

## Computer Lab Week 8: solutions

STAT221

```
library(datasets)
data(faithful)
help(faithful)
```

```
## starting httpd help server ... done
```

```
head(faithful, 6)
```

```
##   eruptions waiting
## 1      3.600      79
## 2      1.800      54
## 3      3.333      74
## 4      2.283      62
## 5      4.533      85
## 6      2.883      55
```

```
summary(faithful)
```

```
##   eruptions      waiting
## Min.   :1.600   Min.   :43.0
## 1st Qu.:2.163   1st Qu.:58.0
## Median :4.000   Median :76.0
## Mean   :3.488   Mean   :70.9
## 3rd Qu.:4.454   3rd Qu.:82.0
## Max.   :5.100   Max.   :96.0
```

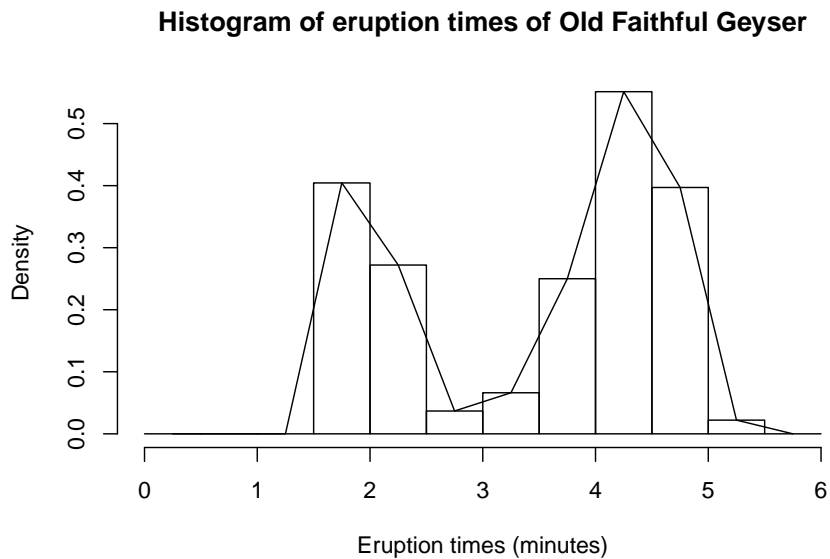
The range of the variable `eruptions` is 1.6 to 5.1, so we can plot the histogram x axis from 0 to 6, say.

### Frequency polygons and ASH plots

1. In lab 7 you used your judgement to select your preferred best bin width for a histogram of this dataset. Using that bin width, add a frequency polygon to your histogram.

```
# From last lab asked for a density histogram
H = hist(faithful$eruptions, breaks = seq(0, 6, 0.5), freq = FALSE,
        main = "Histogram of eruption times of Old Faithful Geyser",
        xlab = "Eruption times (minutes)")

lines(H$mids, H$density, type = "l")
```

