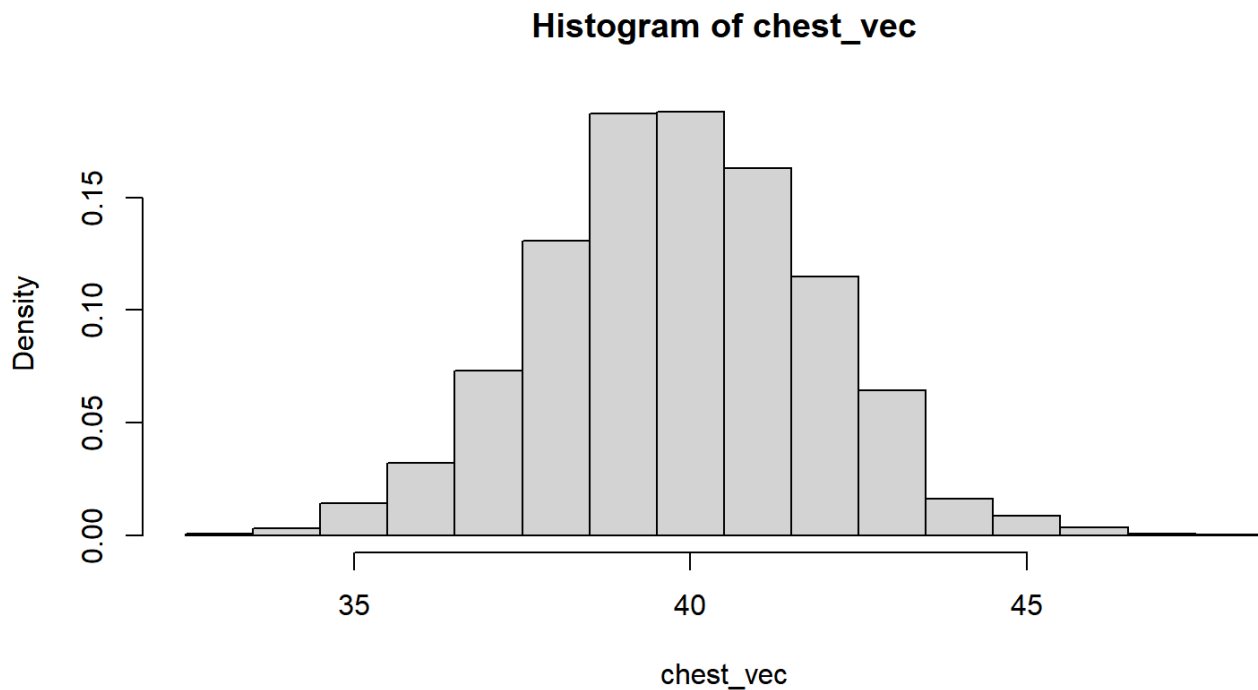
```
## $breaks
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##
## $counts
## [1] 21 81 185 420 749 1073 1079 934 658 370 92 50 21 4 1
##
## $density
## [1] 0.0036598118 0.0141164169 0.0322411990 0.0731962356 0.1305332869
## [6] 0.1869989543 0.1880446148 0.1627744859 0.1146741025 0.0644823980
## [11] 0.0160334611 0.0087138376 0.0036598118 0.0006971070 0.0001742768
##
## $mids
## [1] 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5 47.5
```

```
chest_vec %>%
  hist(plot = FALSE) %>%
  list(breaks = .$breaks,
       counts = .$counts,
       density = .$density,
       mids = .$mids)
```

```
## [[1]]
## $breaks
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##
## $counts
## [1] 21 81 185 420 749 1073 1079 934 658 370 92 50 21 4 1
##
## $density
## [1] 0.0036598118 0.0141164169 0.0322411990 0.0731962356 0.1305332869
## [6] 0.1869989543 0.1880446148 0.1627744859 0.1146741025 0.0644823980
## [11] 0.0160334611 0.0087138376 0.0036598118 0.0006971070 0.0001742768
##
## $mids
## [1] 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5 47.5
##
## $xname
## [1] "."
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
##
## $breaks
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##
## $counts
## [1] 21 81 185 420 749 1073 1079 934 658 370 92 50 21 4 1
##
## $density
## [1] 0.0036598118 0.0141164169 0.0322411990 0.0731962356 0.1305332869
## [6] 0.1869989543 0.1880446148 0.1627744859 0.1146741025 0.0644823980
## [11] 0.0160334611 0.0087138376 0.0036598118 0.0006971070 0.0001742768
##
## $mids
## [1] 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5 47.5
```

- 평균값과 표준편차로부터 히스토그램의 위치가 0.5만큼 왼쪽으로 치우쳐 있다는 것을 알 수 있음. 제자리에 옮겨 놓기 위해서 `breaks` 매개변수를 32.5부터 48.5까지 1간격으로 설정

```
hist(chest_vec,
     probability = TRUE,
     breaks = 32.5:48.5)
```



- 위의 히스토그램을 그리느라고 계산된 값들은?

```
h_chest_2 <- hist(chest_vec,
                  breaks = 32.5:48.5,
                  plot = FALSE)
list(breaks = h_chest_2$breaks,
     counts = h_chest_2$counts,
     density = h_chest_2$density,
     mids = h_chest_2$mids)
```

