HW2

110077443

2/26/2022

Question 1

a) Create and visualize a new Distribution 2

```
# Three normally distributed data sets
d4 <- rnorm(n=500, mean=45, sd=4)
d5 <- rnorm(n=200, mean=30, sd=4)
d6 <- rnorm(n=100, mean=15, sd=4)

# Combining them into a composite dataset
d456 <- c(d4, d5, d6)
mean(d456)
```

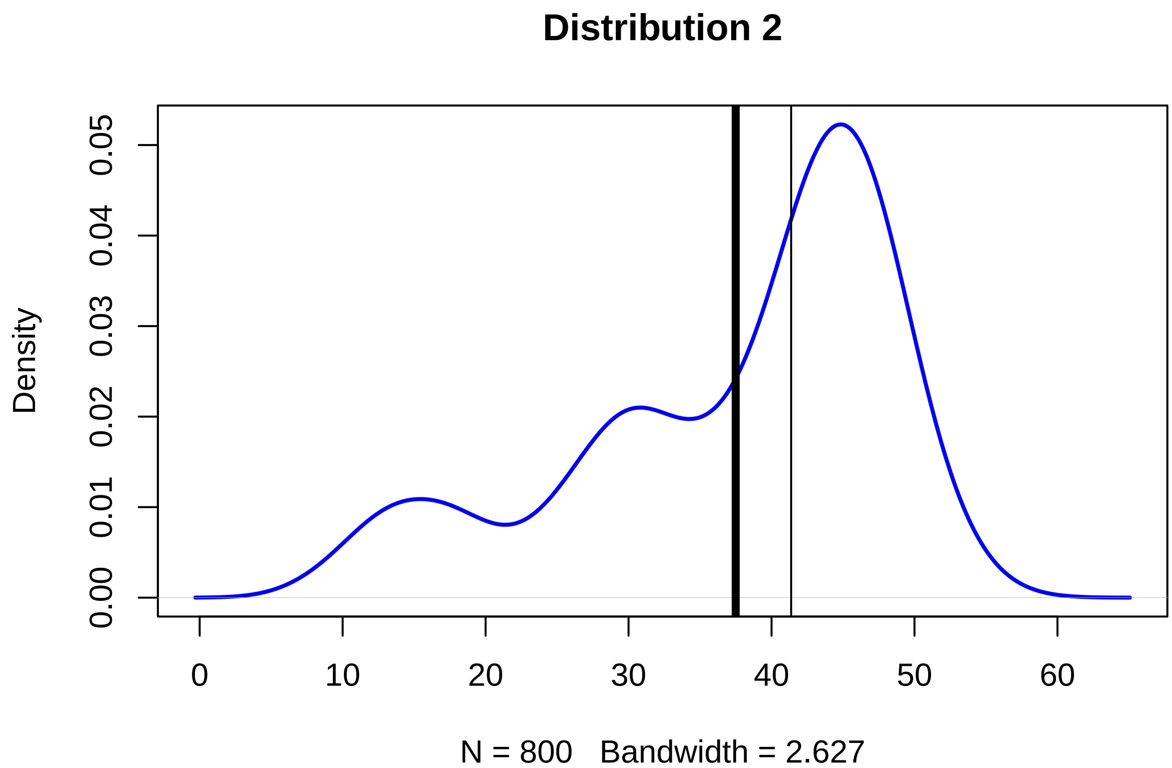
```
## [1] 37.49413
```

```
median(d456)
```

```
## [1] 41.37497
```

```
# Let's plot the density function of d123
plot(density(d456), col="blue", lwd=2,
     main = "Distribution 2")

# Add vertical lines showing mean and median
abline(v=mean(d456), lwd = 4) #Thicken mean line
abline(v=median(d456))
```



b) Create a Distribution 3: a single dataset that is normally distributed

```r
# Construct a normal distribution
db <- rnorm(n=800, mean=20, sd=3)

# Compute mean
mean(db)
```

