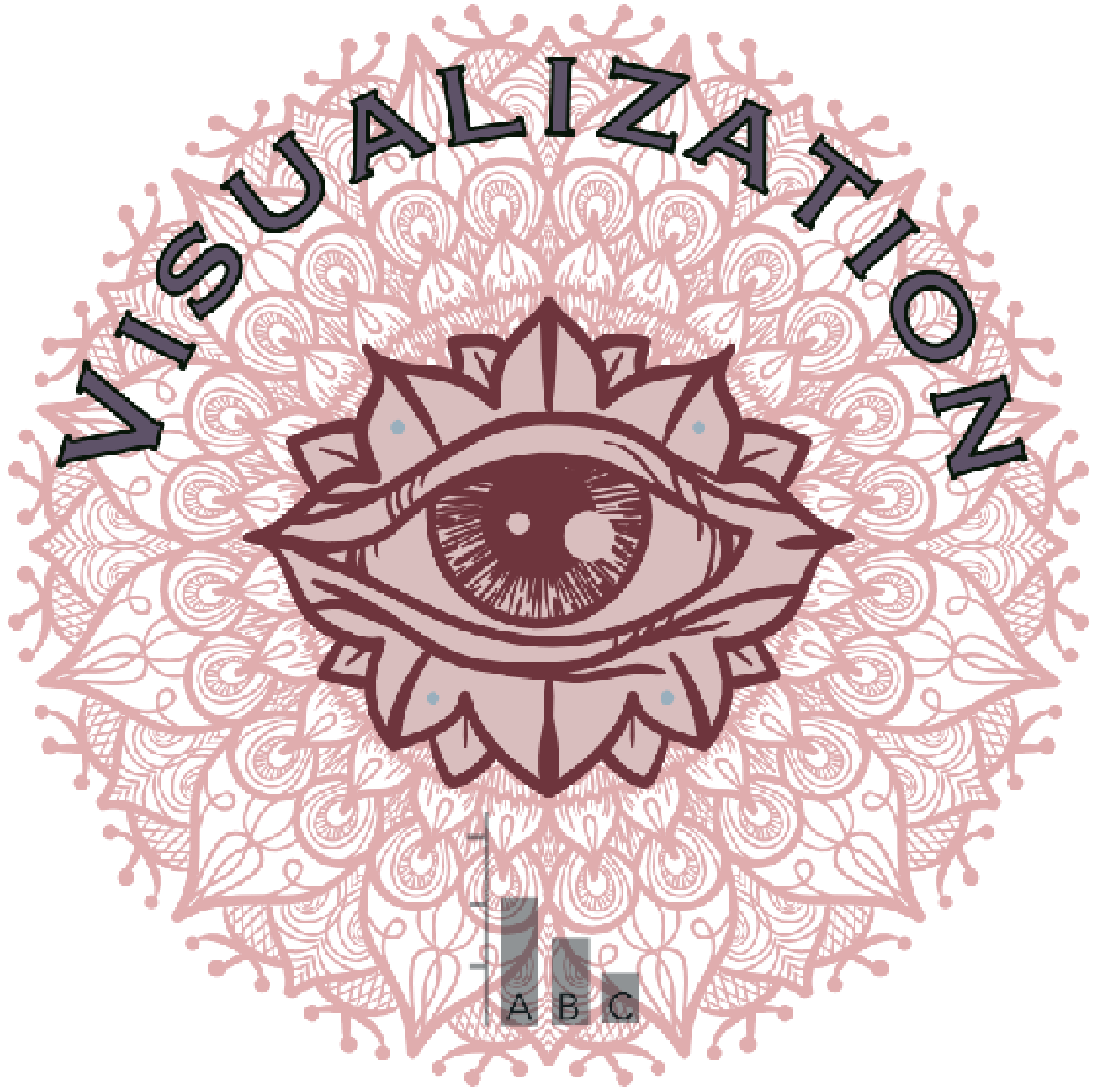
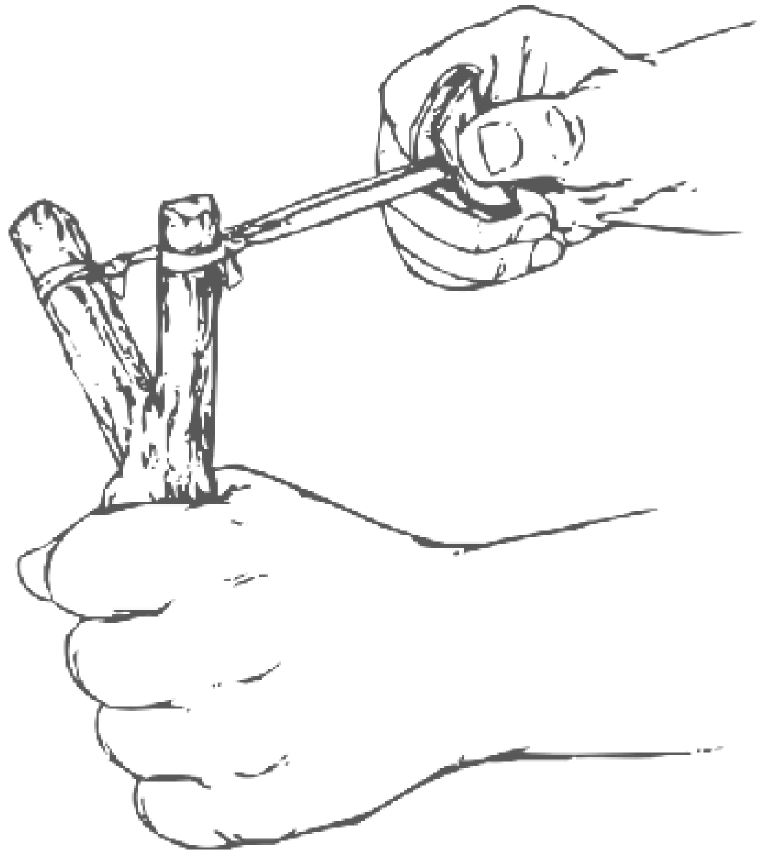


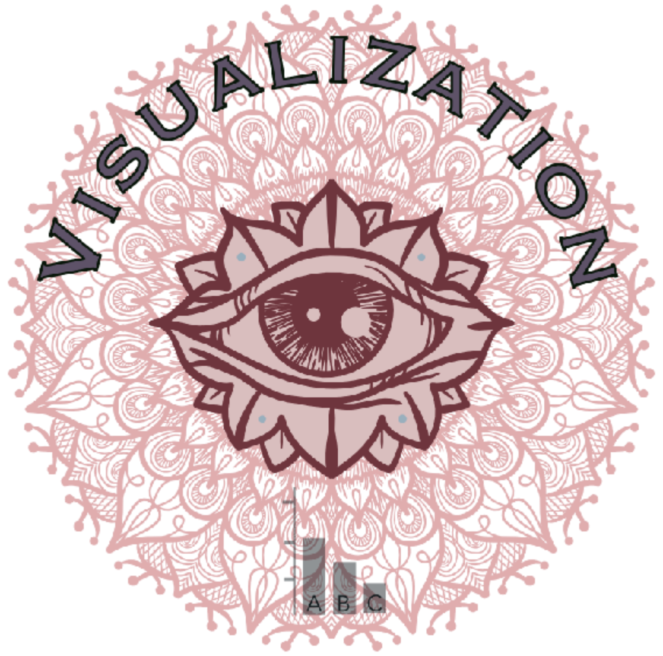
DATA ANALYSIS

DATA VISUALIZATION

LEARNING GOALS



- obtain a basic understanding of better/worse plotting
  - understand the idea of hypothesis-driven visualization
- develop a basic understanding of the 'grammar of graphs'
- get familiar with frequent visualization strategies
  - barplots, densities, violins, error bars etc.
- be able to fine-tune graphs for better visualization