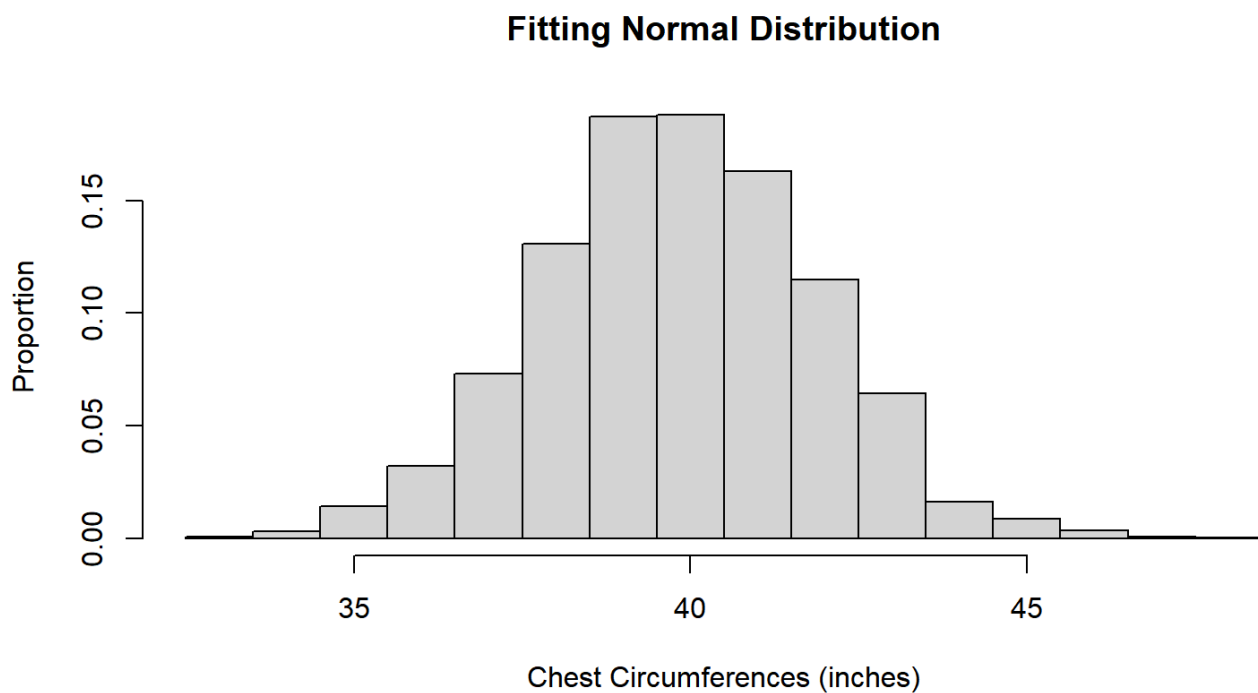
```
## $breaks
## [1] 32.5 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5
## [16] 47.5 48.5
##
## $counts
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
##
## $density
## [1] 0.0005228303 0.0031369815 0.0141164169 0.0322411990 0.0731962356
## [6] 0.1305332869 0.1869989543 0.1880446148 0.1627744859 0.1146741025
## [11] 0.0644823980 0.0160334611 0.0087138376 0.0036598118 0.0006971070
## [16] 0.0001742768
##
## $mids
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
```

```
chest_vec %>%
  hist(breaks = 32.5:48.5,
       plot = FALSE) %>%
  list(breaks = .$breaks,
       counts = .$counts,
       density = .$density,
       mids = .$mids)
```

```
## [[1]]
## $breaks
## [1] 32.5 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5
## [16] 47.5 48.5
##
## $counts
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
##
## $density
## [1] 0.0005228303 0.0031369815 0.0141164169 0.0322411990 0.0731962356
## [6] 0.1305332869 0.1869989543 0.1880446148 0.1627744859 0.1146741025
## [11] 0.0644823980 0.0160334611 0.0087138376 0.0036598118 0.0006971070
## [16] 0.0001742768
##
## $mids
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
##
## $xname
## [1] "."
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
##
## $breaks
## [1] 32.5 33.5 34.5 35.5 36.5 37.5 38.5 39.5 40.5 41.5 42.5 43.5 44.5 45.5 46.5
## [16] 47.5 48.5
##
## $counts
## [1] 3 18 81 185 420 749 1073 1079 934 658 370 92 50 21 4
## [16] 1
##
## $density
## [1] 0.0005228303 0.0031369815 0.0141164169 0.0322411990 0.0731962356
## [6] 0.1305332869 0.1869989543 0.1880446148 0.1627744859 0.1146741025
## [11] 0.0644823980 0.0160334611 0.0087138376 0.0036598118 0.0006971070
## [16] 0.0001742768
##
## $mids
## [1] 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
```

- 히스토그램을 보기 쉽게 하기 위해서 메인 타이틀과 서브 타이틀, x축 라벨, y축 라벨 설정


```
main_title <- "Fitting Normal Distribution"
# sub_title <- "Chest Circumferences of Scottish Soldiers"
sub_title <- ""
x_lab <- "Chest Circumferences (inches)"
y_lab <- "Proportion"
hist(chest_vec,
     breaks = 32.5:48.5,
     probability = TRUE,
     main = main_title,
     sub = sub_title,
     xlab = x_lab,
     ylab = y_lab)
```