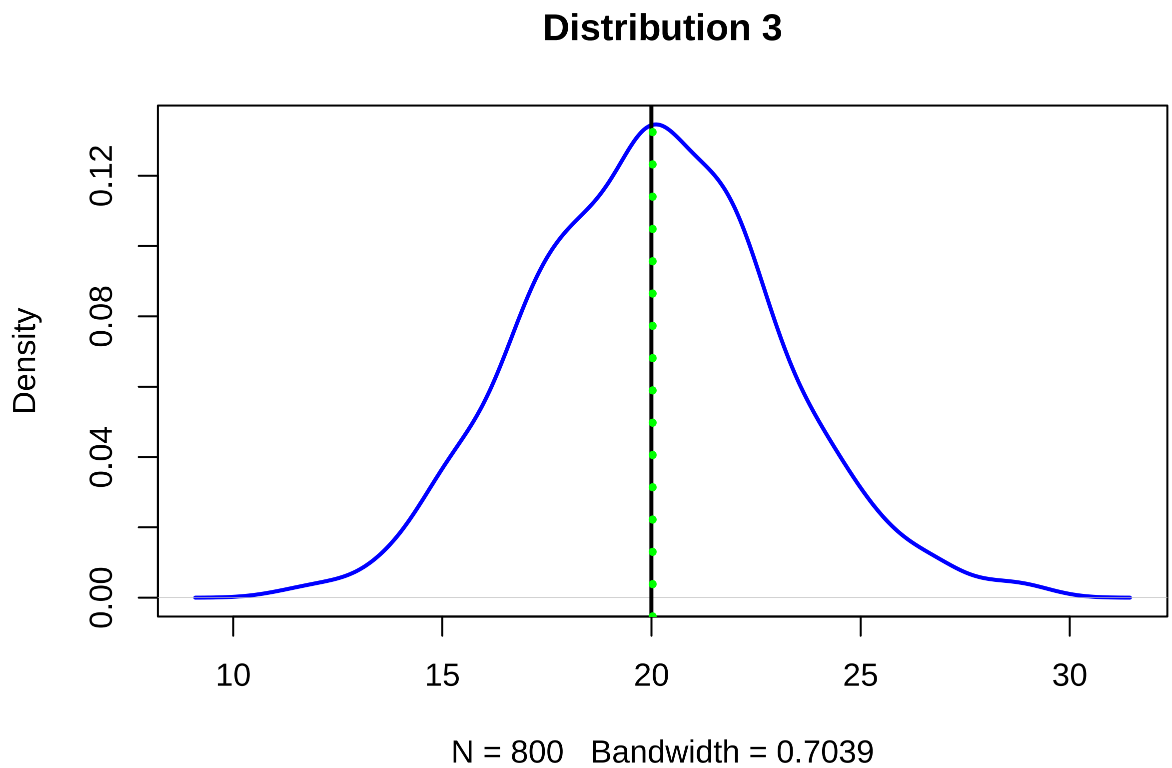
```
## [1] 19.99804
```

```r
# Compute median
median(db)
```

```
## [1] 20.02858
```

```r
# Plot
plot(density(db), col = "blue", lwd=2, main = "Distribution 3")

# Add vertical lines showing mean and median
abline(v=mean(db), lwd = 2)
abline(v=median(db), lwd = 4, lty = 3, col = "green") #color to show overlap
```



c) Which measure of central tendency (mean or median) is more sensitive to outliers

```
## I believe the mean is the most sensitive measure of central tendency
## because extreme values/outliers will alter the mean too much
```

Question 2

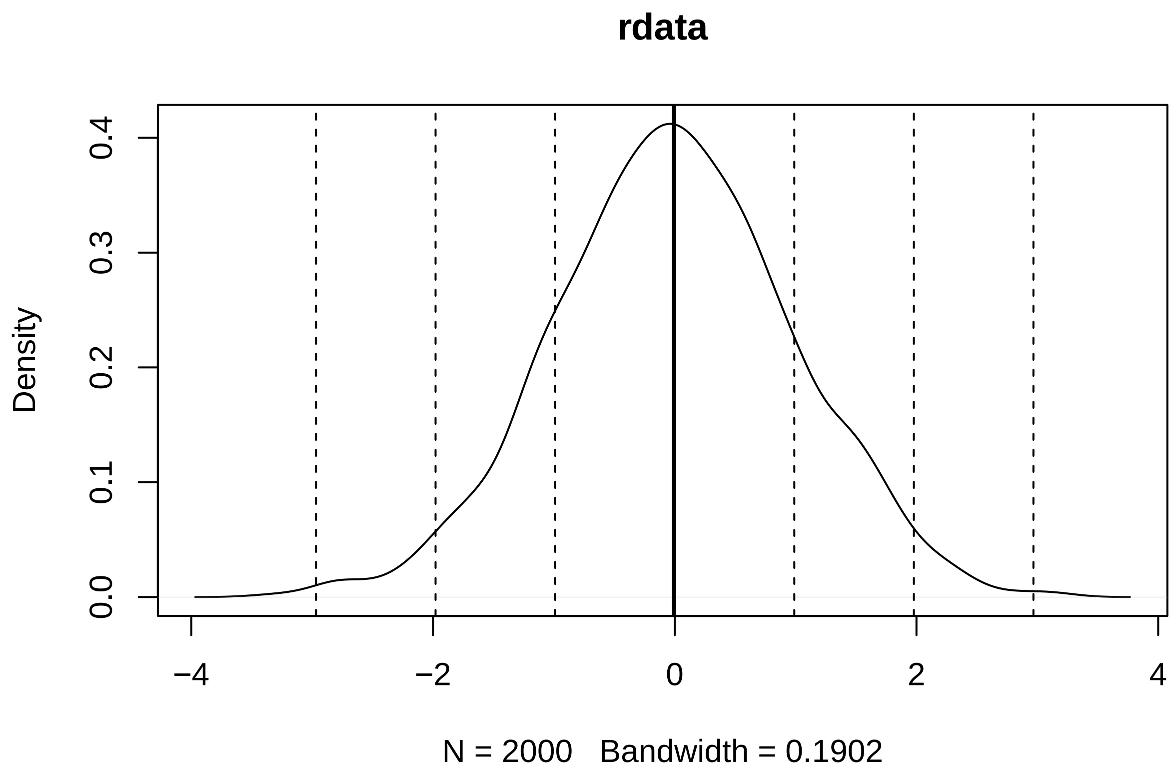
a) Create a random dataset, normally distributed with: n=2000, mean=0, sd=1

```
# Construct a normal distribution with mean=0 sd=1
set.seed(1234) # Making the random procedures reproducible
rdata <- rnorm(n=2000, mean=0, sd=1)

# density plot with solid vertical line on mean the mean
plot(density(rdata), main="rdata")
abline(v=mean(rdata), lwd = 2)

# dashed vertical lines on 1st,2nd, and 3rd standard deviations (left to right)
abline(v= -3 * sd(rdata), lty = "dashed")
abline(v= -2 * sd(rdata), lty = "dashed")
```

```
abline(v= -1 * sd(rdata), lty = "dashed")
abline(v= 1 * sd(rdata), lty = "dashed")
abline(v= 2 * sd(rdata), lty = "dashed")
abline(v= 3 * sd(rdata), lty = "dashed")
```