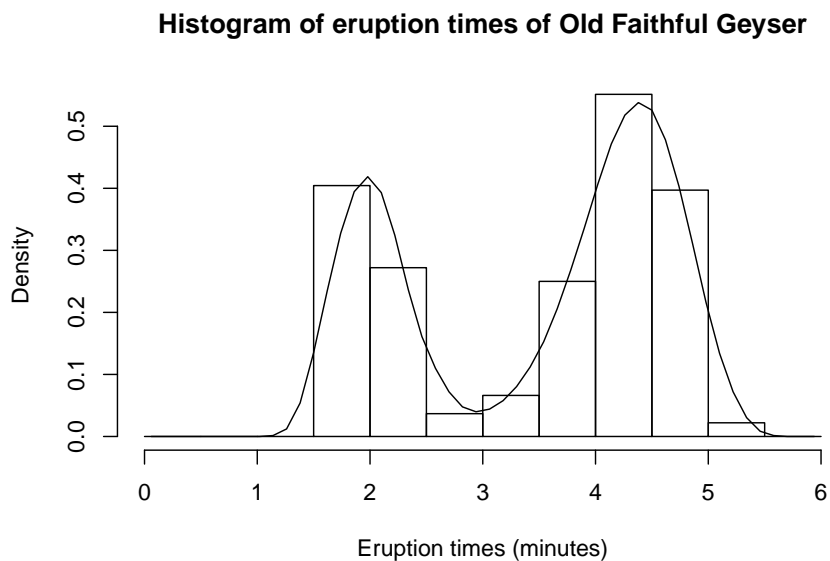


2. Replot the histogram, and this time add an ASH (average shifted histogram). You will need to use the package `ash` and the code in the lecture notes.

```
library(ash)

bins = bin1(faithful$eruptions, ab = c(0, 6), 50) # 50 bins
ash1 = ash1(bins, m = 5) # smoothing parameter is 5

hist(faithful$eruptions,
     breaks = seq(0, 6, 0.5), # max is 5.1 for the eruptions variable
     freq = FALSE,
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)",
     lines(ash1$x, ash1$y)) # add the ASH plot
```



3. Overlay on the graph further ASH plots, with the same smoothing parameter, but different numbers of bins. Each ASH plot should use a different coloured line.

```

ab = c(0, 6)

bins1 = bin1(faithful$eruptions, ab, 5)
ash1a = ash1(bins1, m=5)

## [1] "ash estimate nonzero outside interval ab"

bins2 = bin1(faithful$eruptions, ab, 10)
ash2a = ash1(bins2, m=5)

## [1] "ash estimate nonzero outside interval ab"

bins3 = bin1(faithful$eruptions, ab, 20)
ash3a = ash1(bins3, m=5)

## [1] "ash estimate nonzero outside interval ab"

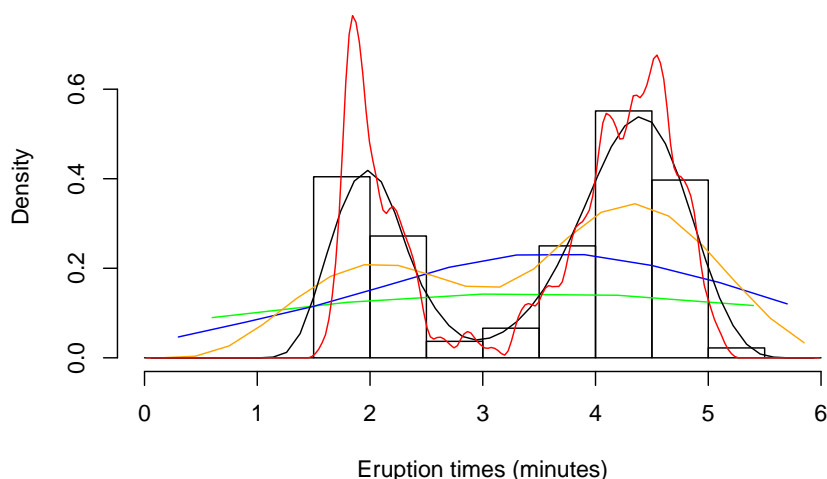
bins4 = bin1(faithful$eruptions, ab, 50)
ash4a = ash1(bins4, m=5)

bins5 = bin1(faithful$eruptions, ab, 200)
ash5a = ash1(bins5, m=5)

max.y = max(ash1a$y, ash2a$y, ash3a$y, ash4a$y, ash5a$y)

hist(faithful$eruptions, breaks = seq(0, 6, 0.5),
     freq = FALSE, ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)")
lines(ash1a$x, ash1a$y, col = "green")
lines(ash2a$x, ash2a$y, col = "blue")
lines(ash3a$x, ash3a$y, col = "orange")
lines(ash4a$x, ash4a$y, col = "black")
lines(ash5a$x, ash5a$y, col = "red")

```



There are some warning messages saying that when we use a smaller number of bins (when there are 5, 10, or 20 bins) that “ash estimate nonzero outside interval ab”. This is because the ASH plot doesn’t go down to zero within the  $x$  range of the plot. The ASH function still runs just fine, but it is giving a warning to prompt us to check we are happy with what it has calculated.