Motivation

WHY VISUALIZE

- a picture can be worth a million words (and numbers)

- every data analysis should start with a ‘getting to know the data’ phase
  - visualization of different aspects of data is key to get intimate with the data

- data visualization as a means of communication (with others)
  - hypothesis-driven visualization: obtain visual (suggestive) evidence regarding a research question of relevance

WHY VISUALIZE

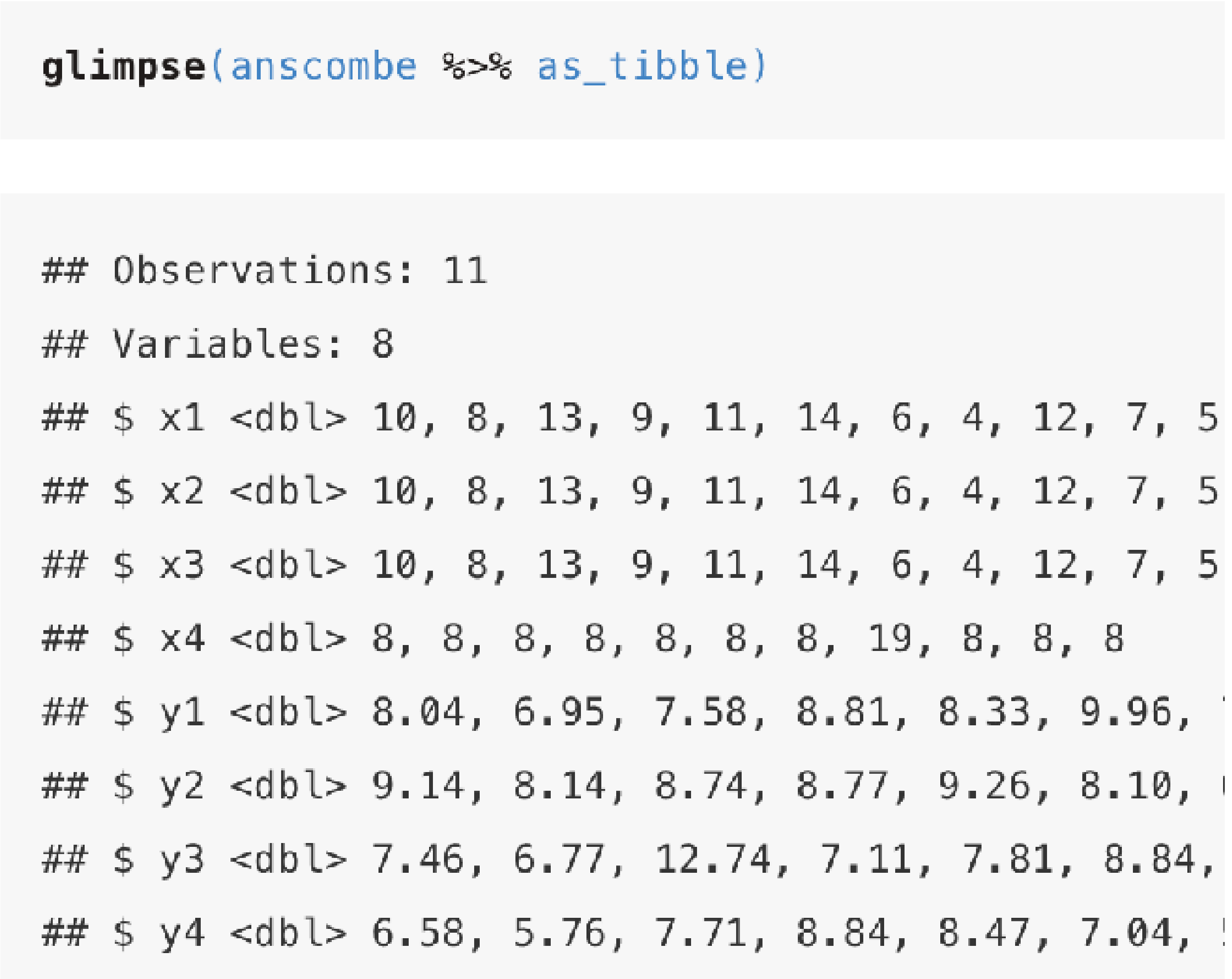
- a picture can be worth a million words (and numbers)
  - summary statistics can be misleading (because of information loss)
- every data analysis should start with a ‘getting to know the data’ phase
  - use extensive visualization to get intimate with the data
- data visualization as a means of communication (with others / with yourself)
  - hypothesis-driven visualization: obtain visual (suggestive) evidence regarding a research question of relevance

BEYOND SUMMARY STATISTIC

are your summary statistics hiding something interesting?
@allison_horst

Motivating example: Anscombe’s quartet

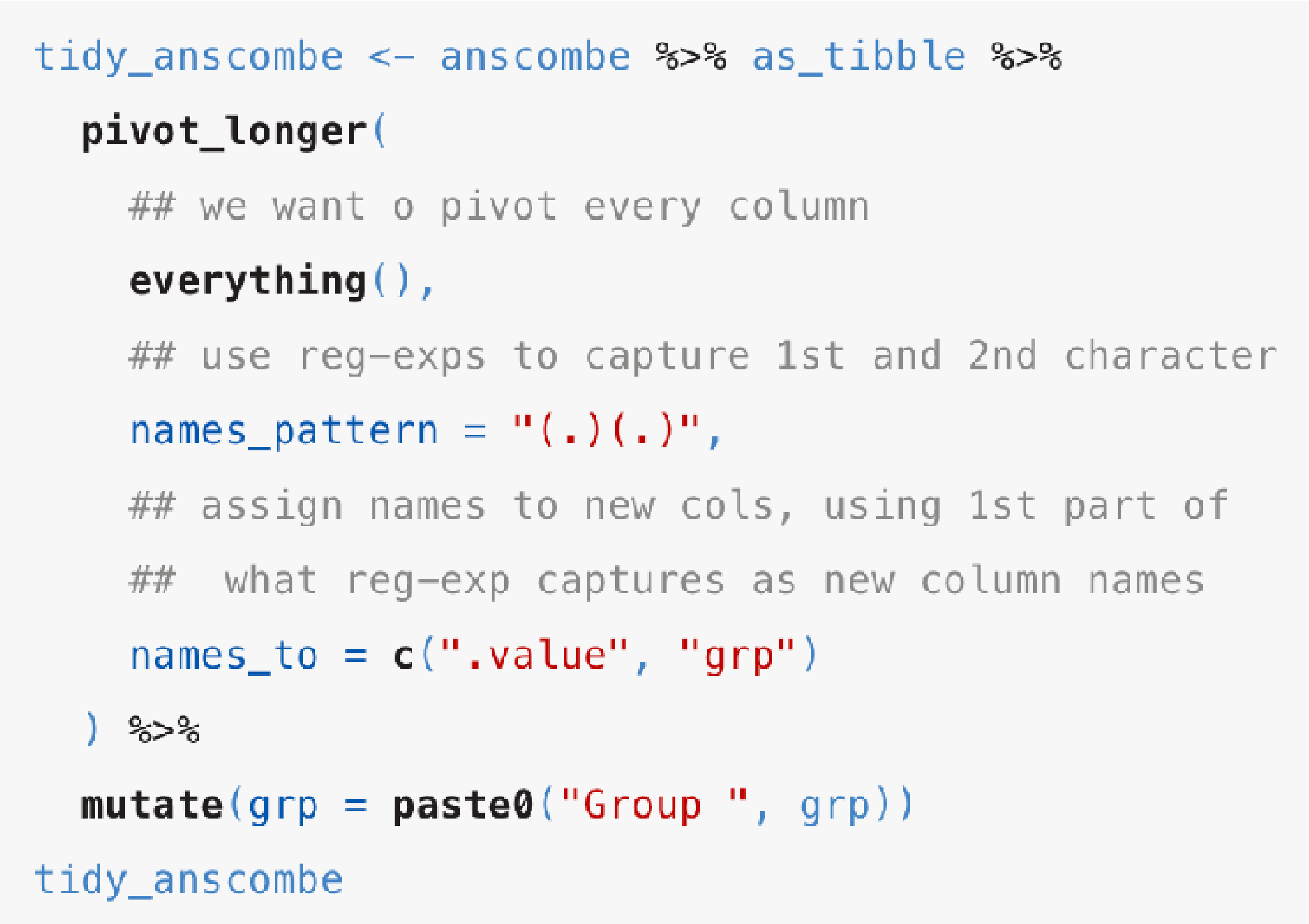
- famous data set, ships with core R



```r
glimpse(anscombe %>% as_tibble)
```