b) Calculate 25th, 50th, and 75th percentiles and SD each are from mean

```
# Calculate 25th, 50th, and 75th percentiles using quantile function
rdata_quantile <- quantile(rdata, probs = c(.25, .5, .75))
rdata_quantile # Check values
```

```
##         25%         50%         75%
## -0.65132459 -0.01600958  0.64377727
```

```
# How many standard deviations are the quartiles from mean
rdata_distance <- ((rdata_quantile-mean(rdata))/sd(rdata))
rdata_distance # Check values
```

```
##         25%         50%         75%
## -0.65228281 -0.01007325  0.65687373
```

```
#1st quartile = -0.65228281 standard deviations away from mean.
#2nd quartile = -0.01007325 standard deviations away from mean.
#3rd quartile =  0.65687373 standard deviations away from mean.
```

c) Create a new random dataset, normally distributed with: n=2000, mean=35, sd=3.5

```
# Construct a new normal distribution with mean=35 sd=3.5
set.seed(12345)
rdata_new <- rnorm(n=2000, mean=35, sd=3.5)

# Calculate 25th, 50th, and 75th percentiles using quantile function
rdata_new_quantile <- quantile(rdata_new, probs = c(.25, .5, .75))
rdata_new_quantile # Check values
```

```
##      25%      50%      75%
## 32.67740 35.05362 37.38101
```

```
# How many standard deviations are the quartiles from mean
rdata_distance <- ((rdata_new_quantile-mean(rdata_new))/sd(rdata_new))
rdata_distance # Check values
```

```
##          25%          50%          75%
## -0.668799243  0.007386277  0.669679349
```

```
# 1st quartile = -0.668799243 standard deviations away from mean.
# 2nd quartile =  0.007386277 standard deviations away from mean.
# 3rd quartile =  0.669679349 standard deviations away from mean.
## There is minimal difference between (b) and (c) corresponding to
## 1st and 3rd quartiles (not significant)
```

d) Dataset d123 - How many standard deviations are the 1st and 3rd quartiles from mean and compare with (b)

```
# Three normally distributed data sets
set.seed(123)
d1 <- rnorm(n=500, mean=15, sd=5)
d2 <- rnorm(n=200, mean=30, sd=5)
d3 <- rnorm(n=100, mean=45, sd=5)

# Combining them into a composite dataset
d123 <- c(d1, d2, d3)

# Calculate 25th and 75th percentiles using quantile function
d123_quantile <- quantile(d123, probs = c(.25, .75))
d123_quantile # Check values
```

```
##      25%      75%
## 13.84221 29.52890
```

```
# How many standard deviations are the quartiles from mean
d123_distance <- ((d123_quantile - mean(d123))/sd(d123))
d123_distance # Check values
```

```
##        25%        75%
## -0.7463353  0.5974905
```

```
# 1st quartile = -0.7463353 standard deviations away from mean.
# 3rd quartile =  0.5974905 standard deviations away from mean.
## There is minimal difference between (b) and (d) corresponding to
## 1st and 3rd quartiles (not significant)
```

Question 3

a) Which formula does Rob Hyndman's answer and what's its benefit?

```
## Rob Hyndman suggests the Freedman-Diaconis rule as opposed to Scott's rule.
## The benefit of this rule is that it is less sensitive to outliers in data since
## Scott's rule uses standard deviation.
```