If we were concerned about this, we could try widening the  $x$  limits of the plot:

```

ab = c(-30,30)

bins1 = bin1(faithful$eruptions, ab, 5)
ash1a = ash1(bins1, m=5)

## [1] "ash estimate nonzero outside interval ab"

bins2 = bin1(faithful$eruptions, ab, 10)
ash2a = ash1(bins2, m=5)

bins3 = bin1(faithful$eruptions, ab, 20)
ash3a = ash1(bins3, m=5)

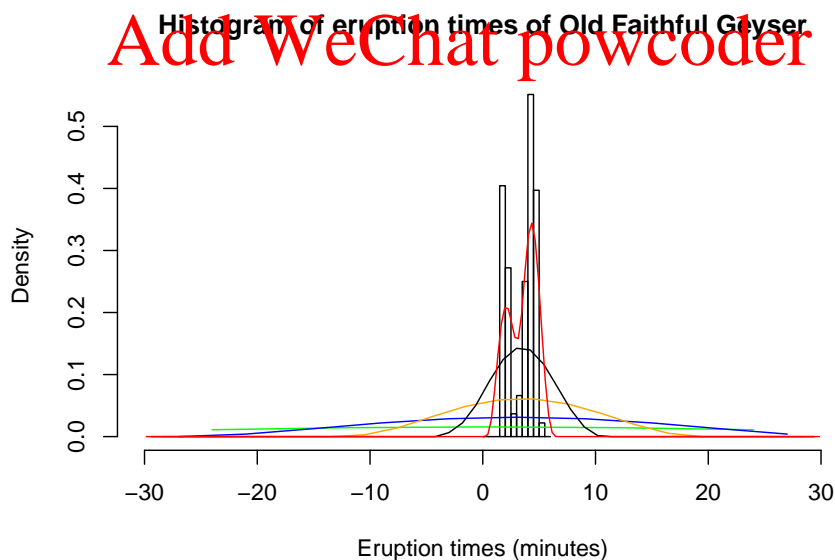
bins4 = bin1(faithful$eruptions, ab, 50)
ash4a = ash1(bins4, m=5)

bins5 = bin1(faithful$eruptions, ab, 200)
ash5a = ash1(bins5, m=5)

max.y = max(ash1a$y, ash2a$y, ash3a$y, ash4a$y, ash5a$y)

hist(faithful$eruptions, breaks = seq(0, 6, 0.5),
     freq = FALSE, xlim=c(-30,30), # ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)",
     lines(ash1a$x, ash1a$y, col = "green")
     lines(ash2a$x, ash2a$y, col = "blue")
     lines(ash3a$x, ash3a$y, col = "orange")
     lines(ash4a$x, ash4a$y, col = "black")
     lines(ash5a$x, ash5a$y, col = "red"))

```



4. Choose one of the bin interval settings and reproduce the graph with different smoothing parameter values.

```

ab = c(0, 6)
bins = bin1(faithful$eruptions, ab, 50)
ash1 = ash1(bins, m=0.5)
ash2 = ash1(bins, m=2)
ash3 = ash1(bins, m=5)

```

```

ash4 = ash1(bins, m=10)

## [1] "ash estimate nonzero outside interval ab"

ash5 = ash1(bins, m=50)

## [1] "ash estimate nonzero outside interval ab"

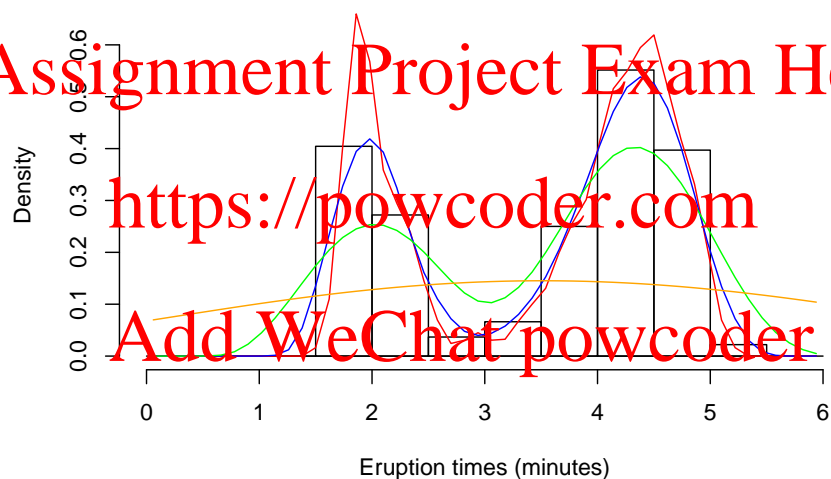
max.y = max(ash1$y, ash2$y, ash3$y, ash4$y, ash5$y)

hist(faithful$eruptions, breaks = seq(0, 6, 0.5), freq = FALSE, ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)")

lines(ash1$x, ash1$y)
lines(ash2$x, ash2$y, col = "red")
lines(ash3$x, ash3$y, col = "blue")
lines(ash4$x, ash4$y, col = "green")
lines(ash5$x, ash5$y, col = "orange")

