

Statistical Computing with R: MDS 503 (S13) Fourth Batch, SMS, TU, 2025

Shital Bhandary

Associate Professor

Statistics/Bio-statistics, Demography and Public Health Informatics

Patan Academy of Health Sciences, Lalitpur, Nepal

Faculty, Masters in Medical Research, NHRC/Kathmandu University

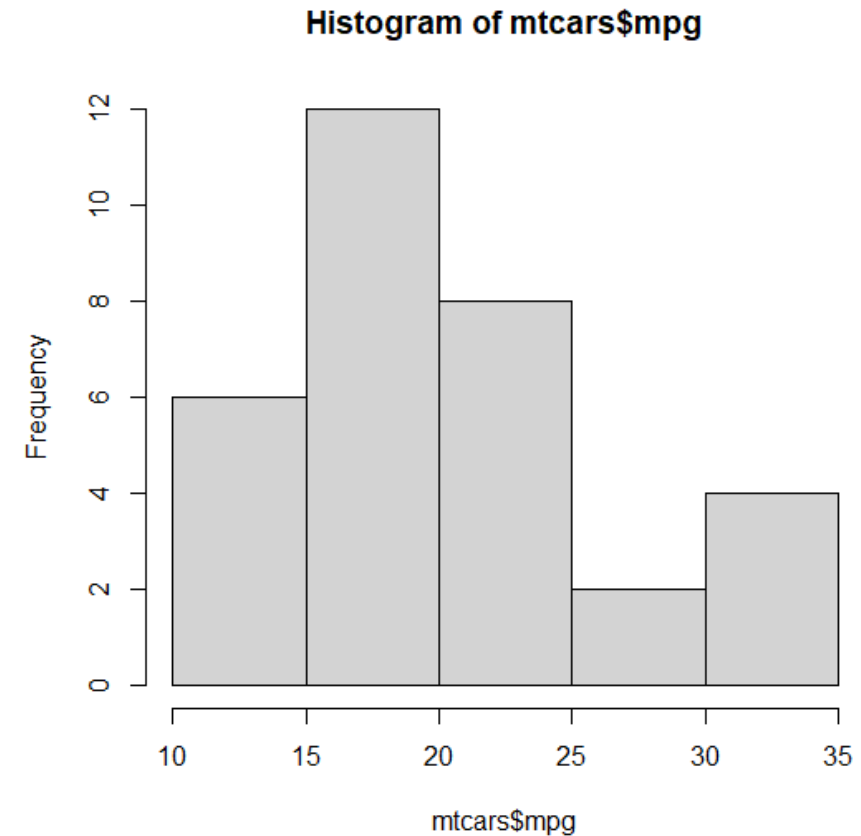
Faculty, FAIMER Fellowship in Health Professions Education, India/USA

Review Preview

- Basic graphics/plots:
 - Plots from raw data
- Special graph:
 - Additional features

Graph from data frame

- Check the structure of in-built “mtcars” data
- Barplot of “mpg” variable
- Histogram of “mpg” variable
- Which one do you prefer?



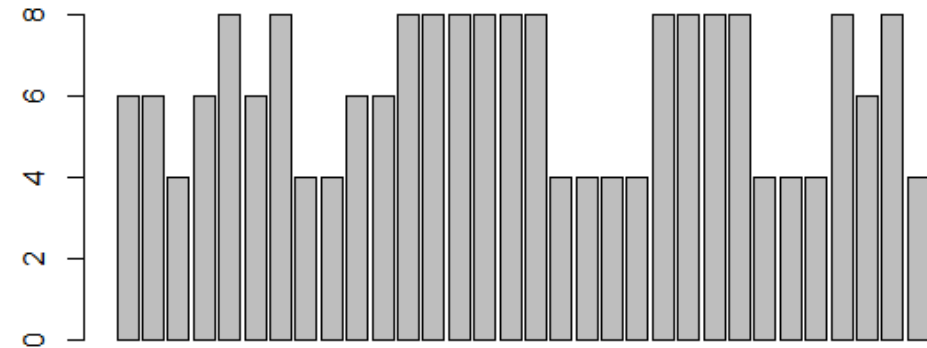
How to get bar diagram of a categorical variable from data frame?

#Define as data frame, if required

- `df <- as.data.frame(mtcars)`

#Bar plot of cylinder data

- `barplot(df$cyl)`
- **This barplot shows the number of cylinders for 50 cars of the dataset**
- **Do we want this?**



How to get bar diagram of a categorical variable from data frame?

#Let's define cyl as factor variable

- `f.cyl <- as.factor(df$cyl)`

#And get bar plot of cylinder data

- `barplot(f.cyl)`

- Did you get the barplot?

- **Why?**

- **Error in `barplot.default(f.cyl)` :
'height' must be a vector or a matrix**

- **This means variable is factor but its frequencies are not found!**

- What to do now?

How to get bar diagram from data frame?