b) Given a random normal distribution:

```
set.seed(123456)
rand_data <- rnorm(800, mean=20, sd = 5)
rand_data # Taking a look at the values
```

```
##   [1] 24.168666 18.619761 18.224991 20.437437 31.261279 24.172301 26.562078
##   [8] 32.513227 25.841159 17.869172 15.019351 14.430250 19.721342 25.872162
##  [15] 25.266093 20.288030 16.324786 24.652642 28.341055 22.798439 16.230126
##  [22] 26.282771 20.192463 20.947699 22.312975 17.863185 20.082930 23.524396
##  [29] 24.859247 16.897542 15.720665 20.347792 14.769009  6.255658 14.350702
##  [36] 15.691576 27.800369 25.075442 25.219972 14.420474 14.643478 24.839105
##  [43] 20.855163 15.518749 20.791446 17.490258 15.170387 19.431329 25.429749
##  [50] 13.941759 11.161390 17.541540 21.607329 27.303304 27.686213 18.302157
##  [57] 14.612756 12.546174 18.736267 19.390387 16.755083 21.565379 20.614053
##  [64] 15.815798 23.002109 18.770327 19.076677 20.115784 17.576742 16.310468
##  [71] 26.512538 22.822497 10.974778 22.191695 12.794555 24.764146 16.385088
##  [78] 20.243802 12.182339  8.488168 25.789818 16.476323 12.130889 22.591758
##  [85] 14.675347 20.235272 24.240237 22.163389 22.614027 18.730570 17.515740
##  [92] 26.300506 22.824901 18.276811 23.624049 24.320461 21.846233 27.917981
##  [99] 20.301852 20.643555 15.746054 25.492756 13.968225 22.909311 24.318449
## [106] 24.915260 13.643699 19.994686 15.195449 26.296047 15.666966 21.245221
## [113] 19.569159 17.535578 18.655005 18.003799 15.278512 25.065942 17.956480
## [120] 19.094566 25.918779 11.957528  8.637801  9.796960 16.759968 23.588611
```

```
## [127] 19.483000 21.635087 15.112013 21.707215 20.219112 14.892108 23.967139
## [134] 22.508122 12.227648 24.417489 18.961161 29.156870 20.886322 15.970797
## [141] 29.739662 20.605985 18.225254 25.264331 25.815570 16.005836 23.877276
## [148] 10.024963 18.247189 21.904324 21.291069 24.822551 17.543609 20.408194
## [155] 19.191146 26.364085 16.398654 27.705258 24.268308 24.011997 18.286889
## [162] 20.502000 20.759941  7.720937 18.289855 13.425092 10.449740 22.897609
## [169] 11.424111 18.008104 12.106258 29.224016 24.878798 18.348997 25.967005
## [176] 18.332404 20.831839 17.848781 29.829793 22.796843 16.792473 19.765127
## [183] 25.670720 21.446156 22.311981 25.909343 12.951362 16.881968 16.714550
## [190] 23.539119 11.860367 17.759416 25.595072 20.929983 18.189702 21.235010
## [197] 21.792841 20.084665 18.823614 18.046567 14.397235 19.688380 23.570709
## [204] 23.865225 29.918528 24.406361 14.237184 22.776337 21.341790 26.339660
## [211] 10.458393 22.393845 19.857631 20.503727 24.287733 18.943312 22.360088
## [218] 17.769001 22.963638 27.086934 27.383978 19.378077 11.186062 20.011225
## [225] 20.421306 20.123873 16.757051 14.699989 16.997426 25.316496 26.950245
## [232] 16.010010 24.436744 17.014697 19.326808 18.420359 16.053319 23.706133
## [239] 16.960203 17.791038 21.413060 19.131284 21.676106 16.758948 19.846822
## [246] 23.625464 15.644751 17.846888 19.360970 20.430444 21.425885 17.165518
## [253] 20.647313 22.501415  8.131987 22.629900 23.027207 16.662687 13.146619
## [260] 22.878786 18.374390 24.698385 25.240030 25.074911 11.800556 32.938661
## [267] 18.137899 19.501374 16.269358 21.854434 20.124051 21.433904 14.243816
## [274] 21.266734 22.882063 28.134948 29.386954 19.686578  8.837422 22.873957
## [281] 18.199073 28.421697 20.719975 25.150915 23.607765 18.206521 21.541650
## [288] 24.402666 26.997643 18.332913 17.187078 15.702006 25.105667 22.631820
## [295] 15.938557  9.654270 21.640890 15.197677 31.854181 31.047655 25.756158
## [302] 23.376825 26.070848 16.383444 14.997585 19.382042 15.585808 17.369188
## [309] 25.096706 17.922887  1.170791 17.155014 27.069001 23.556177 13.652370
## [316] 21.720891 23.628847 27.958968 10.075404 25.855987 22.120714 18.718828
## [323] 20.491585 12.733047 24.953492 13.662022 27.808469 17.157667 27.332785
## [330] 24.630074 14.073768 23.067853 10.982841 24.369412 19.572791 14.902804
## [337] 21.482904 20.816468 25.088267 20.569188 22.380650 23.361795 22.759668
## [344] 20.746153 23.401733 21.921528 15.195009 16.372208 20.916453 14.843970
## [351] 21.407211 19.280898 18.462544 17.211914  8.433002 23.148722 27.074151
## [358] 26.620404 12.227012 20.664894 11.855974 25.409050 13.527421 30.044840
## [365] 11.291278 23.139789 29.028464 16.470919 22.869923 25.651288 20.282186
## [372] 19.071078 27.785693 27.328848 33.394464 22.989078 25.742536 26.364270
## [379] 14.882278  6.829995 22.850895 26.039598 16.383468 27.069286 22.302764
## [386] 26.594798 19.332438 22.548778 13.598647 18.538093 17.906088 17.887869
## [393]  9.291906 19.538152 23.066781 15.726330 17.387226 24.501433 19.691224
## [400] 21.471372 21.236909 21.852862 23.971906 20.056038 18.163681 12.476338
## [407] 21.484611 23.069371 22.724376 13.463097 26.002377 13.359892 13.461443
## [414] 26.277843 16.703317 11.995022 18.200704 22.143099 23.016301 20.308807
## [421] 16.786354 16.284784 20.592543 22.569539 26.864979 22.493086 20.547156
## [428] 22.833922 22.192523 15.477895 10.888632 15.046575 19.113939 18.600905
## [435] 21.605709 18.059182 18.899381 17.275336 28.615540 17.848395 15.179188
## [442] 24.108048 13.244548 17.304603 13.810117 22.033193 26.748135 19.613615
## [449] 17.343245 14.698256 26.427497 20.474908 20.950286 16.373536 15.177121
## [456] 22.171438 23.105219 14.434420 24.111968 18.637988 16.309860 22.078416
## [463] 14.326573 14.996633 27.844257 24.387048 17.885876 13.264569 12.797996
## [470] 23.638467 20.653242 12.110263 21.264221 20.953348 17.391008 30.380144
## [477] 17.609096 12.131751 22.962459 15.832215 18.030871 16.583865 18.612577
## [484] 17.608240 17.041493 14.115381 24.629083 20.680316 18.383487 32.447726
## [491] 21.233517 16.217085 27.485979 23.417306 25.014476 22.583868 19.018689
## [498] 22.106389 19.156101 17.624887 17.206850 24.954572 19.785041 28.327314
```

```
## [505] 18.644835 33.078087 21.449801 19.913395 17.812906 22.002773 29.199318
## [512] 20.217022 28.527843 21.574495 23.647530 23.398910 17.401333 22.373138
## [519] 24.356303 15.017998 22.769066 13.516303 29.031558 25.098124 28.984213
## [526] 14.775109 17.363140 18.410903 14.730245 20.659188 27.608780 21.289362
## [533] 15.775631 14.722697 20.695229 22.730408 14.847834 19.295484 29.740027
## [540] 12.031009 27.024285 19.988500 21.524522 24.606876 20.607896 21.899126
## [547] 28.353481 18.843460 16.011885 21.110990 21.829046 14.901866 23.321739
## [554] 23.482035 27.240804 21.008496 19.307111 21.082862 28.773291 20.680135
## [561] 21.410048 13.048723 18.648300 19.450322 22.980849 23.710506 21.440687
## [568] 13.658988 29.806489 27.440402 23.159076 14.612937 23.209111 21.213508
## [575] 28.818414 15.021691 25.418499 16.327475 18.757141 24.962601 26.075442
## [582] 23.030045 21.307075 18.121257 22.289924 19.435407 17.899642 18.632379
## [589] 12.083140 16.845930 21.295764 16.311188 15.815029 12.579327 15.761356
## [596]  9.472035 20.938142 14.918759 29.918865 23.377963 20.767522 21.616750
## [603] 24.603996 14.600146 30.887465 13.853265 20.988449 19.537453 18.904812
## [610] 22.228428 22.315801 23.248445 19.186116 24.846797 12.054714 19.241175
## [617] 19.341746 24.560781 21.732071 17.961687 18.302763 26.185461 26.872988
## [624] 18.637369 22.566695 22.123893 21.325301 18.575912 10.793190 22.794579
## [631] 17.508246 16.506955 26.882558 11.322454 25.447796 21.989159 19.793819
## [638] 23.676473 17.073013 21.970794 16.753311 23.271942 20.760575 15.953503
## [645] 20.200420 19.432144 27.926602 24.719323 27.301009 20.253710 20.006453
## [652] 19.464992 26.257350 14.039137 25.752497 16.951943 26.604153 11.180788
## [659] 17.827868 24.251802 26.136958 25.187137 25.381131 23.854471 18.131550
## [666] 17.938737 18.957664 18.487598 18.353489 21.534200 11.476370 17.734636
## [673] 17.779962 20.171619 23.964638 13.142208 10.711967 16.833719 11.933753
## [680] 16.560042 20.031118 20.737853 23.731905 12.996919 18.485362 24.128915
## [687] 22.294998 25.978955 25.328845 20.900819 16.046725 14.237595  7.494684
## [694] 17.230672 18.264583 21.100065 20.373770 24.401796 22.004184 14.746003
## [701] 20.318366 32.653974 15.676828 13.863713 17.571365 18.821020 16.315096
## [708] 21.985329 18.846163 19.617798 22.523140 21.135111 17.494425 20.752086
## [715] 30.857022 14.942582 13.887601 19.997827 14.893752 12.413396 20.396740
## [722] 12.577147 20.742577 16.403184 18.935047 22.175514 21.818479 16.685075
## [729] 13.304455 11.859573  9.378147 14.673224 16.876922 26.473696 20.409547
## [736] 22.402078 20.371332 27.920927 21.796235 20.306652 21.760297 20.440328
## [743] 23.147862 26.931660 13.504391 23.401772 14.625028 24.556627 22.792690
## [750] 22.721488  6.612220 14.038805 16.372703 20.583611 11.758782 24.929121
## [757] 23.419690 25.691399 34.951449 25.408914 22.708571 15.712986 12.601618
## [764]  5.910553 22.090713 31.255166 15.543757 23.529827 12.409711 26.003785
## [771] 26.432312 19.781425 18.319312 20.324147 17.655109 20.452763 20.574342
## [778] 16.560337 24.340410 13.236631 16.347989 18.117561 20.467637 21.100524
## [785] 23.926957 23.235751 21.445166 20.200458  8.326485 14.992388 19.947204
## [792] 23.084068 11.630823 21.339664 24.411042 16.287421  9.936137 23.267238
## [799] 20.967550 22.063014
```