```

Histogram of eruption times of Old Faithful Geyser



## Kernel Density Estimation

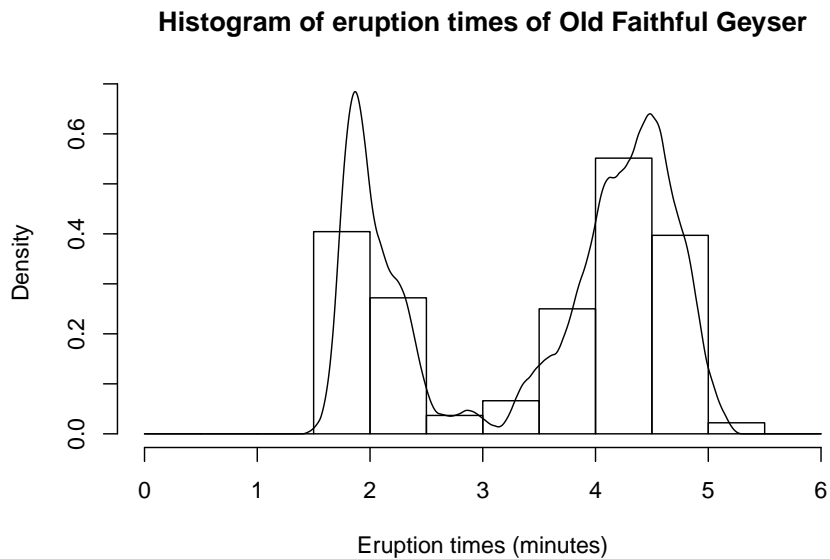
1. Reproduce the density histogram for the eruption time, with the bin width set at your chosen value. Use the kernel density estimation function in the evmix library to produce a kernel density estimate and overlay this on the graph. Use a bandwidth of  $h = 0.2$  and a biweight kernel.

```

library(evmix)
h = 0.2
x_values = seq(0, 6, 0.001)
y_values = dkden(x_values, faithful$eruptions, h, kernel = "biweight")
max.y = max(y_values)

hist(faithful$eruptions, breaks = seq(0, 6, 0.5),
     freq = FALSE, ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)")
lines(x_values, y_values)

```



2. Overlay on the graph further kernel density estimates, with the same bandwidth, but different kernels. To get a list of the available kernels, look at the help file:

```
help(kernels)
```

Each kernel density estimate should use a different coloured line.

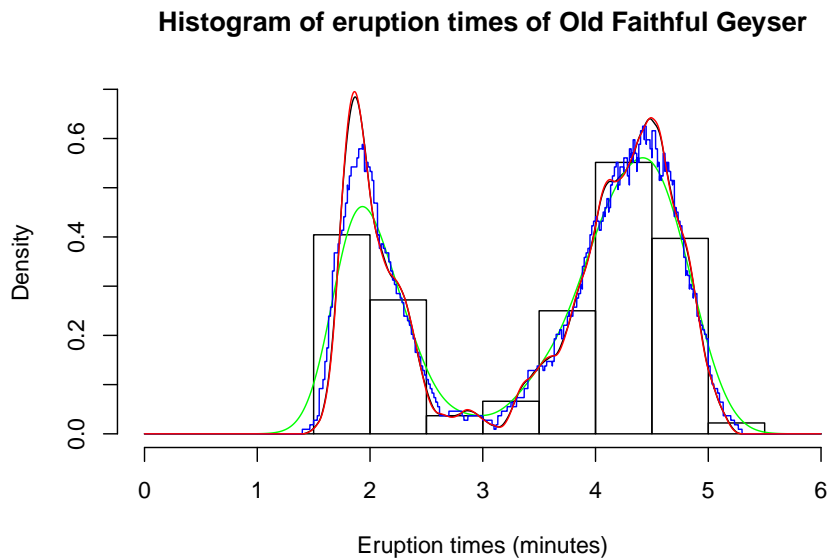
```
h = 0.2
```

```
x_values = seq(0, 6, 0.5)
y_values_kd1 = dkden(x_values,
                     faithful$eruptions, h,
                     kernel = "biweight")
y_values_kd2 = dkden(x_values,
                     faithful$eruptions, h,
                     kernel = "gaussian")
y_values_kd3 = dkden(x_values,
                     faithful$eruptions, h,
                     kernel = "uniform")
y_values_kd4 = dkden(x_values,
                     faithful$eruptions, h,
                     kernel = "cosine")

max.y = max(y_values_kd1, y_values_kd2, y_values_kd3, y_values_kd4)

hist(faithful$eruptions, breaks = seq(0, 6, 0.5), freq = FALSE, ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)")

lines(x_values, y_values_kd1, col = "black")
lines(x_values, y_values_kd2, col = "green")
lines(x_values, y_values_kd3, col = "blue")
lines(x_values, y_values_kd4, col = "red")
```