Mean \pm SD contains 2/3 of total number of counts

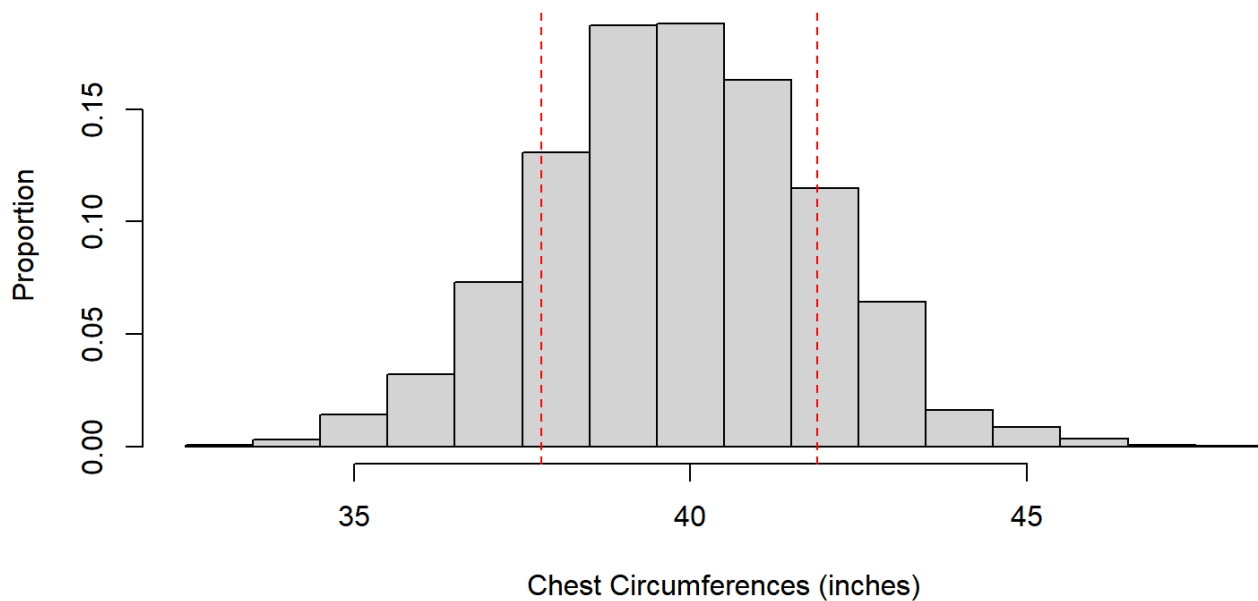
- 평균을 중심으로 \pm 표준편차 만큼 떨어진 자료를 붉은 색 수직점선으로 표시.

```

mean_chest <- mean(chest_vec)
sd_chest <- sd(chest_vec)
x_lower <- mean_chest - sd_chest
x_upper <- mean_chest + sd_chest
sd_chest <- chest_vec %>%
  sd
x_lower <- chest_vec %>%
  mean %>%
  `-(sd_chest)`
x_upper <- chest_vec %>%
  mean %>%
  `+(sd_chest)`
hist(chest_vec,
     breaks = 32.5:48.5,
     probability = TRUE,
     main = main_title,
     sub = sub_title,
     xlab = x_lab,
     ylab = y_lab)
abline(v = c(x_lower, x_upper),
      lty = 2,
      col = "red")

```

Fitting Normal Distribution



- 그 사이의 영역을 빗금으로 표시하기 위하여 다각형의 좌표를 계산

```
h_chest_2$density[6:10]
```

```
## [1] 0.1305333 0.1869990 0.1880446 0.1627745 0.1146741
```

```
y <- h_chest_2$density[6:10]
```

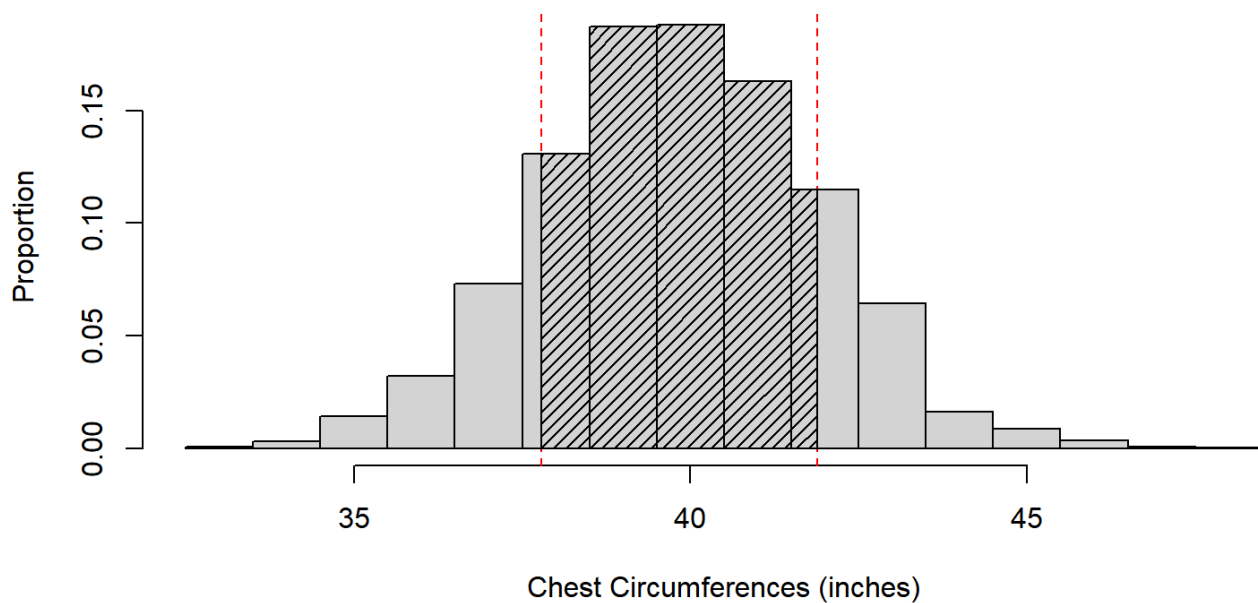
- 5개의 직사각형으로 파악하고 향후 면적 계산을 쉽게 하기 위하여 다음과 같이 좌표 설정

```

x_coord <- rep(c(x_lower, 38.5:41.5, x_upper), each = 2)
y_coord <- c(0, rep(y, each = 2), 0)
poly_df <- data.frame(x = x_coord, y = y_coord)
hist(chest_vec,
     breaks = 32.5:48.5,
     probability = TRUE,
     main = main_title,
     sub = sub_title,
     xlab = x_lab,
     ylab = y_lab)
abline(v = c(x_lower, x_upper),
      lty = 2,
      col = "red")
# polygon(x_coord, y_coord, density = 20)
polygon(poly_df,
#       col = "grey",
#       border = NA)
      density = 20)

```

Fitting Normal Distribution



- 이론적으로 빗금친 부분의 면적은 $\text{pnorm}(1) - \text{pnorm}(-1) = 0.6826895$ 에 가까울 것으로 예상. 5개 직사각형의 면적을 구하여 합하는 과정은 다음과 같음.

```

options(digits = 3)
x_area <- c(x_lower, 38.5:41.5, x_upper)
y