First we **need frequencies** of cars with 4, 6 and 8 cylinders

- `table(df$cyl)`

#Bar **plot of freq. of cylinder** data

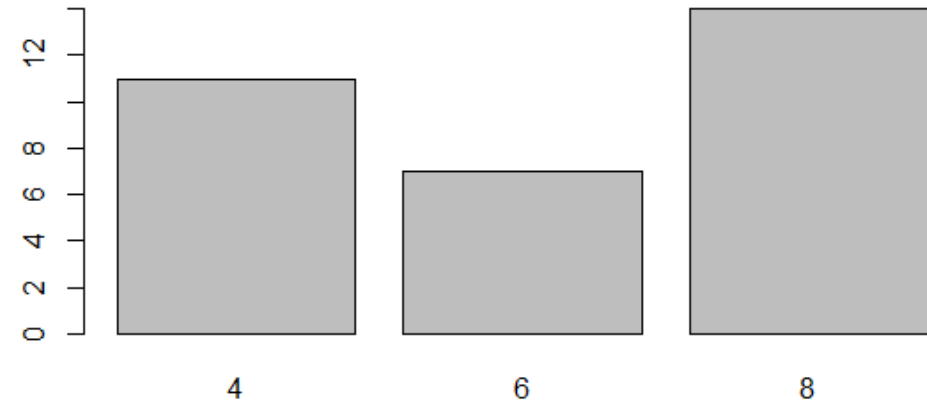
- `barplot(table(df$cyl))`

#We can assign this as object

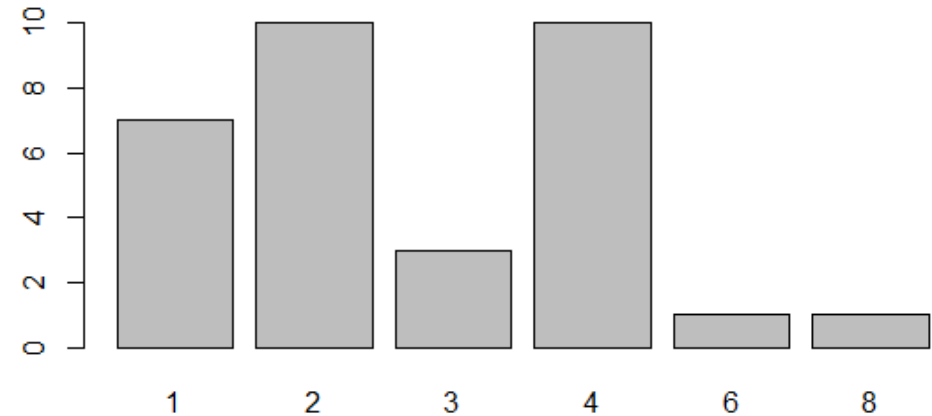
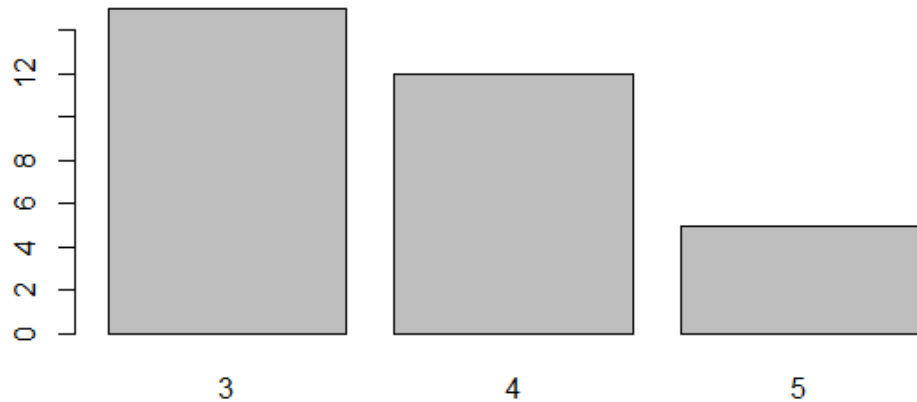
- `bpd <- table(df$cyl)`

#Get the barplot

- `barplot(bpd)`



We can get the barplot of “gear” and “carb” too as they are factors (categorical variables)



Class work: How to get barplot of “mpg” variable?

mpg: miles per gallon (continuous variable)

#MPG – range for class interval

- `range(df$mpg)`
- `R = 33.9 - 10.4` `#23.5`
- `I = round(sqrt(R))` `# 5`

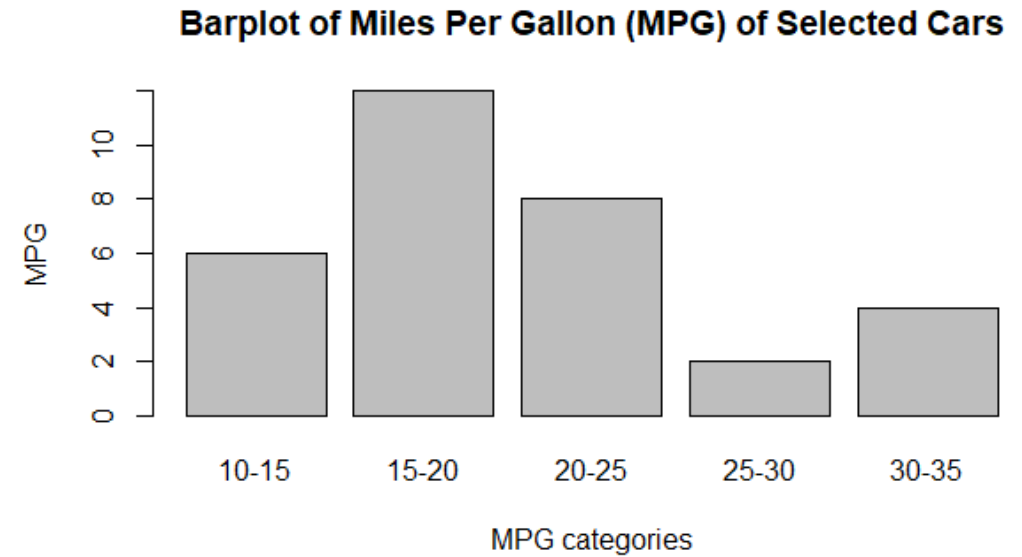
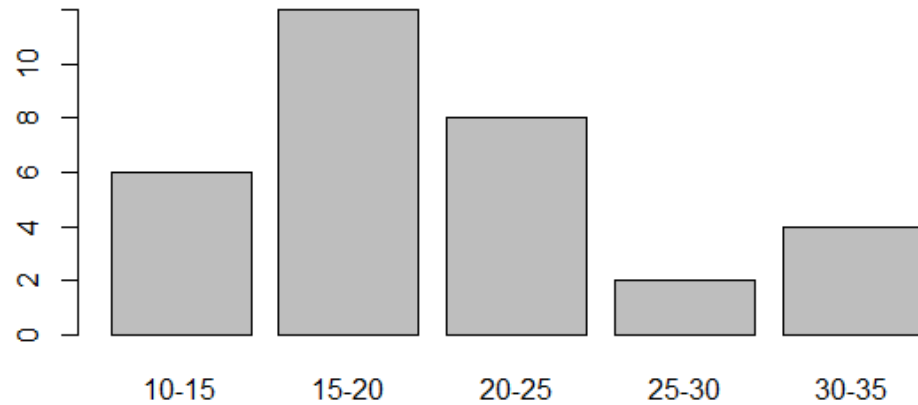
#We need to construct 5 classes with width of 5 (10, 15, 20, 25, 30)