Compute the bin widths (h) and number of bins (k) according to each of the following formula:

i) Sturges' formula

```r
# Compute the number of bins (k)
sturge_k <- ceiling(log2(length(rand_data)))+1
sturge_k #11 bins
```

```
## [1] 11
```

```r
# Compute the bin widths (h)
sturge_h <- (max(rand_data) - min(rand_data))/sturge_k
sturge_h #bin width is 3.070969
```

```
## [1] 3.070969
```

ii) Scott's normal reference rule (uses standard deviation)

```r
#For this calculation we need the bin width first and then the number of bins
# Compute the bin widths (h)
scott_h <- 3.49*sd(rand_data)/(length(rand_data)^1/3)
scott_h #bin width is 0.06515691
```

```
## [1] 0.06515691
```

```r
# Compute the number of bins (k)
scott_k <- ceiling((max(rand_data) - min(rand_data))/scott_h)
scott_k #519 bins
```

```
## [1] 519
```

iii) Freedman-Diaconis' choice (uses IQR)

```r
#For this calculation we need the bin width first and then the number of bins
# Compute the bin widths (h)
dia_h <- 2*IQR(rand_data)/(length(rand_data)^1/3)
dia_h #bin width is 0.04983888
```

```
## [1] 0.04983888
```

```r
# Compute the number of bins (k)
dia_k <- ceiling((max(rand_data) - min(rand_data))/dia_h)
dia_k #678
```

```
## [1] 678
```