```

```
## [1] 0.131 0.187 0.188 0.163 0.115
```

```
diff(x_area)
```

```
## [1] 0.718 1.000 1.000 1.000 0.381
```

```
diff(x_area) * y
```

```
## [1] 0.0937 0.1870 0.1880 0.1628 0.0437
```

```
sum(diff(x_area) * y)
```

```
## [1] 0.675
```

```
source("./area.R")
area_R
```

```
## function (x, y)
## {
##     sum(diff(x) * (head(y, -1) + tail(y, -1))/2)
## }
```

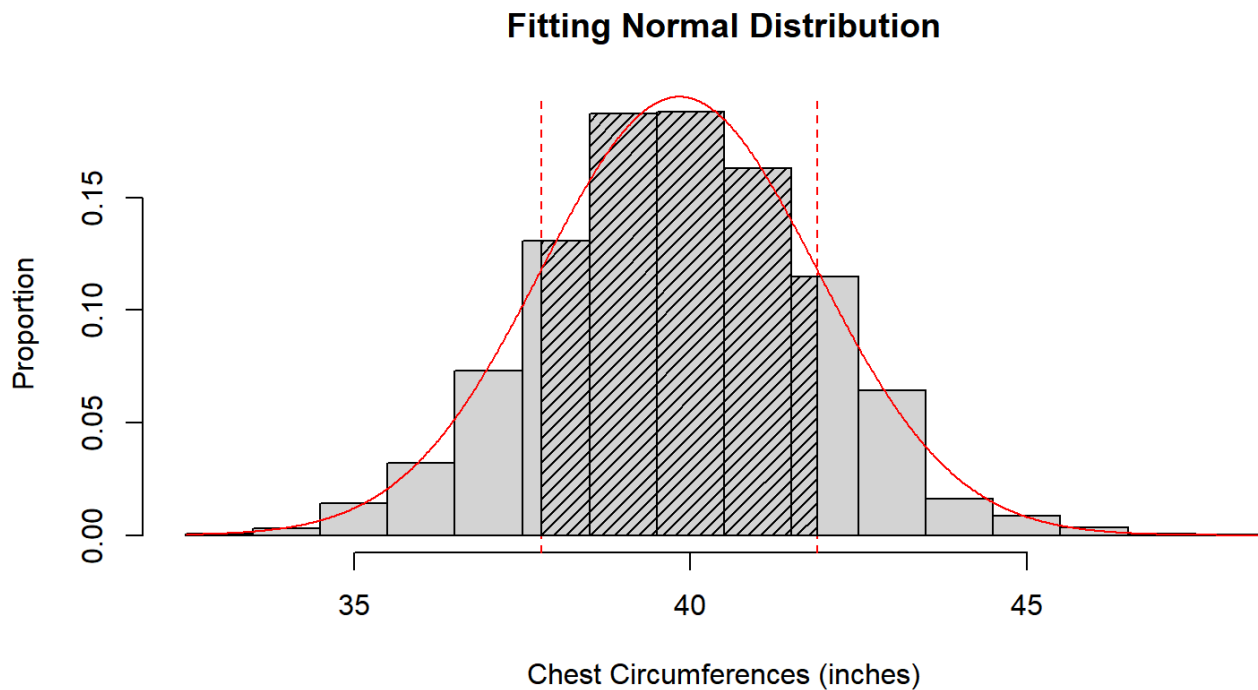
```
area_R(x_coord, y_coord)
```

```
## [1] 0.675
```

Comparison with normal curve

- 이론적인 정규분포 밀도함수 곡선을 히스토그램에 덧붙여 그림.

```
x_chest <- seq(32.5, 48.5,
              length = 1000)
y_norm <- dnorm(x_chest,
               mean = mean_chest,
               sd = sd_chest)
hist(chest_vec,
     breaks = 32.5:48.5,
     probability = TRUE,
     main = main_title,
     sub = sub_title,
     xlab = x_lab,
     ylab = y_lab)
abline(v = c(x_lower, x_upper),
      lty = 2,
      col = "red")
# abline(v = c(38, 42), lty = 2, col = "red")
polygon(poly_df,
      density = 20)
# polygon(x_coord, y_coord, density = 20)
lines(x_chest, y_norm, col = "red")
```