#We need to define the breaks

`breaks = c(10, 15, 20, 25, 30, 35)` or
`breaks = seq(10, 35, by=5)`

- `mpg.bin <- cut(df$mpg, breaks, labels = c("10-15", "15-20", "20-25", "25-30", "30-35"))`
- `mpg.bin`
- `table(mpg.bin)`
- `barplot(table(mpg.bin))`

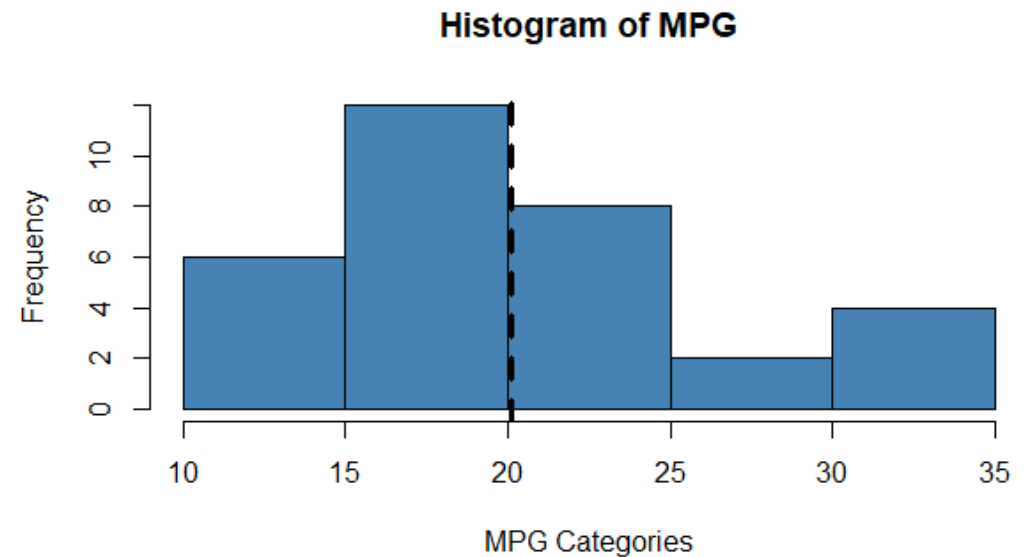
Outputs:



WHAT ARE THE TWO KEY DIFFERENCES BETWEEN BAR DIAGRAM AND HISTOGRAM?

Histogram and abline for mean of “mpg”:

- `hist(df$mpg, col = "steelblue", main = "Histogram of MPG", xlab = "MPG Categories")`
- `abline(v=mean(df$mpg), lwd=3, lty=2)`
- `v` = vertical “abline”
- `h` = horizontal “abline”
- `lwd` = line width (3=3 times wide)
- `lty` =line types (2 = dashed line)



Line types:

- lty = 1 (solid line)
- lty = 2 (dashed line)
- lty = 3 (dotted line)
- lty = 4 (dot and dashed line)
- lty = 5 (long dash line)
- lty = 6 (two dashed line)

6. 'twodash'



5. 'longdash'



4. 'dotdash'



3. 'dotted'



2. 'dashed'



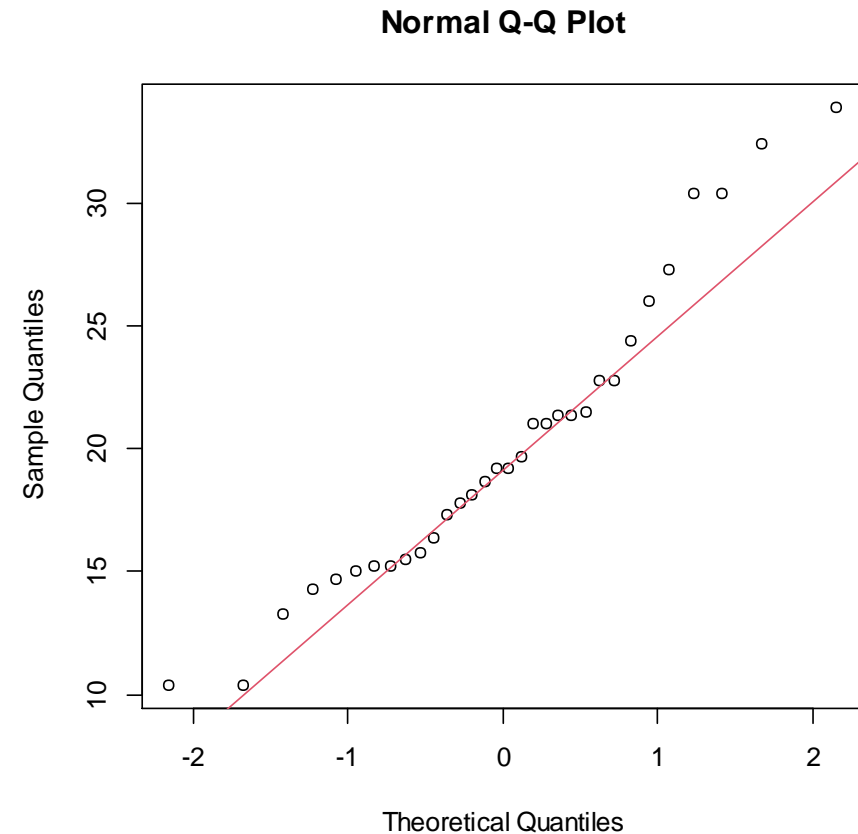
1. 'solid'



0. 'blank'

Can you justify the use of mean for “mpg” variable in the histogram?