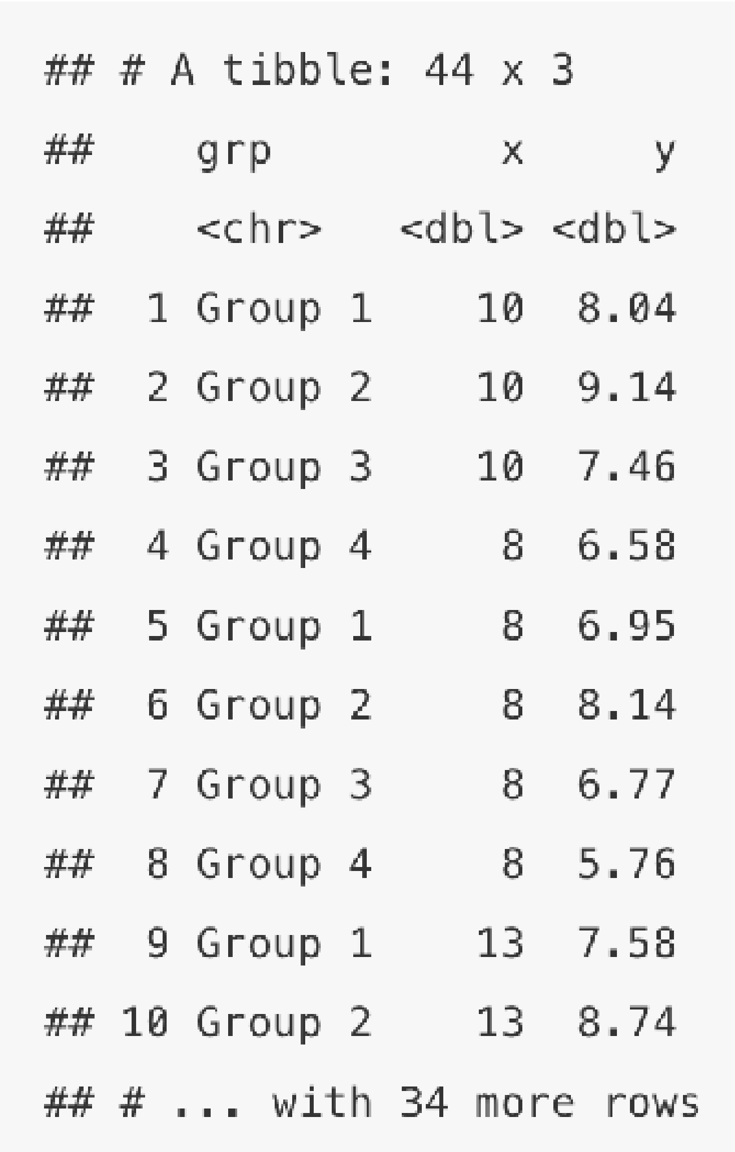
```
## Observations: 11
## Variables: 8
## $ x1 <dbl> 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
## $ x2 <dbl> 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
## $ x3 <dbl> 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
## $ x4 <dbl> 8, 8, 8, 8, 8, 8, 8, 19, 8, 8, 8
## $ y1 <dbl> 8.04, 6.95, 7.58, 8.81, 8.33, 9.96,
## $ y2 <dbl> 9.14, 8.14, 8.74, 8.77, 9.26, 8.10,
## $ y3 <dbl> 7.46, 6.77, 12.74, 7.11, 7.81, 8.84,
## $ y4 <dbl> 6.58, 5.76, 7.71, 8.84, 8.47, 7.04,
```

messy start



```r
tidy_anscombe <- anscombe %>% as_tibble %>%
  pivot_longer(
    ## we want o pivot every column
    everything(),
    ## use reg-exps to capture 1st and 2nd character
    names_pattern = "(.)(.)",
    ## assign names to new cols, using 1st part of
    ##  what reg-exp captures as new column names
    names_to = c(".value", "grp")
  ) %>%
  mutate(grp = paste0("Group ", grp))
tidy_anscombe
```

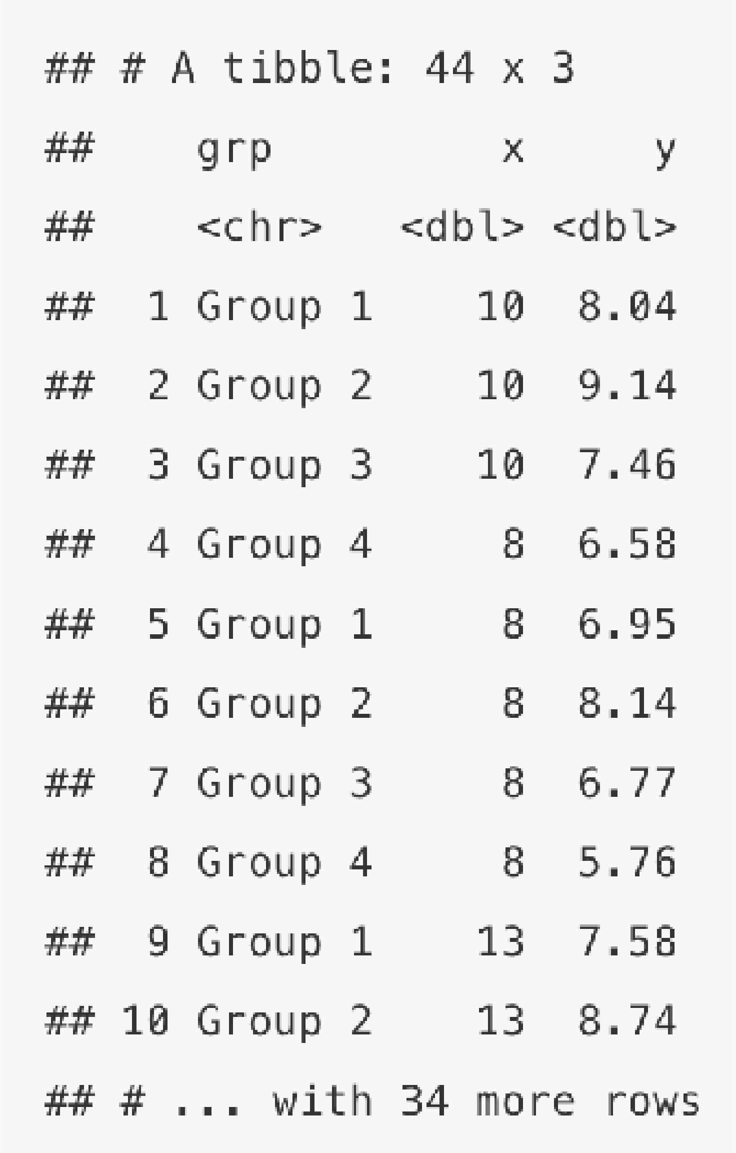
tidy up



```
## # A tibble: 44 x 3
##    grp         x     y
##    <chr>   <dbl> <dbl>
##  1 Group 1    10  8.04
##  2 Group 2    10  9.14
##  3 Group 3    10  7.46
##  4 Group 4     8  6.58
##  5 Group 1     8  6.95
##  6 Group 2     8  8.14
##  7 Group 3     8  6.77
##  8 Group 4     8  5.76
##  9 Group 1    13  7.58
## 10 Group 2    13  8.74
## # ... with 34 more rows
```