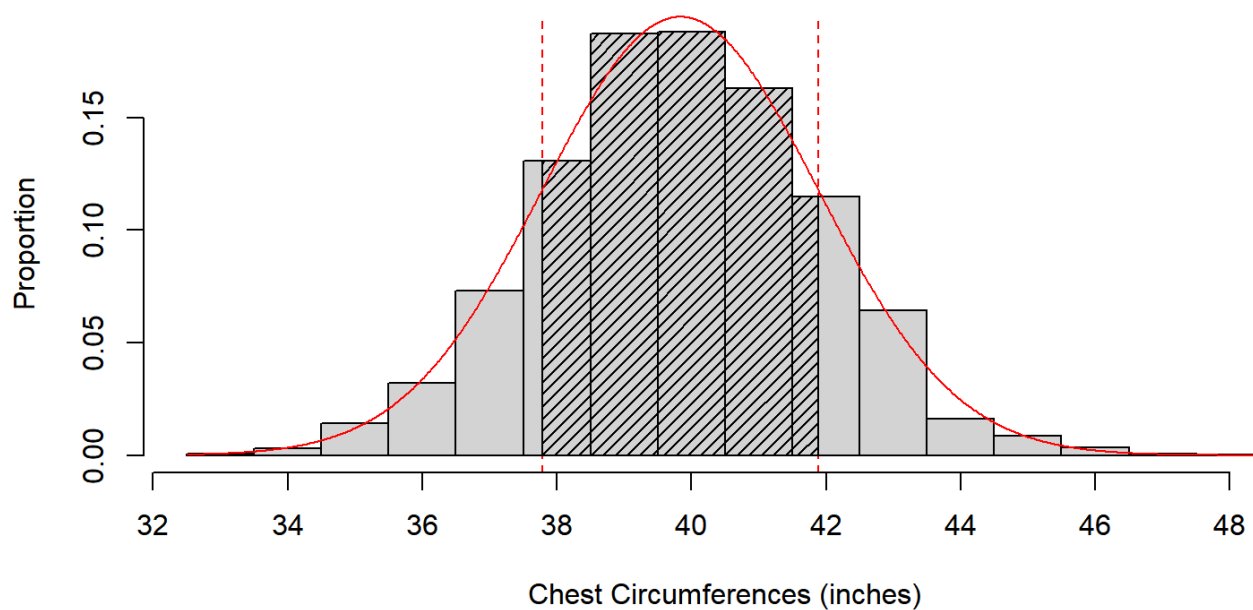


Changing tick marks of x axis

- default로 주어지는 x 축의 눈금을 제대로 볼 수 있게 고치려면,

```
hist(chest_vec,
     breaks = 32.5:48.5,
     probability = TRUE,
     main = main_title,
     sub = sub_title,
     xlab = x_lab,
     ylab = y_lab,
     axes = FALSE)
abline(v = c(x_lower, x_upper),
       lty = 2,
       col = "red")
polygon(poly_df,
       density = 20)
# polygon(x_coord, y_coord, density = 20)
lines(x_chest, y_norm, col = "red")
axis(side = 1,
     at = seq(32, 48, by = 2),
     labels = seq(32, 48, by = 2))
axis(side = 2)
```

Fitting Normal Distribution

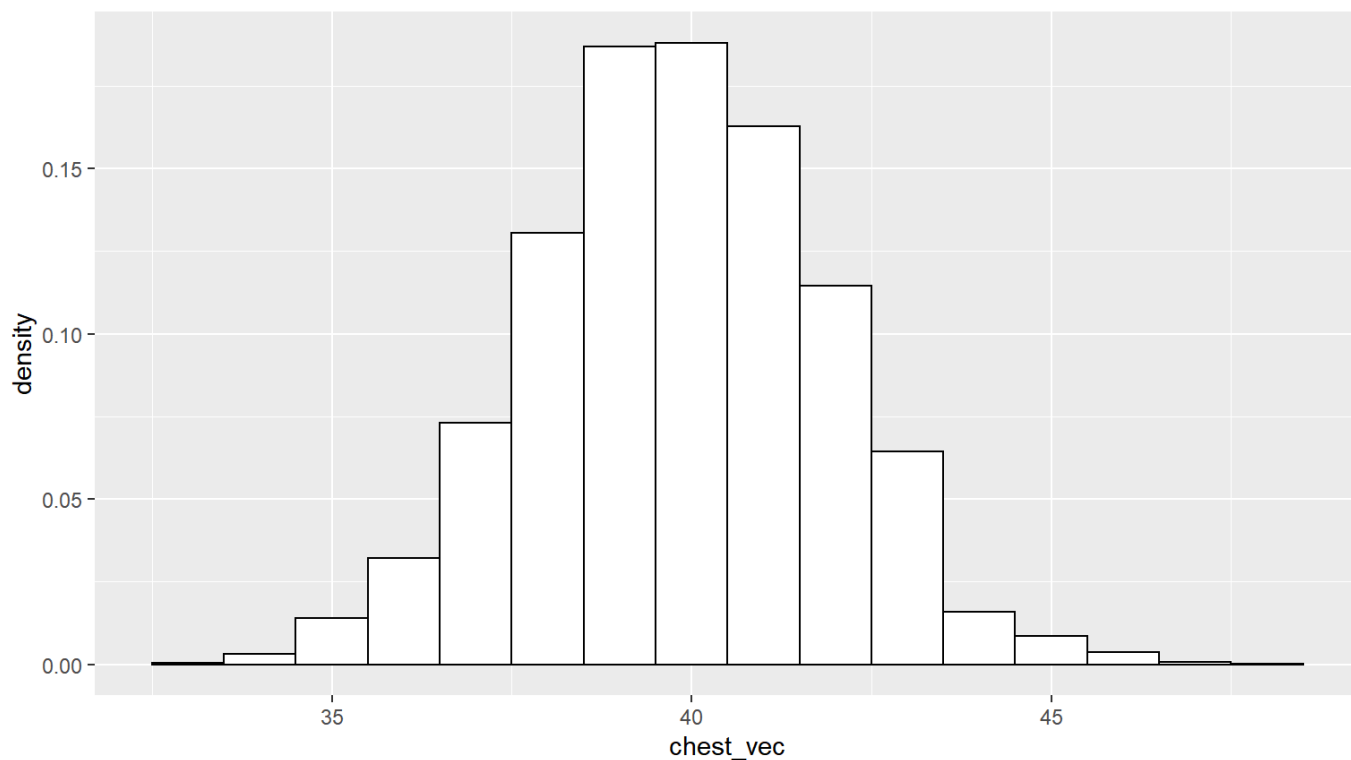


ggplot

- data frame으로 작업.