- `qqnorm(mtcars$mpg)`
- `> qqline(mtcars$mpg, col=2)`
- Which measure of central tendency is most useful for the mpg variable?
- Which measure of central tendency can be located by histogram “graphically”?

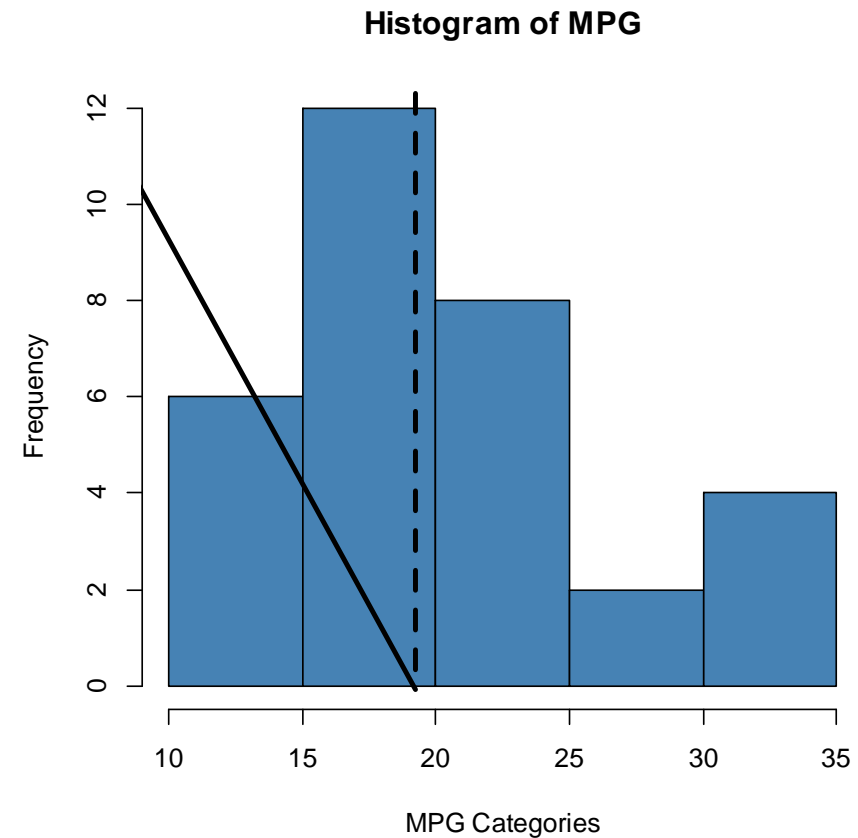


Histogram and abline of median of “mpg”:

- `hist(df$mpg, col = "steelblue", main = "Histogram of MPG", xlab = "MPG Categories")`
- `abline(v=median(df$mpg), lwd=3, lty=2)`
- `v` = vertical “abline”
- `h` = horizontal “abline”

`summary(df$mpg)`

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
10.40	15.43	19.20	20.09	22.80	33.90



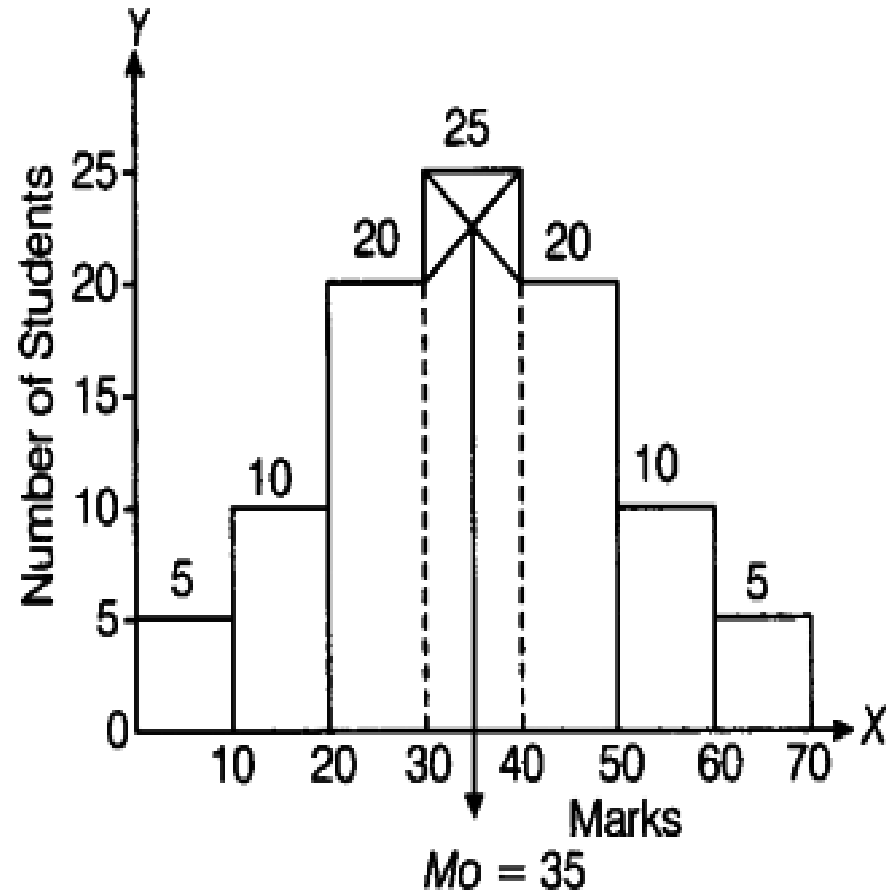
Assignment: Locate median **graphically** for “mpg” variable

- Create a more than cumulative frequency curve
- Create a less than cumulative frequency curve
- Draw both of them in a single plot
- The point of intersection of more than and less than cumulative frequency curve will give the “median” value
- Draw a perpendicular to the x-axis from this point of intersection to find the median
- Compare it with the result of the “median” function of R!