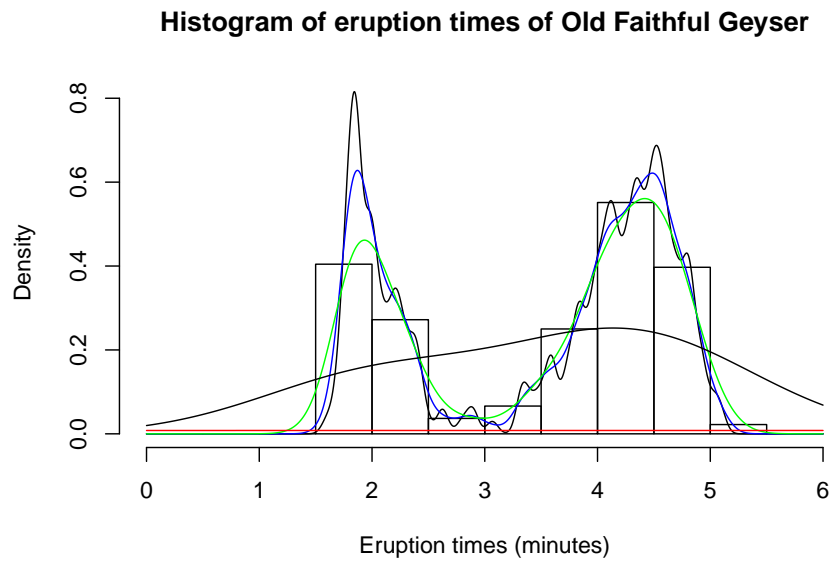


3. Choose one of the kernels and reproduce the graph with different bandwidths.

```
h1 = 0.05
h2 = 0.1
h3 = 0.2
h4 = 1
h5 = 50

x_values = seq(0, 6, 0.001)
y_values_h1 = dkden(x_values,
                    faithful$eruptions, h1,
                    kernel = "gaussian")
y_values_h2 = dkden(x_values,
                    faithful$eruptions, h2,
                    kernel = "gaussian")
y_values_h3 = dkden(x_values,
                    faithful$eruptions, h3,
                    kernel = "gaussian")
y_values_h4 = dkden(x_values,
                    faithful$eruptions, h4,
                    kernel = "gaussian")
y_values_h5 = dkden(x_values,
                    faithful$eruptions, h5,
                    kernel = "gaussian")
max.y = max(y_values_h1, y_values_h2, y_values_h3, y_values_h4, y_values_h5)
hist(faithful$eruptions, breaks = seq(0, 6, 0.5), freq = FALSE, ylim = c(0, max.y),
     main = "Histogram of eruption times of Old Faithful Geyser",
     xlab = "Eruption times (minutes)")

lines(x_values, y_values_h1)
lines(x_values, y_values_h2, col = "blue")
lines(x_values, y_values_h3, col = "green")
lines(x_values, y_values_h4, col = "black")
lines(x_values, y_values_h5, col = "red")
```



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder