

Data File Used in this Analysis:

```
# Math 3070 - 1    County populations    Treibergs
#
# Source: C Goodall, K Kafadar and J Tukey, "Computing and Using Rural
# versus Urban Measures in Statistical Applications." American Statistician
# 1989 as given in Navidi, Statistics for Engineers and Scientists, McGraw
# Hill 2006. The data gives population frequencies of US counties.
# Populations are in log2 scale. the first interval 6.0 -< 12.4
# gives counties with at least 2^6=64 people but less than 2^12.4=5404.
#
#
# x = log_2 Population
# y = Number of Counties
#
xleft symbol xright number
6.0 -< 12.4 305
12.4 -< 13.1 294
13.1 -< 13.6 331
13.6 -< 14.0 286
14.0 -< 14.4 306
14.4 -< 14.8 273
14.8 -< 15.3 334
15.3 -< 16.0 326
16.0 -< 17.0 290
17.0 -< 23.0 323
```

R Session:

R version 2.10.1 (2009-12-14)
Copyright (C) 2009 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.31 (5537) powerpc-apple-darwin9.8.0]

```

> tt <- read.table("M3073CountyData.txt",header=TRUE)
> tt
  xleft symbol xright number
1   6.0    -<  12.4   305
2  12.4    -<  13.1   294
3  13.1    -<  13.6   331
4  13.6    -<  14.0   286
5  14.0    -<  14.4   306
6  14.4    -<  14.8   273
7  14.8    -<  15.3   334
8  15.3    -<  16.0   326
9  16.0    -<  17.0   290
10 17.0    -<  23.0   323

> attach(tt)

> # Note that the frequencies are given in this data. However, the class intervals
> # do not have uniform width. Thus, to make the histogram, we need to make bar
> # heights such that the AREAS of the bars are proportional to the number of counties
> # in each class. The height of the bars has little meaning so is suppressed.
>
> # We illustrate making the histogram in three ways. The first is to use the
> # barplot function. The second is to draw bars using the plot function.
> # The third is to use hist() to compute graphics parameters, to modify them
> # with count data (as far as I know, there is no direct way of inputting
> # counts into hist()), and then use plot.hist to do the graphics for us.

> ##### PREPARE HEIGHTS AND WIDTHS #####
> # wi is the vector of class widths
> wi <- xright-xleft

> # The area is proportional to the number, so the height is proportional to number
> # divided by width
> y <- number/wi

> # cllab is just a character vector containing class labels.
> cllab<-paste(xleft,xright,sep=" -< ")
> cllab
[1] "6 -< 12.4"    "12.4 -< 13.1" "13.1 -< 13.6" "13.6 -< 14"
[5] "14 -< 14.4"    "14.4 -< 14.8" "14.8 -< 15.3" "15.3 -< 16"
[9] "16 -< 17"     "17 -< 23"

> # the total number of counties in the US
> su<-sum(number);su
[1] 3068
> yy <- y/su

> # We ask that labels be printed vertically and margins increased.
> opar <- par(las=3,mai=c(3,1,1,0.5))

```

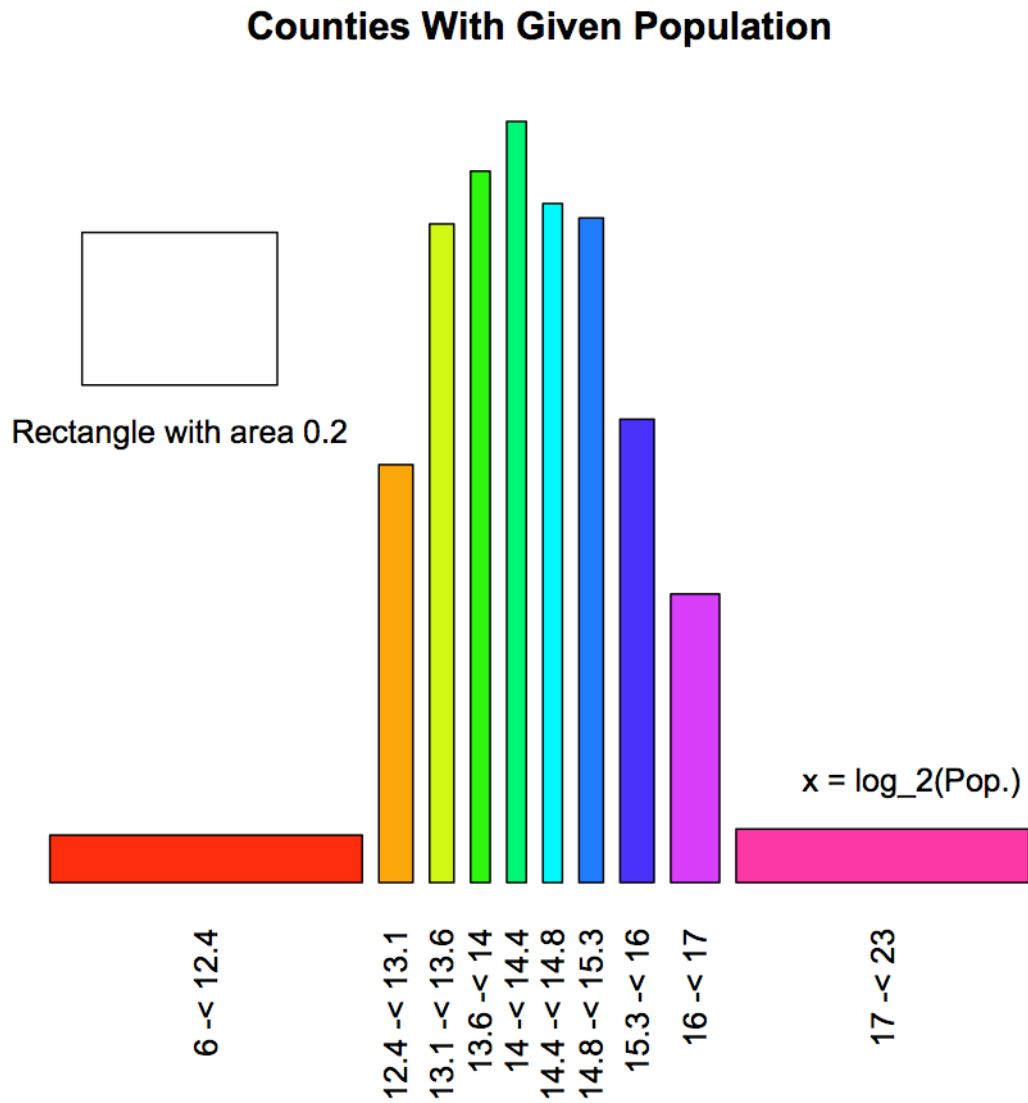
```

> barplot(y,width=wi,main="Counties With Given Population", xlab="",
  names.arg=c1lab,col=rainbow(10),axes=FALSE)

> # A rectangle that has 0.2 of the total area and has width = 4 units
> # must have height su/20 units.

> dh<-su/20;lines(c(1,1,5,5,1),c(500,500+dh,500+dh,500,500))
> text(3,450,"Rectangle with area 0.2")
> text(18,100,"x = log_2(Pop.)")
> par(opar)

```

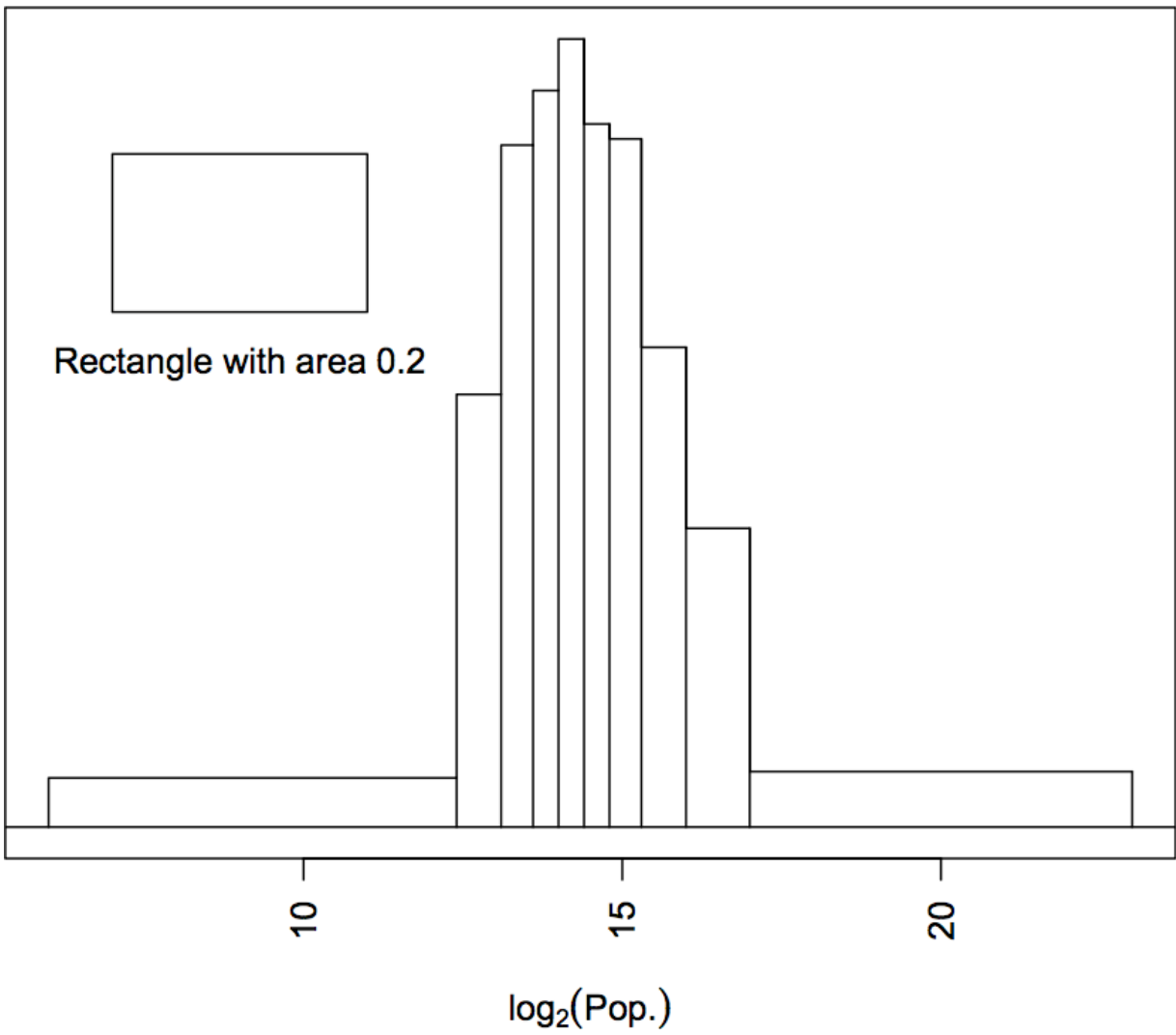


```

> ##### HISTOGRAM "BY HAND" #####
> # Barplot by hand. We use a feature of plot: type = "S" means that the
> # plotted function is piecewise constant. yaxt = "n" suppresses vertical axis.
> # type = "h" means plot as points with vertical lines from point to x-axis.
>
> xx<-c(xleft[1],xright)
> yy <- c(0,y/su)
> plot(yy~xx,type="S",axes=TRUE, ylab="",
+ main="Histogram of Number of Counties with Given Population",
+ xlab=expression(log[2](Pop.)),yaxt="n")
> points(xx,yy,type="h")
> abline(h=0)
> dh<-su/20;lines(c(7,7,11,11,7),c(500,500+dh,500+dh,500,500)/su)
> text(9,450/su,"Rectangle with area 0.2")