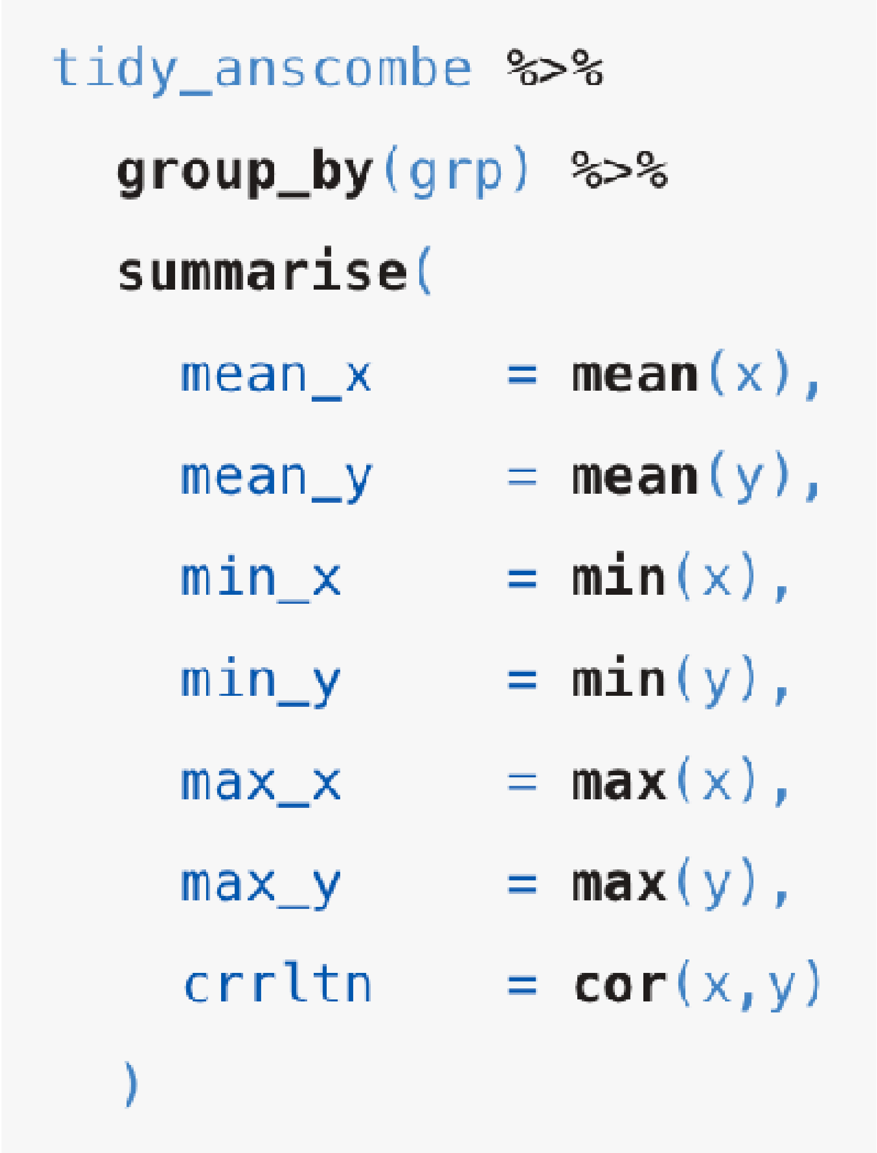
nice!

Motivating example: Anscombe’s quartet



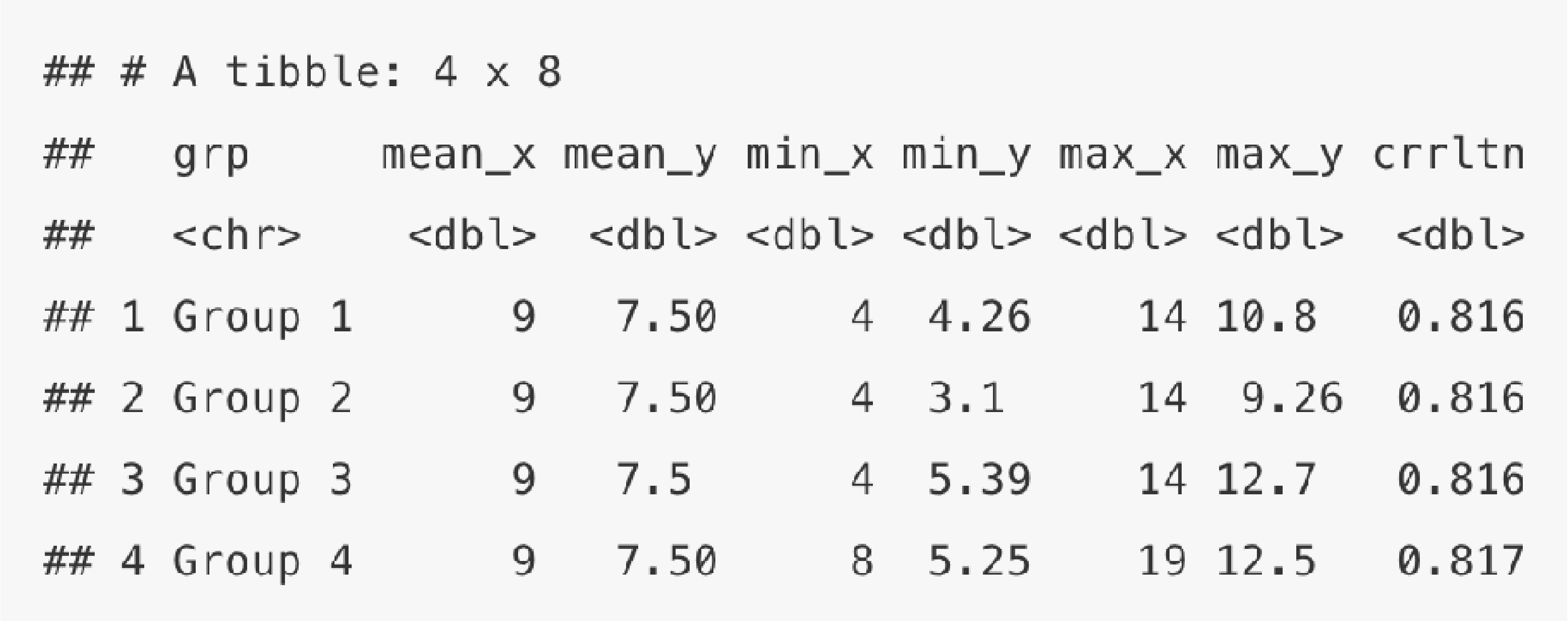
```
## # A tibble: 44 x 3
##    grp         x     y
##    <chr>   <dbl> <dbl>
##  1 Group 1    10  8.04
##  2 Group 2    10  9.14
##  3 Group 3    10  7.46
##  4 Group 4     8  6.58
##  5 Group 1     8  6.95
##  6 Group 2     8  8.14
##  7 Group 3     8  6.77
##  8 Group 4     8  5.76
##  9 Group 1    13  7.58
## 10 Group 2    13  8.74
## # ... with 34 more rows
```