c) Repeat part (b) with some outliers (create a new dataset out_data):

```r
set.seed(123456)
out_data <- c(rand_data, runif(10, min=40, max=60))
out_data # Taking a look at the values
```

```
##   [1] 24.168666 18.619761 18.224991 20.437437 31.261279 24.172301 26.562078
##   [8] 32.513227 25.841159 17.869172 15.019351 14.430250 19.721342 25.872162
##  [15] 25.266093 20.288030 16.324786 24.652642 28.341055 22.798439 16.230126
##  [22] 26.282771 20.192463 20.947699 22.312975 17.863185 20.082930 23.524396
##  [29] 24.859247 16.897542 15.720665 20.347792 14.769009  6.255658 14.350702
##  [36] 15.691576 27.800369 25.075442 25.219972 14.420474 14.643478 24.839105
##  [43] 20.855163 15.518749 20.791446 17.490258 15.170387 19.431329 25.429749
##  [50] 13.941759 11.161390 17.541540 21.607329 27.303304 27.686213 18.302157
##  [57] 14.612756 12.546174 18.736267 19.390387 16.755083 21.565379 20.614053
##  [64] 15.815798 23.002109 18.770327 19.076677 20.115784 17.576742 16.310468
##  [71] 26.512538 22.822497 10.974778 22.191695 12.794555 24.764146 16.385088
##  [78] 20.243802 12.182339  8.488168 25.789818 16.476323 12.130889 22.591758
##  [85] 14.675347 20.235272 24.240237 22.163389 22.614027 18.730570 17.515740
##  [92] 26.300506 22.824901 18.276811 23.624049 24.320461 21.846233 27.917981
##  [99] 20.301852 20.643555 15.746054 25.492756 13.968225 22.909311 24.318449
## [106] 24.915260 13.643699 19.994686 15.195449 26.296047 15.666966 21.245221
## [113] 19.569159 17.535578 18.655005 18.003799 15.278512 25.065942 17.956480
## [120] 19.094566 25.918779 11.957528  8.637801  9.796960 16.759968 23.588611
## [127] 19.483000 21.635087 15.112013 21.707215 20.219112 14.892108 23.967139
## [134] 22.508122 12.227648 24.417489 18.961161 29.156870 20.886322 15.970797
## [141] 29.739662 20.605985 18.225254 25.264331 25.815570 16.005836 23.877276
## [148] 10.024963 18.247189 21.904324 21.291069 24.822551 17.543609 20.408194
## [155] 19.191146 26.364085 16.398654 27.705258 24.268308 24.011997 18.286889
## [162] 20.502000 20.759941  7.720937 18.289855 13.425092 10.449740 22.897609
## [169] 11.424111 18.008104 12.106258 29.224016 24.878798 18.348997 25.967005
## [176] 18.332404 20.831839 17.848781 29.829793 22.796843 16.792473 19.765127
## [183] 25.670720 21.446156 22.311981 25.909343 12.951362 16.881968 16.714550
## [190] 23.539119 11.860367 17.759416 25.595072 20.929983 18.189702 21.235010
## [197] 21.792841 20.084665 18.823614 18.046567 14.397235 19.688380 23.570709
## [204] 23.865225 29.918528 24.406361 14.237184 22.776337 21.341790 26.339660
## [211] 10.458393 22.393845 19.857631 20.503727 24.287733 18.943312 22.360088
## [218] 17.769001 22.963638 27.086934 27.383978 19.378077 11.186062 20.011225
## [225] 20.421306 20.123873 16.757051 14.699989 16.997426 25.316496 26.950245
## [232] 16.010010 24.436744 17.014697 19.326808 18.420359 16.053319 23.706133
## [239] 16.960203 17.791038 21.413060 19.131284 21.676106 16.758948 19.846822
## [246] 23.625464 15.644751 17.846888 19.360970 20.430444 21.425885 17.165518
## [253] 20.647313 22.501415  8.131987 22.629900 23.027207 16.662687 13.146619
## [260] 22.878786 18.374390 24.698385 25.240030 25.074911 11.800556 32.938661
## [267] 18.137899 19.501374 16.269358 21.854434 20.124051 21.433904 14.243816
## [274] 21.266734 22.882063 28.134948 29.386954 19.686578  8.837422 22.873957
## [281] 18.199073 28.421697 20.719975 25.150915 23.607765 18.206521 21.541650
## [288] 24.402666 26.997643 18.332913 17.187078 15.702006 25.105667 22.631820
## [295] 15.938557  9.654270 21.640890 15.197677 31.854181 31.047655 25.756158
## [302] 23.376825 26.070848 16.383444 14.997585 19.382042 15.585808 17.369188
## [309] 25.096706 17.922887  1.170791 17.155014 27.069001 23.556177 13.652370
## [316] 21.720891 23.628847 27.958968 10.075404 25.855987 22.120714 18.718828
## [323] 20.491585 12.733047 24.953492 13.662022 27.808469 17.157667 27.332785
## [330] 24.630074 14.073768 23.067853 10.982841 24.369412 19.572791 14.902804
## [337] 21.482904 20.816468 25.088267 20.569188 22.380650 23.361795 22.759668
## [344] 20.746153 23.401733 21.921528 15.195009 16.372208 20.916453 14.843970
## [351] 21.407211 19.280898 18.462544 17.211914  8.433002 23.148722 27.074151
## [358] 26.620404 12.227012 20.664894 11.855974 25.409050 13.527421 30.044840
## [365] 11.291278 23.139789 29.028464 16.470919 22.869923 25.651288 20.282186
## [372] 19.071078 27.785693 27.328848 33.394464 22.989078 25.742536 26.364270
```

```