Basic histogram

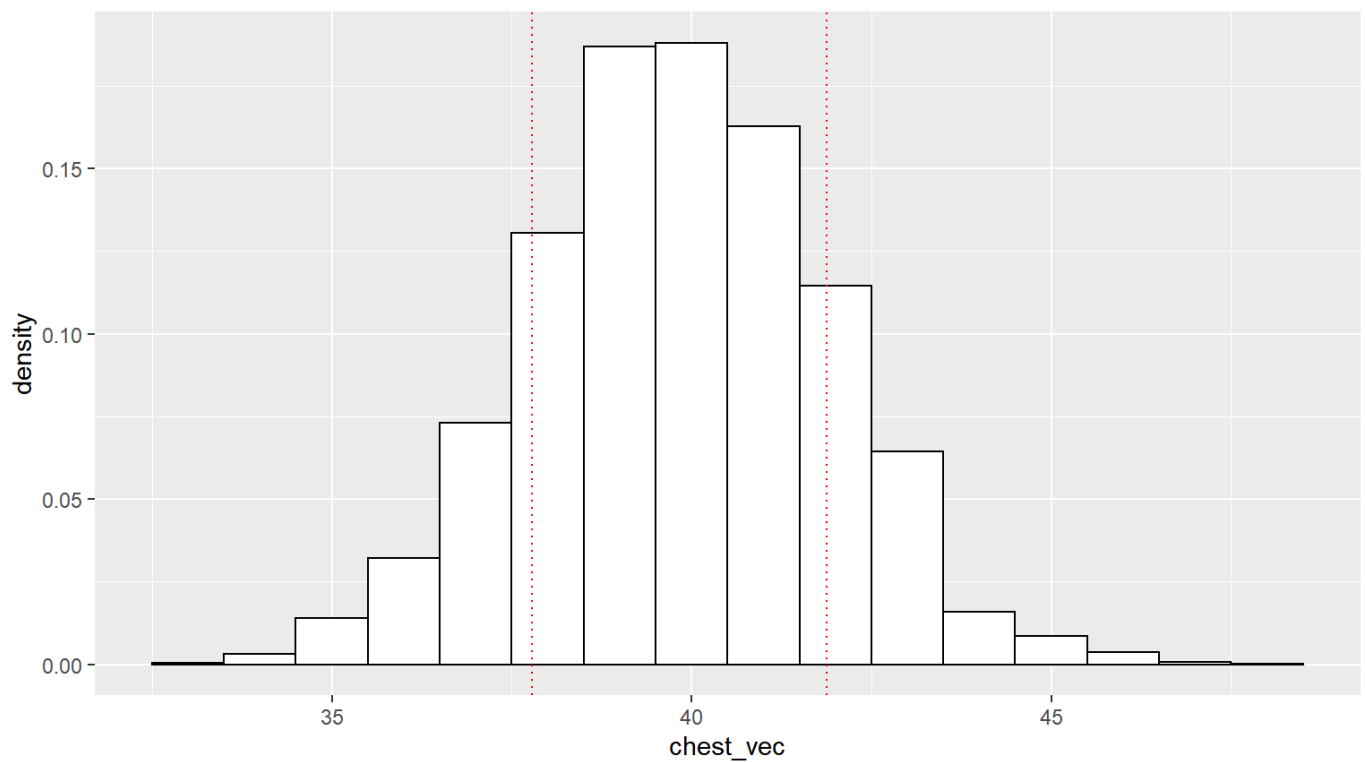
```
library(ggplot2)
# theme_update(plot.title = element_text(hjust = 0.5))
g0 <- ggplot(data = data.frame(chest_vec),
             mapping = aes(x = chest_vec))
(g1 <- g0 +
  stat_bin(aes(y = ..density..),
           binwidth = 1,
           fill = "white",
           colour = "black"))
```



```
# (g1 <- g0 +
#   stat_count(fill = "white",
#             colour = "black"))
# (g1 <- g0 +
#   geom_histogram(aes(y = ..density..),
#                 binwidth = 1,
#                 fill = "white",
#                 colour = "black"))
# (g1 <- g0 +
#   geom_histogram(aes(y = ..density..),
#                 binwidth = 1,
#                 breaks = 32.5:48.5,
#                 fill = "white",
#                 colour = "black"))
```

Mean \pm SD

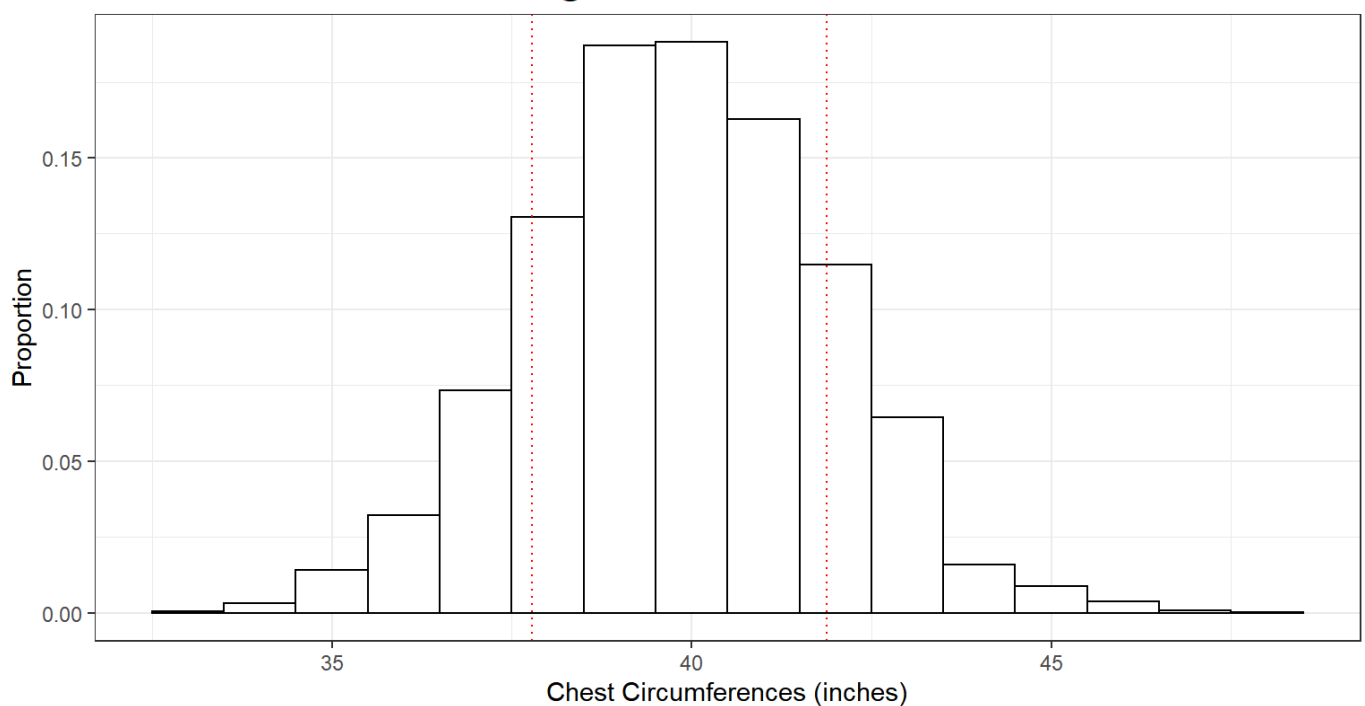
```
(g2 <- g1 +
  geom_vline(xintercept = c(x_lower, x_upper),
            linetype = "dotted",
            colour = "red"))
```



x-axis label and main title

```
(g3 <- g2 +
  theme_bw() +
  # xlab(x_lab) +
  # ylab(y_lab) +
  # ggtitle(main_title) +
  labs(title = main_title,
        x = x_lab,
        y = y_lab) +
  theme(plot.title = element_text(hjust = 0.5,
                                   size = 20)))
```