input data



```
tidy_anscombe %>%
  group_by(grp) %>%
  summarise(
    mean_x    = mean(x),
    mean_y    = mean(y),
    min_x     = min(x),
    min_y     = min(y),
    max_x     = max(x),
    max_y     = max(y),
    crrltn    = cor(x,y)
  )
```

summarise

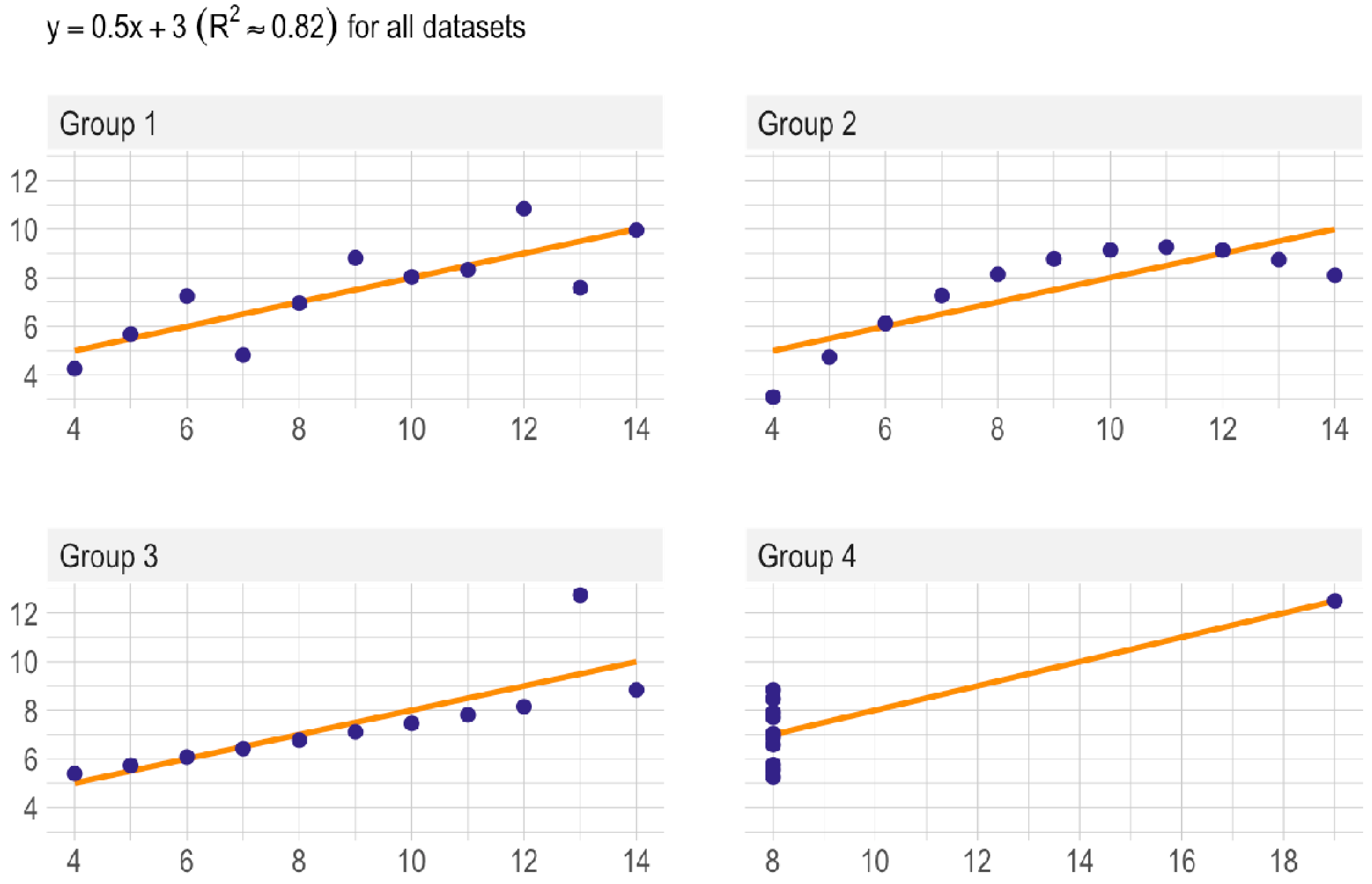


```
## # A tibble: 4 x 8
##   grp     mean_x mean_y min_x min_y max_x max_y crrltn
##   <chr>    <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 Group 1      9   7.50     4  4.26    14 10.8   0.816
## 2 Group 2      9   7.50     4  3.1     14  9.26  0.816
## 3 Group 3      9   7.5      4  5.39    14 12.7   0.816
## 4 Group 4      9   7.50     8  5.25    19 12.5   0.817
```

all four groups look very similar!