Read this link:

<https://tutors4you.com/index.php/finding-median-graphically/>

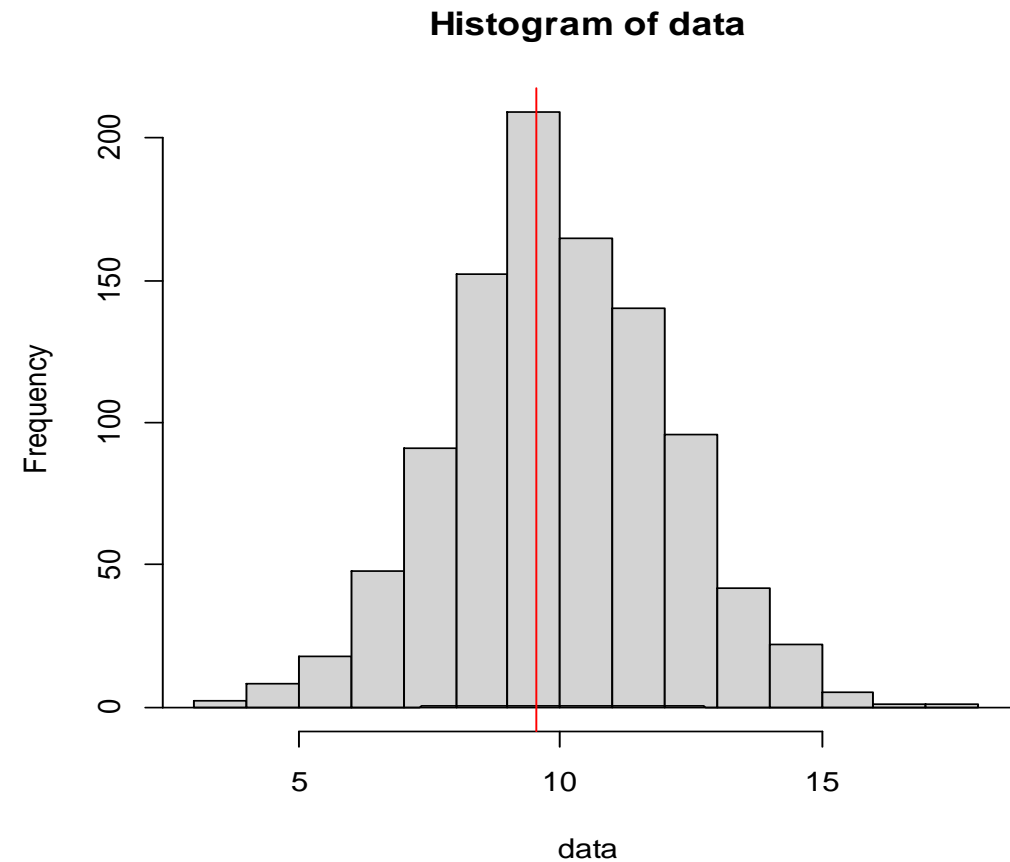
Class work: Locate the mode in the histogram graphically!



- Now do the same for “mpg” variable in R Studio with R Script!
- Check your value with the built-in function of model
- `mode(df$mpg) ??`
- `table(df$mpg) ??` (highest freq?)
- `which.max(df$mpg) ??`

ChatGPT: how to locate mode graphically in r with histogram?

- # Generate some example data
- `data <- rnorm(1000, mean = 10, sd = 2)`
- # Create a histogram of the data
- `hist(data)`
- # Add a density line to the histogram
- `lines(density(data))`
- # Locate the mode
- `density_values <- density(data)$y`
- `max_density <- max(density_values)`
- `mode <- density(data)$x[which.max(density_values)]`
- # Add a vertical line at the mode location
- `abline(v = mode, col = "red")`



I AM NOT CONVINCED! ARE YOU? USE CHATGPT!

How to get mode of a variable with bi-model or multi-model distribution like “mpg”?

x	freq
10.4	2
13.3	1
14.3	1
14.7	1
15.0	1
15.2	2
15.5	1
15.8	1
16.4	1
17.3	1
17.8	1
18.1	1
18.7	1

x	freq
19.2	2
19.7	1
21.0	2
21.4	2
21.5	1
22.8	2
24.4	1
26.0	1
27.3	1
30.4	2
32.4	1
33.9	1

Mode = 3 Median – 2 Mean

```
mode <- 3*median(df$mpg) - 2  
*mean(df$mpg)
```

```
mode  
[1] 17.41875 (How to interpret?)
```