## [379] 14.882278  6.829995 22.850895 26.039598 16.383468 27.069286 22.302764
## [386] 26.594798 19.332438 22.548778 13.598647 18.538093 17.906088 17.887869
## [393]  9.291906 19.538152 23.066781 15.726330 17.387226 24.501433 19.691224
## [400] 21.471372 21.236909 21.852862 23.971906 20.056038 18.163681 12.476338
## [407] 21.484611 23.069371 22.724376 13.463097 26.002377 13.359892 13.461443
## [414] 26.277843 16.703317 11.995022 18.200704 22.143099 23.016301 20.308807
## [421] 16.786354 16.284784 20.592543 22.569539 26.864979 22.493086 20.547156
## [428] 22.833922 22.192523 15.477895 10.888632 15.046575 19.113939 18.600905
## [435] 21.605709 18.059182 18.899381 17.275336 28.615540 17.848395 15.179188
## [442] 24.108048 13.244548 17.304603 13.810117 22.033193 26.748135 19.613615
## [449] 17.343245 14.698256 26.427497 20.474908 20.950286 16.373536 15.177121
## [456] 22.171438 23.105219 14.434420 24.111968 18.637988 16.309860 22.078416
## [463] 14.326573 14.996633 27.844257 24.387048 17.885876 13.264569 12.797996
## [470] 23.638467 20.653242 12.110263 21.264221 20.953348 17.391008 30.380144
## [477] 17.609096 12.131751 22.962459 15.832215 18.030871 16.583865 18.612577
## [484] 17.608240 17.041493 14.115381 24.629083 20.680316 18.383487 32.447726
## [491] 21.233517 16.217085 27.485979 23.417306 25.014476 22.583868 19.018689
## [498] 22.106389 19.156101 17.624887 17.206850 24.954572 19.785041 28.327314
## [505] 18.644835 33.078087 21.449801 19.913395 17.812906 22.002773 29.199318
## [512] 20.217022 28.527843 21.574495 23.647530 23.398910 17.401333 22.373138
## [519] 24.356303 15.017998 22.769066 13.516303 29.031558 25.098124 28.984213
## [526] 14.775109 17.363140 18.410903 14.730245 20.659188 27.608780 21.289362
## [533] 15.775631 14.722697 20.695229 22.730408 14.847834 19.295484 29.740027
## [540] 12.031009 27.024285 19.988500 21.524522 24.606876 20.607896 21.899126
## [547] 28.353481 18.843460 16.011885 21.110990 21.829046 14.901866 23.321739
## [554] 23.482035 27.240804 21.008496 19.307111 21.082862 28.773291 20.680135
## [561] 21.410048 13.048723 18.648300 19.450322 22.980849 23.710506 21.440687
## [568] 13.658988 29.806489 27.440402 23.159076 14.612937 23.209111 21.213508
## [575] 28.818414 15.021691 25.418499 16.327475 18.757141 24.962601 26.075442
## [582] 23.030045 21.307075 18.121257 22.289924 19.435407 17.899642 18.632379
## [589] 12.083140 16.845930 21.295764 16.311188 15.815029 12.579327 15.761356
## [596]  9.472035 20.938142 14.918759 29.918865 23.377963 20.767522 21.616750
## [603] 24.603996 14.600146 30.887465 13.853265 20.988449 19.537453 18.904812
## [610] 22.228428 22.315801 23.248445 19.186116 24.846797 12.054714 19.241175
## [617] 19.341746 24.560781 21.732071 17.961687 18.302763 26.185461 26.872988
## [624] 18.637369 22.566695 22.123893 21.325301 18.575912 10.793190 22.794579
## [631] 17.508246 16.506955 26.882558 11.322454 25.447796 21.989159 19.793819
## [638] 23.676473 17.073013 21.970794 16.753311 23.271942 20.760575 15.953503
## [645] 20.200420 19.432144 27.926602 24.719323 27.301009 20.253710 20.006453
## [652] 19.464992 26.257350 14.039137 25.752497 16.951943 26.604153 11.180788
## [659] 17.827868 24.251802 26.136958 25.187137 25.381131 23.854471 18.131550
## [666] 17.938737 18.957664 18.487598 18.353489 21.534200 11.476370 17.734636
## [673] 17.779962 20.171619 23.964638 13.142208 10.711967 16.833719 11.933753
## [680] 16.560042 20.031118 20.737853 23.731905 12.996919 18.485362 24.128915
## [687] 22.294998 25.978955 25.328845 20.900819 16.046725 14.237595  7.494684
## [694] 17.230672 18.264583 21.100065 20.373770 24.401796 22.004184 14.746003
## [701] 20.318366 32.653974 15.676828 13.863713 17.571365 18.821020 16.315096
## [708] 21.985329 18.846163 19.617798 22.523140 21.135111 17.494425 20.752086
## [715] 30.857022 14.942582 13.887601 19.997827 14.893752 12.413396 20.396740
## [722] 12.577147 20.742577 16.403184 18.935047 22.175514 21.818479 16.685075
## [729] 13.304455 11.859573  9.378147 14.673224 16.876922 26.473696 20.409547
## [736] 22.402078 20.371332 27.920927 21.796235 20.306652 21.760297 20.440328
## [743] 23.147862 26.931660 13.504391 23.401772 14.625028 24.556627 22.792690
## [750] 22.721488  6.612220 14.038805 16.372703 20.583611 11.758782 24.929121
```

```
## [757] 23.419690 25.691399 34.951449 25.408914 22.708571 15.712986 12.601618
## [764]  5.910553 22.090713 31.255166 15.543757 23.529827 12.409711 26.003785
## [771] 26.432312 19.781425 18.319312 20.324147 17.655109 20.452763 20.574342
## [778] 16.560337 24.340410 13.236631 16.347989 18.117561 20.467637 21.100524
## [785] 23.926957 23.235751 21.445166 20.200458  8.326485 14.992388 19.947204
## [792] 23.084068 11.630823 21.339664 24.411042 16.287421  9.936137 23.267238
## [799] 20.967550 22.063014 55.955686 55.071302 47.825114 46.831134 47.225882
## [806] 43.966895 50.697159 41.930525 59.756939 43.351390
```