Motivating example: Anscombe's quartet

- quite different patterns despite similar correlation



$y = 0.5x + 3$ $(R^2 \approx 0.82)$ for all datasets

Group 1

Group 2

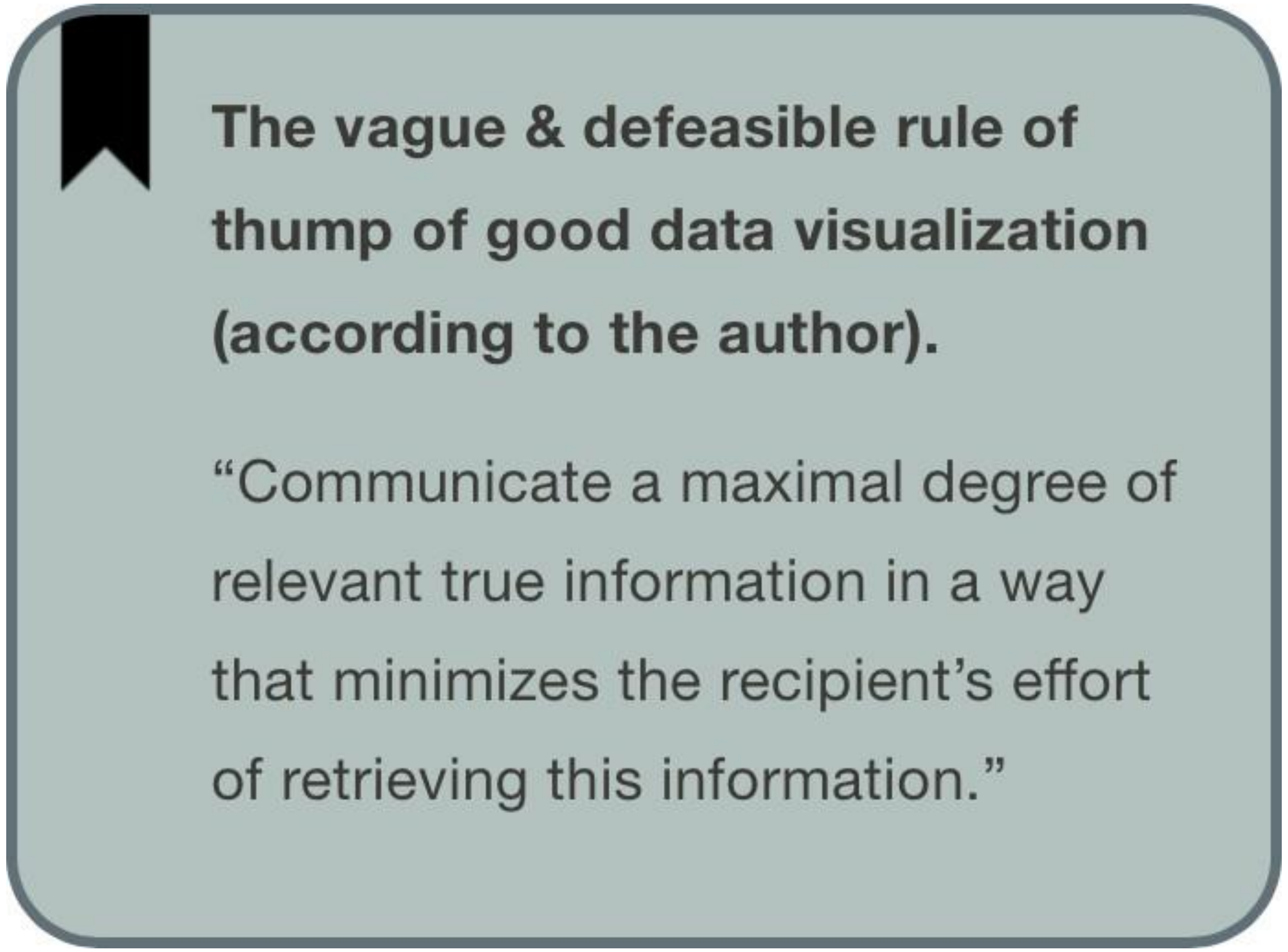
Group 3

Group 4

The good, the bad and the info-graphic

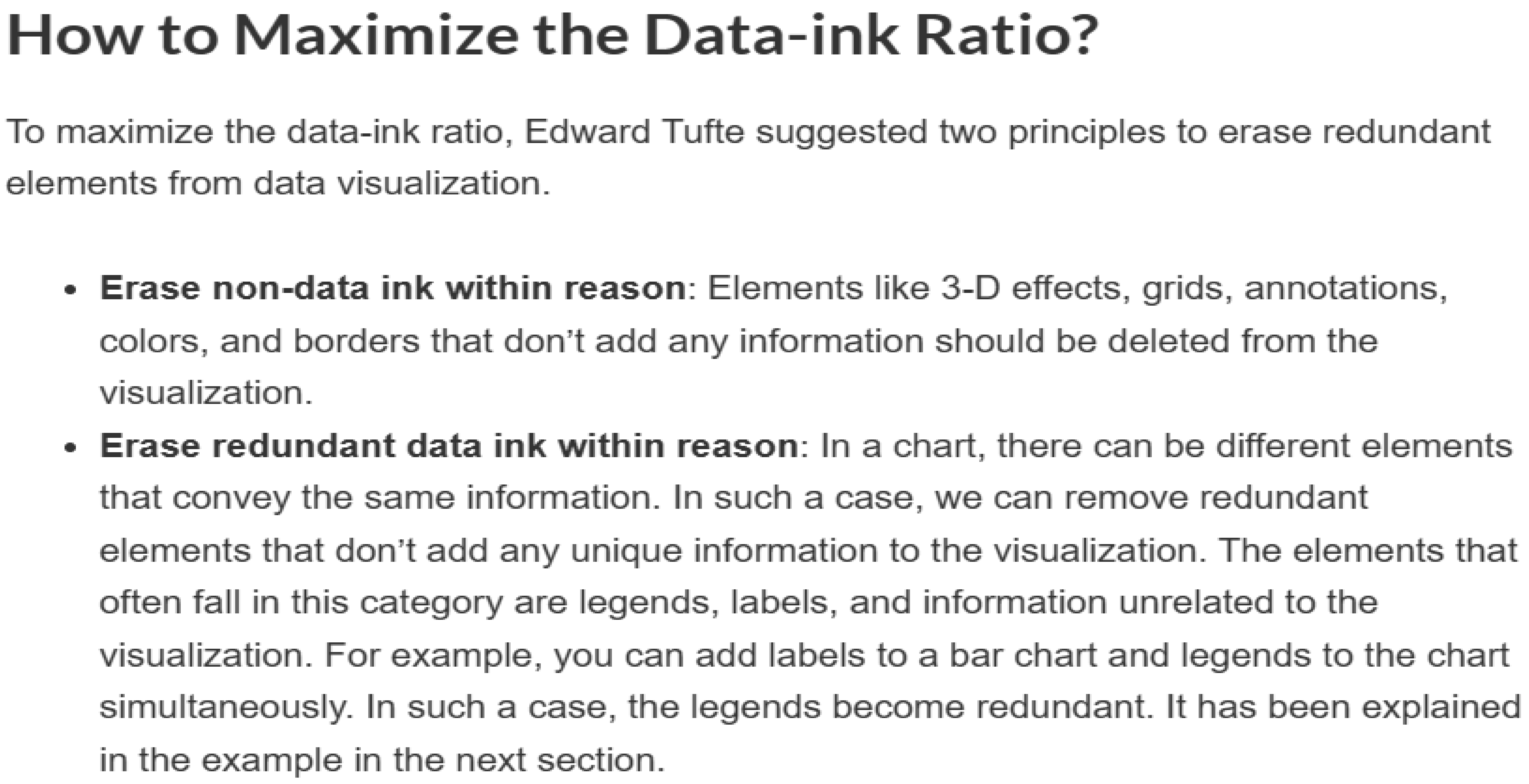
PRINCIPLE OF GOOD VISUALIZATION

- maximize data-ink ratio (Tufte 1983)
  - maximize information, minimize ink
  - eliminate chart junk
  - ink vs. processing effort
- analogy to language
  - information flow
  - ease of processing
  - bound by conventional rules
- hypothesis-driven visualization
  - relevance of information



The vague & defeasible rule of thump of good data visualization (according to the author).