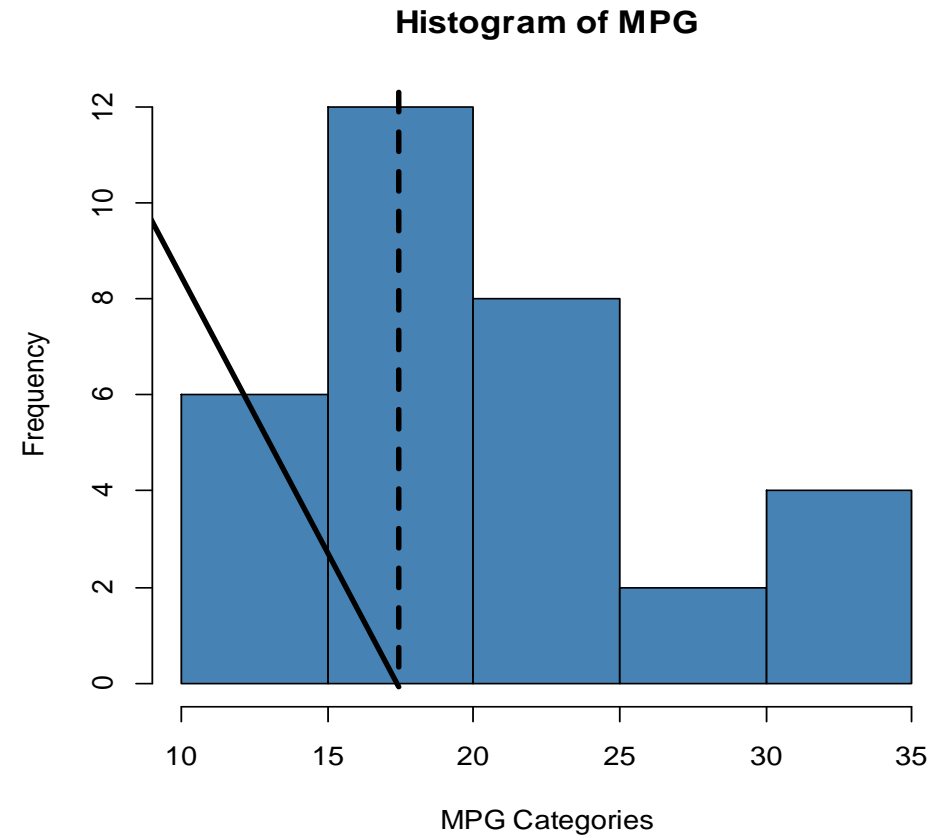
Now show this value as “mode” in the histogram!

Histogram and abline of mode of “mpg”:

- `hist(df$mpg, col = "steelblue", main = "Histogram of MPG", xlab = "MPG Categories")`
- `abline(v=3*median(df$mpg)-2*mean(df$mpg), lwd=3, lty=2)`
- `v` = vertical “abline”
- `h` = horizontal “abline”

`summary(df$mpg)`

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
10.40	15.43	19.20	20.09	22.80	33.90



Assignment: Show the mode, median and mean of “mpg” variable in a single histogram

- With ablines of different colors
- With a legend for each color

Scatterplot with horizontal “abline”:

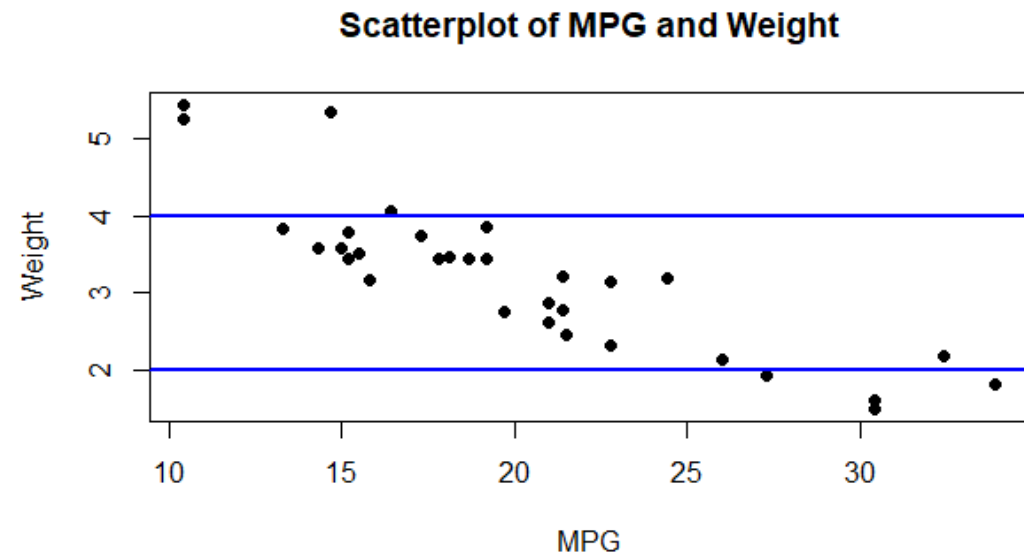
```
#Scatterplot with abline
```

```
plot(df$mpg, df$wt, pch=16, main =  
"Scatterplot of MPG and Weight", xlab = "MPG", ylab =  
"Weight")
```

```
abline(h=2, col = "blue", lwd=2)
```