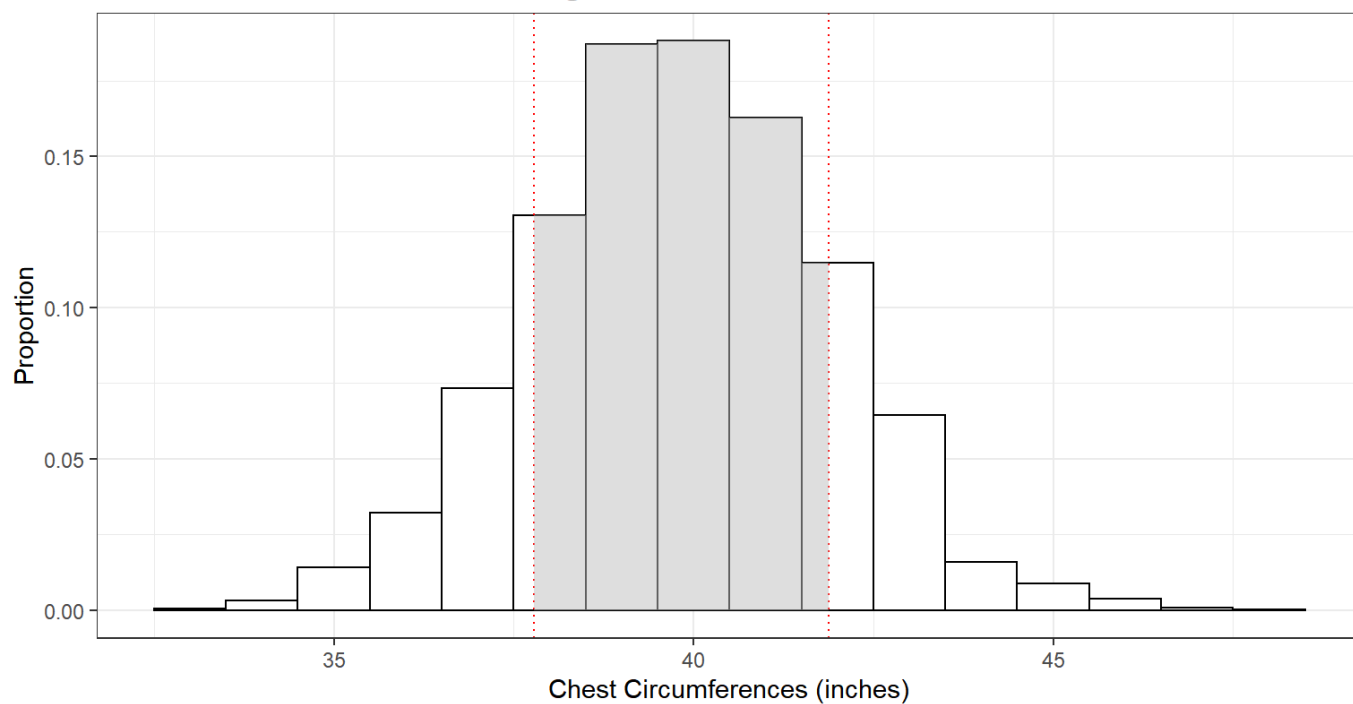
Fitting Normal Distribution



Shading the area

```
(g4 <- g3 +
  geom_polygon(data = poly_df,
    mapping = aes(x = x, y = y),
    alpha = 0.5,
    fill = "grey"))
```