```

Histogram of Number of Counties with Given Population



```

> ##### MODIFY hist() PARAMETERS AND REPRINT #####
>
> # Make a vector of breaks.
> br <- c(xleft[1],xright)
> br
[1] 6.0 12.4 13.1 13.6 14.0 14.4 14.8 15.3 16.0 17.0 23.0
> # Sort the histogram list.
> hi <- hist(xleft,breaks=br,freq=FALSE)

> hi
$breaks
[1] 6.0 12.4 13.1 13.6 14.0 14.4 14.8 15.3 16.0 17.0 23.0

$counts
[1] 305 294 331 286 306 273 334 326 290 323

$intensities
[1] 0.0312500 0.1428571 0.2000000 0.2500000 0.2500000 0.2500000 0.2000000 0.1428571
[9] 0.1000000 0.0000000

$density
[1] 0.0312500 0.1428571 0.2000000 0.2500000 0.2500000 0.2500000 0.2000000 0.1428571
[9] 0.1000000 0.0000000

$mids
[1] 9.20 12.75 13.35 13.80 14.20 14.60 15.05 15.65 16.50 20.00

$xname
[1] "xleft"

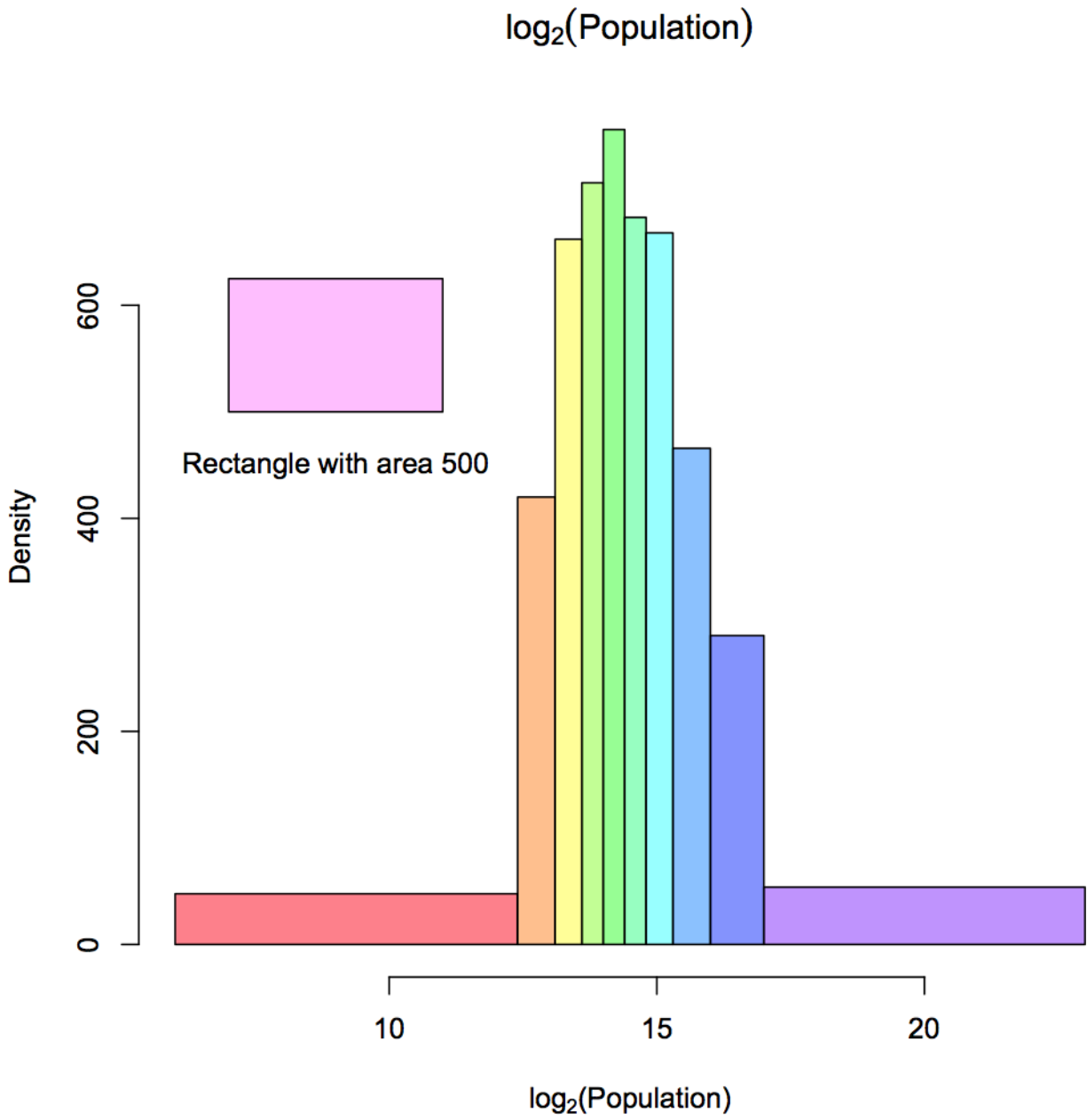
$equidist
[1] FALSE

attr("class")
[1] "histogram"

> # Modify the counts as given by number, and the heights as given by y.

> hi$counts <- number
> hi$density <- y
> hi$xname=expression(paste(log[2],"(Population)"))
> plot(hi,main=expression(log[2](Population)),col=rainbow(12,alpha=.5),freq=FALSE)
> dh<-125;polygon(c(7,7,11,11,7),c(500,500+dh,500+dh,500,500),col=rainbow(12,alpha=.3)[11])
> text(9,450,"Rectangle with area 500")

```



```
> ##### REPLICATE VALUES A COUNT NUMBER OF TIMES #####
> This way is a kludge (quick and dirty workaraound.)
> kludge <- rep(.5*(xleft+xright),number)
>
> hist(kludge, breaks = br, freq=FALSE, main=expression(log[2](Population)), col=heat.colors(12),
+ xlab = expression(paste(log[2],"(Population)")))
> dh<-125
> polygon(c(7,7,11,11,7),c(500,500+dh,500+dh,500,500)/su,col=heat.colors(12)[11])
> text(9,450/su,"Rectangle with area 500")
> # M3073CountyHist6.pdf
```