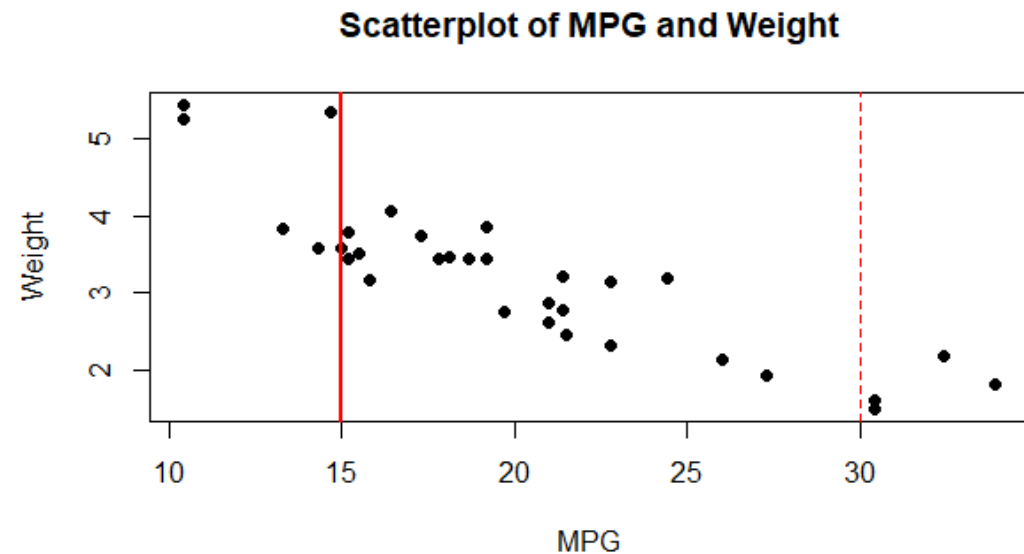
```
abline(h=4, col = "blue", lwd=2)
```

Here, h = horizontal line in y-axis
and lwd = line width parameter



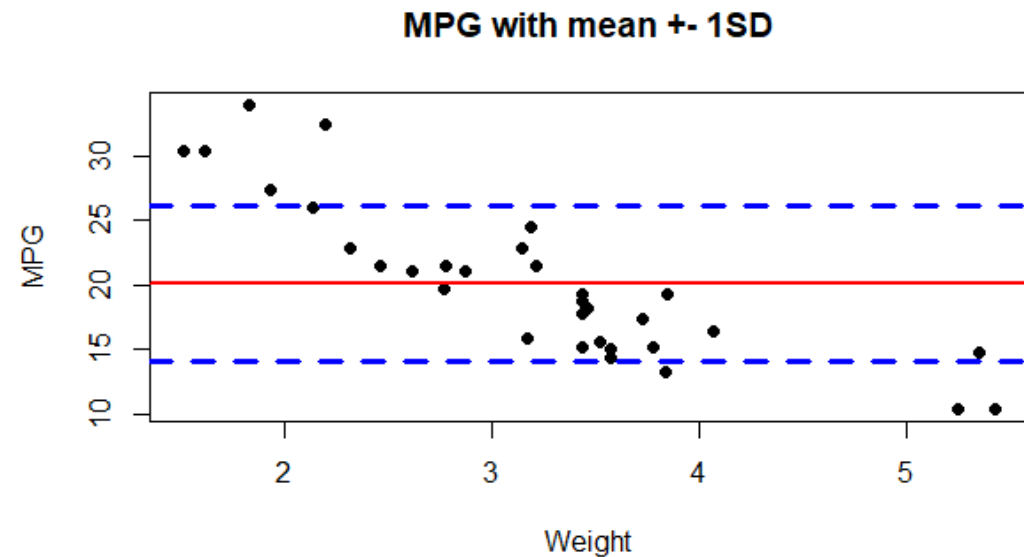
Scatterplot with vertical “abline”:

- `plot(dfmpg, dfwt, pch=16, main = "Scatterplot of MPG and Weight", xlab = "MPG", ylab = "Weight")`
- `abline(v=15, col = "red", lwd=2)`
- `abline(v=30, col = "red", lty=2)`
- Here, v=Vertical line at x-axis and lty = line type parameter



Scatterplot with mean ± 1 *sd of y-variable:

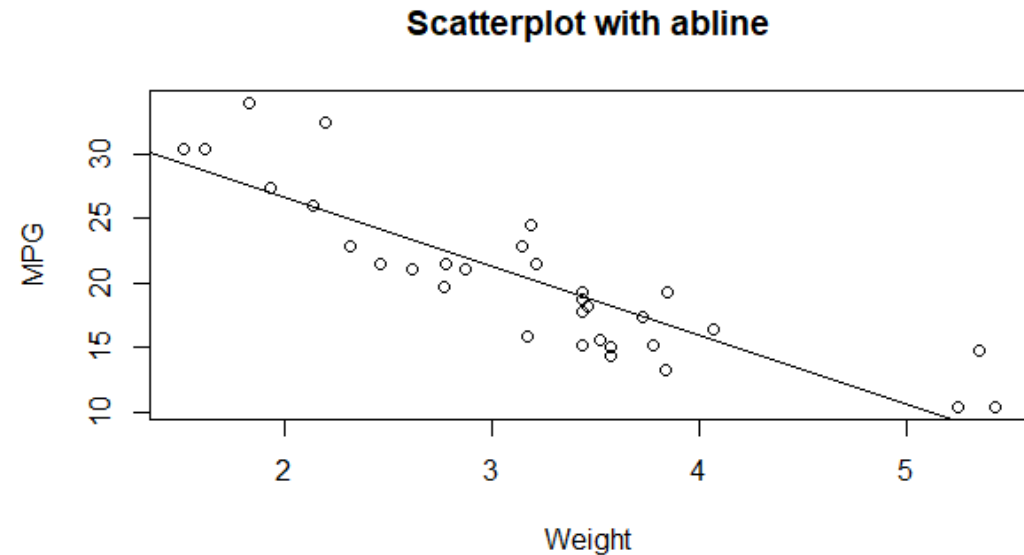
- `plot(dfwt, dfmpg, pch=16)`
- `abline(h=mean(df$mpg), lwd = 2, col = "red")`
- `abline(h=mean(df$mpg) + 1*sd(df$mpg), col = "blue", lwd=3, lty = 2)`
- `abline(h=mean(df$mpg) - 1*sd(df$mpg), col = "blue", lwd=3, lty = 2)`



Scatterplot with “abline” from a model:

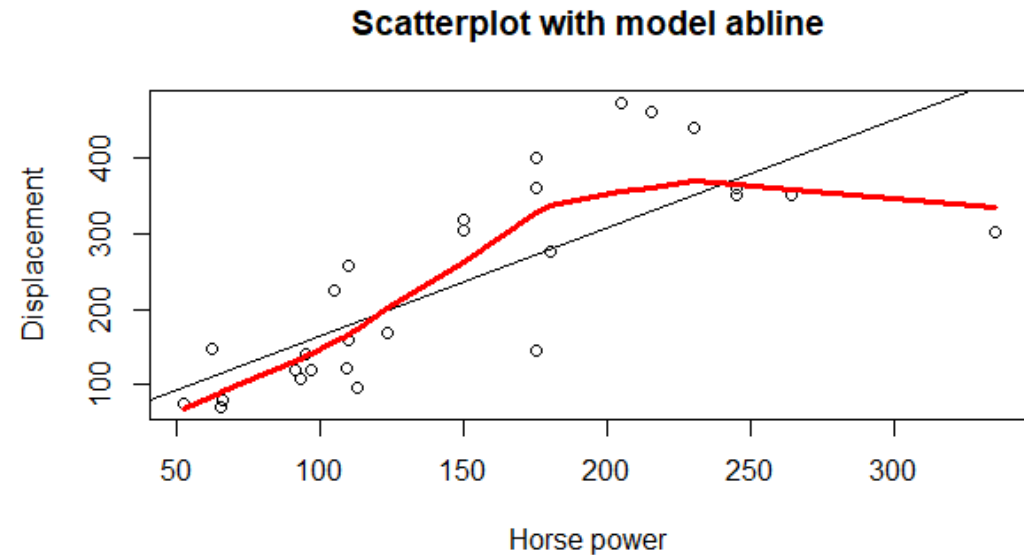
- `plot(dfwt, dfmpg, main = "Scatterplot with abline", xlab = "Weight", ylab = "MPG")`
- `reg_mod <- lm(df$mpg ~ df$wt)`
- `abline(reg_mod)`

- `plot(dfwt, dfmpg, main = "Scatterplot with abline", xlab = "Weight", ylab = "MPG")`
- `abline(lm(df$mpg ~ df$wt))`



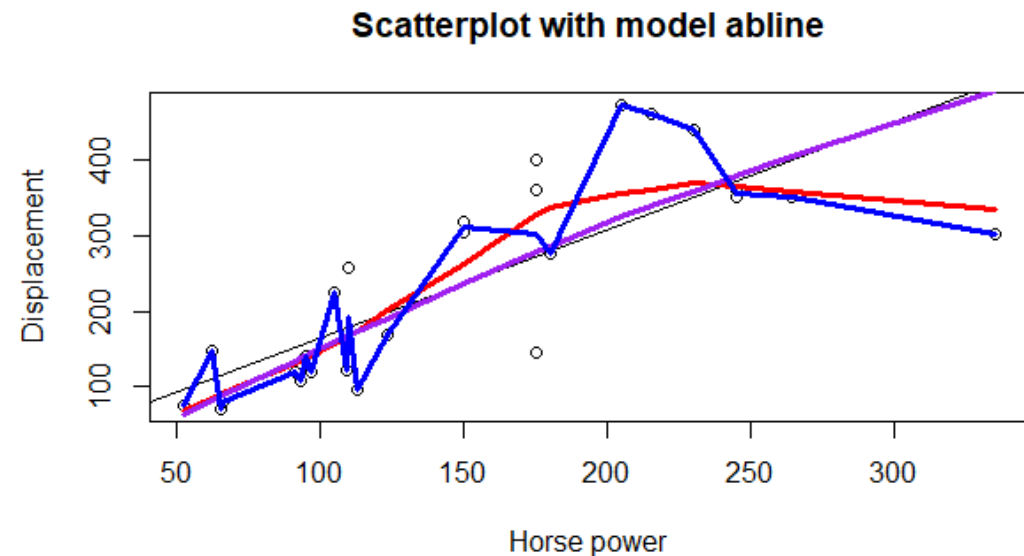
Scatterplot with “abline” and “lines” for a non-linear data:

- `plot(dfhp, dfdisp, main = "Scatterplot with model abline", xlab = "Horse power", ylab = "Displacement")`
- `abline(lm(df$disp ~ df$hp))`
- `lines(lowess(dfhp, dfdisp), col = "red", lwd = 3)`
- **Lowess = Locally weighted Scatterplot Smoothing**



Scatterplot with “abline” and “lines” for a non-linear data: **DO NOT OVERFIT!**

- `plot(dfhp, dfdisp, main = "Scatterplot with model abline", xlab = "Horse power", ylab = "Displacement")`
- `abline(lm(df$disp ~ df$hp))`
- `lines(lowess(dfhp, dfdisp), col = "red", lwd = 3)`
- `lines(lowess(dfhp, dfdisp, f=1), col = "purple", lwd = 3)`
- `lines(lowess(dfhp, dfdisp, f=0.1), col = "blue", lwd = 3)`



More on plots: <https://r-coder.com/plot-r/>

- `set.seed(1)`
- `# Generate sample data`
- `x <- rnorm(500)`
- `y <- x + rnorm(500)`
- `# Plot the data`
- `plot(x, y)`
- `# Equivalent`
- `M <- cbind(x, y)`
- `plot(M)`

Function and arguments

Output plot

- `plot(x,y)`
 - Scatterplot of x and y numeric vectors
- `plot(factor)`
 - Barplot of the factor
- `plot(factor, y)`
 - Boxplot of the numeric vector and the levels of the factor
- `plot(time_series)`
 - Time series plot

Function and arguments

- `plot(date, y)`
- `plot(function, lower, upper)`

Output plot

- Plots a date-based vector
- Plot of the function between the lower and maximum value specified

Question/Queries?

Thank you!

@shitalbhandary