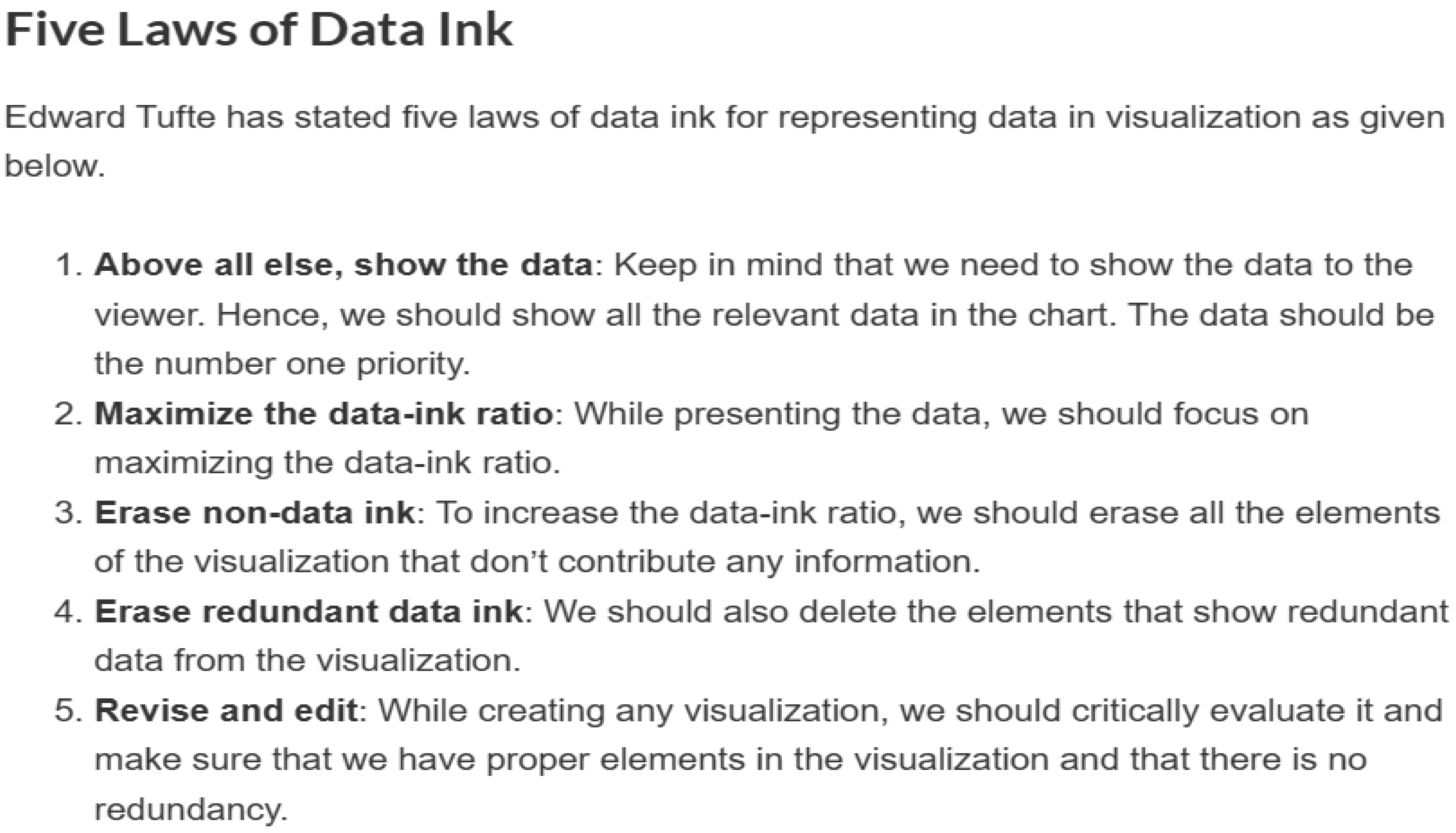
"Communicate a maximal degree of relevant true information in a way that minimizes the recipient's effort of retrieving this information."



How to Maximize the Data-ink Ratio?

To maximize the data-ink ratio, Edward Tufte suggested two principles to erase redundant elements from data visualization.

- **Erase non-data ink within reason**: Elements like 3-D effects, grids, annotations, colors, and borders that don't add any information should be deleted from the visualization.
- **Erase redundant data ink within reason**: In a chart, there can be different elements that convey the same information. In such a case, we can remove redundant elements that don't add any unique information to the visualization. The elements that often fall in this category are legends, labels, and information unrelated to the visualization. For example, you can add labels to a bar chart and legends to the chart simultaneously. In such a case, the legends become redundant. It has been explained in the example in the next section.