i) Sturges formula

```r
# Compute the number of bins (k)
out_sturge_k <- ceiling(log2(length(out_data))) +1
out_sturge_k #11 bins
```

```
## [1] 11
```

```r
# Compute the bin widths (h)
out_sturge_h <- (max(out_data) - min(out_data))/out_sturge_k
out_sturge_h #bin width is 5.326013
```

```
## [1] 5.326013
```

ii) Scott's normal reference rule (uses standard deviation)

```r
#For this calculation we need the bin width first and then the number of bins
# Compute the bin widths (h)
out_scott_h <- 3.49*sd(out_data)/(length(out_data)^1/3)
out_scott_h #bin width is 0.07675489
```

```
## [1] 0.07675489
```

```r
# Compute the number of bins (k)
out_scott_k <- ceiling((max(out_data) - min(out_data))/out_scott_h)
out_scott_k #764 bins
```

```
## [1] 764
```

iii) Freedman-Diaconis' choice (uses IQR)

```r
#For this calculation we need the bin width first and then the number of bins
# Compute the bin widths (h)
out_dia_h <- 2*IQR(out_data)/(length(out_data)^1/3)
out_dia_h #bin width is 0.05026184
```

```
## [1] 0.05026184
```

```r
# Compute the number of bins (k)
out_dia_k <- ceiling((max(out_data) - min(out_data))/out_dia_h)
out_dia_k #1166 bins
```

```
## [1] 1166
```

Which of the methods does the bin width (h) change the least when outliers are added and why?

```r
## I think Freedman-Diaconis' choice is the least sensitive to outliers
## in terms of bin width because it uses the IQR instead of the standard deviation.
```