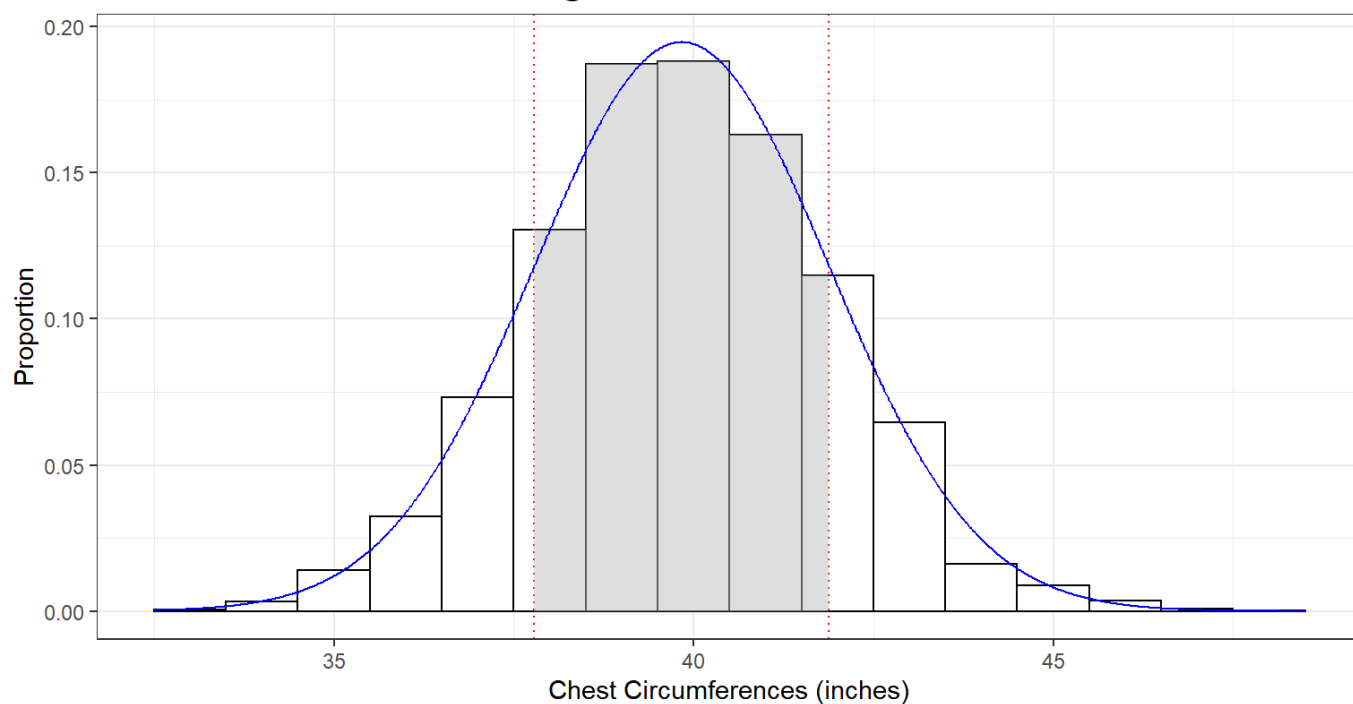
Fitting Normal Distribution



Normal curve added

```
# x_curve <- seq(32.5, 48.5, length = 100)
# y_curve <- dnorm(x_curve, mean = mean_chest, sd = sd_chest)
curve_df <- data.frame(x = x_curve, y = y_curve)
(g5 <- g4 +
  geom_line(data = curve_df,
    mapping = aes(x = x, y = y),
    colour = "blue"))
```

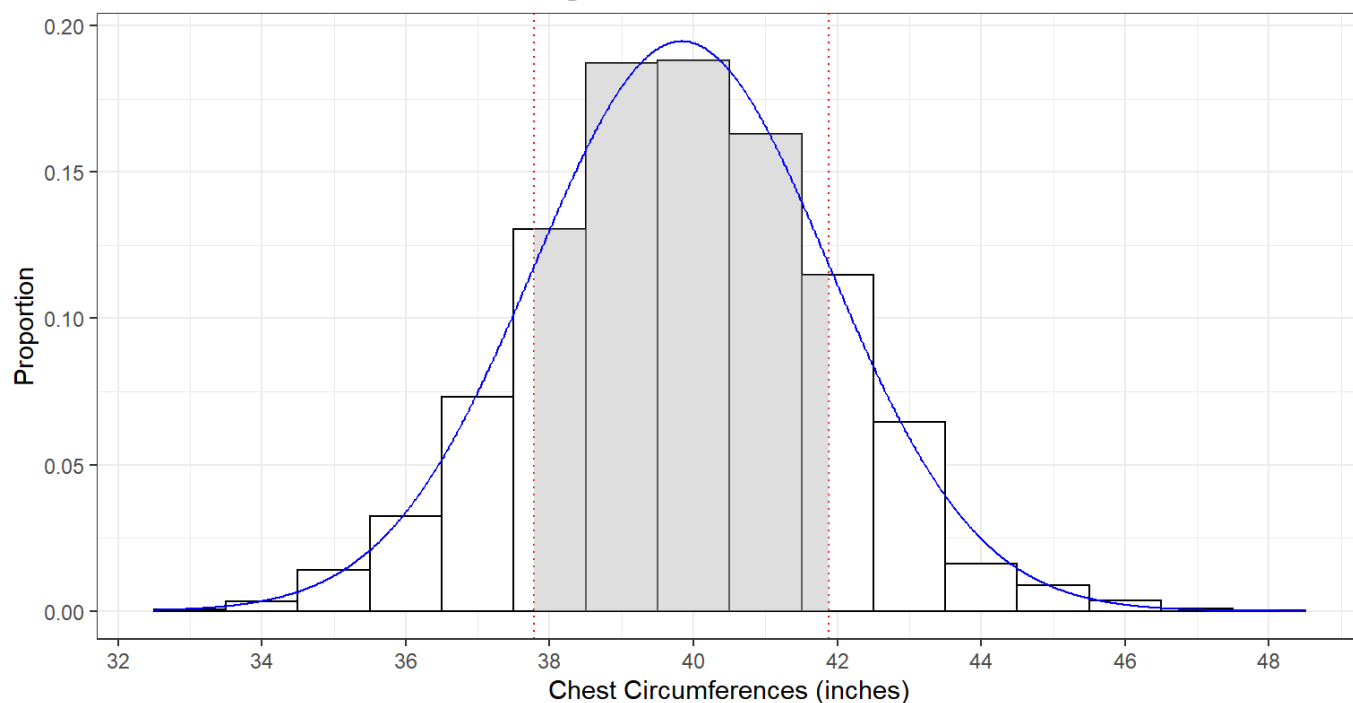
Fitting Normal Distribution



x-axis tick marks

```
(g6 <- g5 +
  scale_x_continuous(breaks = seq(32, 48, by = 2),
    labels = seq(32, 48, by = 2)))
```

Fitting Normal Distribution



Save

```
getwd()
```

```
## [1] "C:/Users/kki96/Documents/r_programming_2/R"
```

```
save.image(file = "./Quetelet_chest.RData")
```

Comments

이번 시간에는 케틀레가 만든 군인의 가슴둘레 분포표를 시각화 하는 법을 배울 수 있는 시간 이였습니다. 데이터프레임을 만드는 과정에서 실수와 정수의 차이에 대해 알 수 있는 시간이었고, piping을 쓴 것과 쓰지 않은 것을 비교하여 piping에 대하여 더 이해되는 시간이였습니다. piping으로 ']'을 표현하는 법을 좀 더 쉽게 이해할 수 있었습니다. piping을 활용하여 확률히스토그램을 만드는 과정을 배울 수 있었는데 기존보다 더 간단하게 만드는 과정을 알게 되었습니다. 면적을 구하는 방법을 area.R을 통해 알게 되었습니다. 이 과정들을 통해 정규분포와 닮았다는 것을 알게 되었습니다. 또한, ggplot 2를 이용해 만드는 방법 또한 배우게 되었는데, 비슷한 결과를 나타내는 여러 함수를 통하여, 나에게 맞는 것을 찾을 수 있게 되었던 것 같습니다.