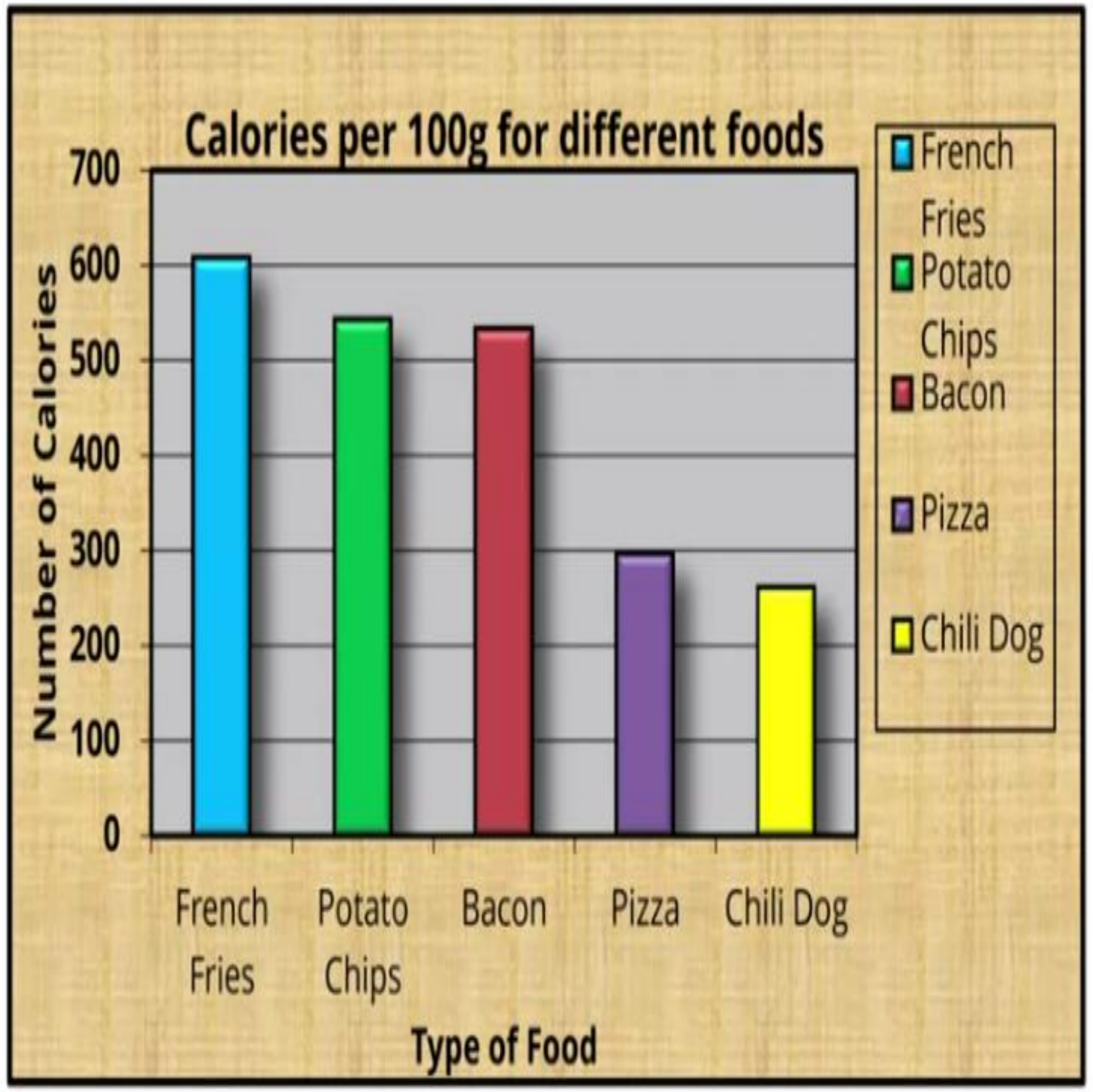


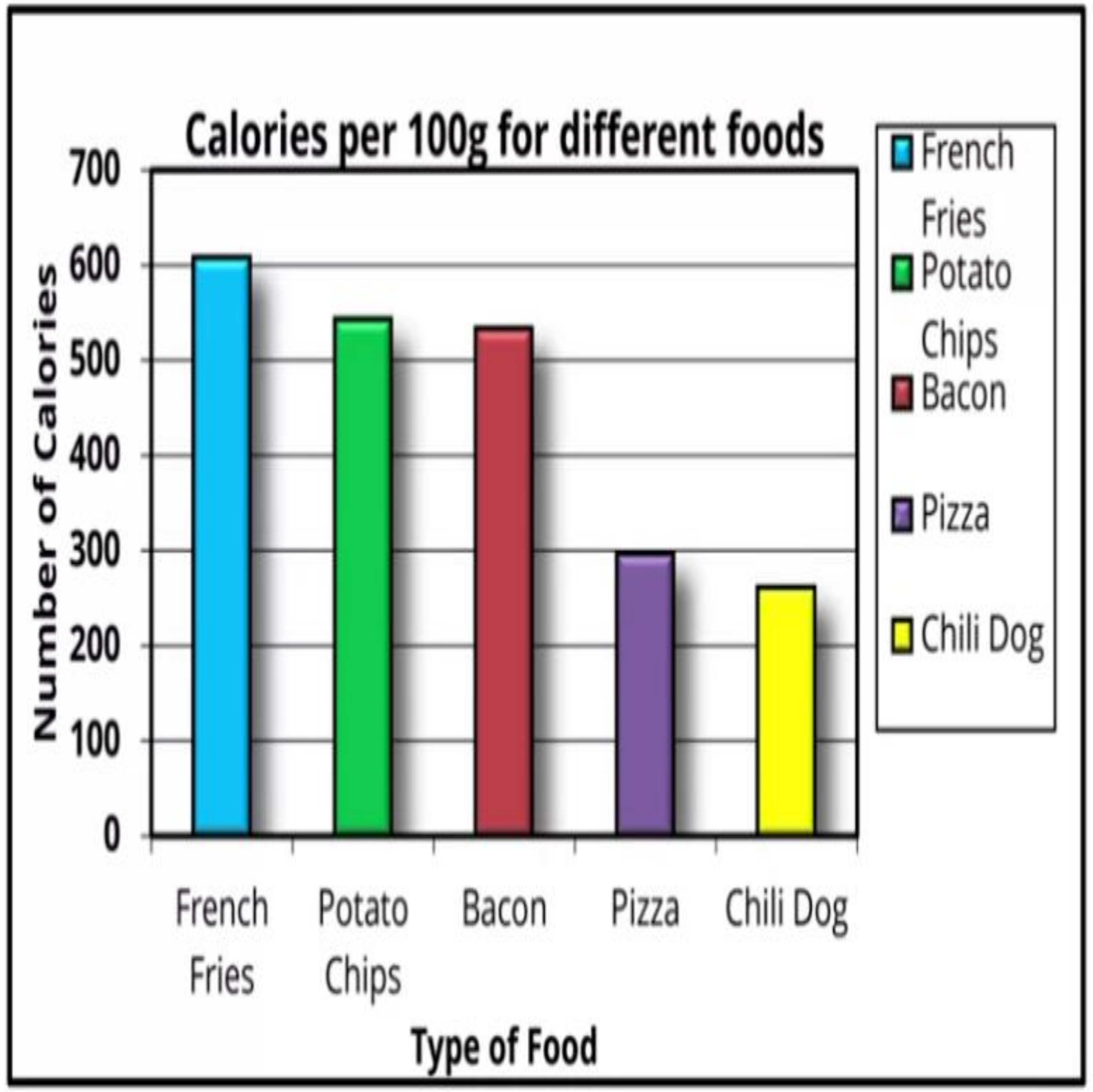
Five Laws of Data Ink

Edward Tufte has stated five laws of data ink for representing data in visualization as given below.

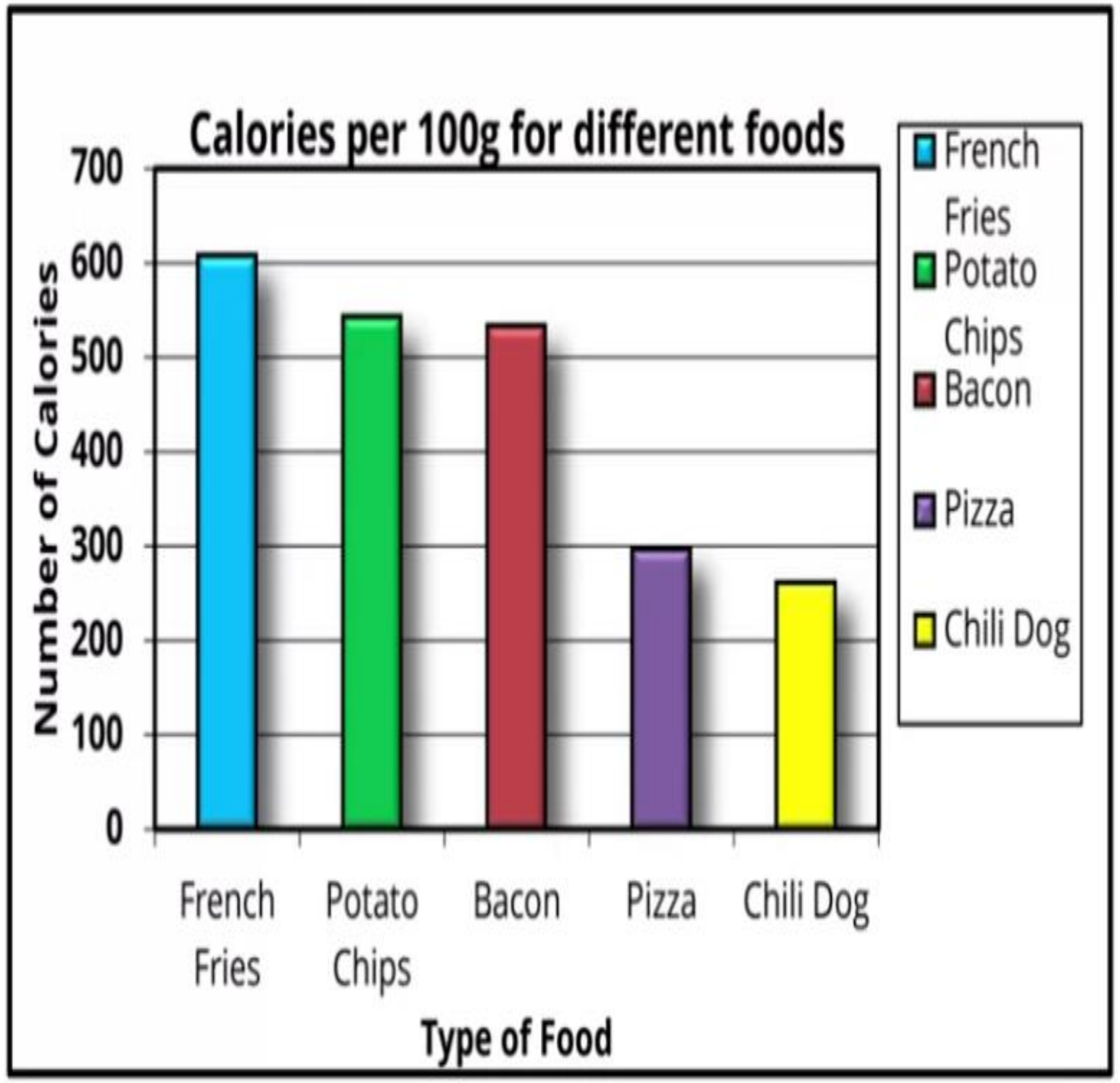
1. **Above all else, show the data**: Keep in mind that we need to show the data to the viewer. Hence, we should show all the relevant data in the chart. The data should be the number one priority.
2. **Maximize the data-ink ratio**: While presenting the data, we should focus on maximizing the data-ink ratio.
3. **Erase non-data ink**: To increase the data-ink ratio, we should erase all the elements of the visualization that don’t contribute any information.
4. **Erase redundant data ink**: We should also delete the elements that show redundant data from the visualization.
5. **Revise and edit**: While creating any visualization, we should critically evaluate it and make sure that we have proper elements in the visualization and that there is no redundancy.

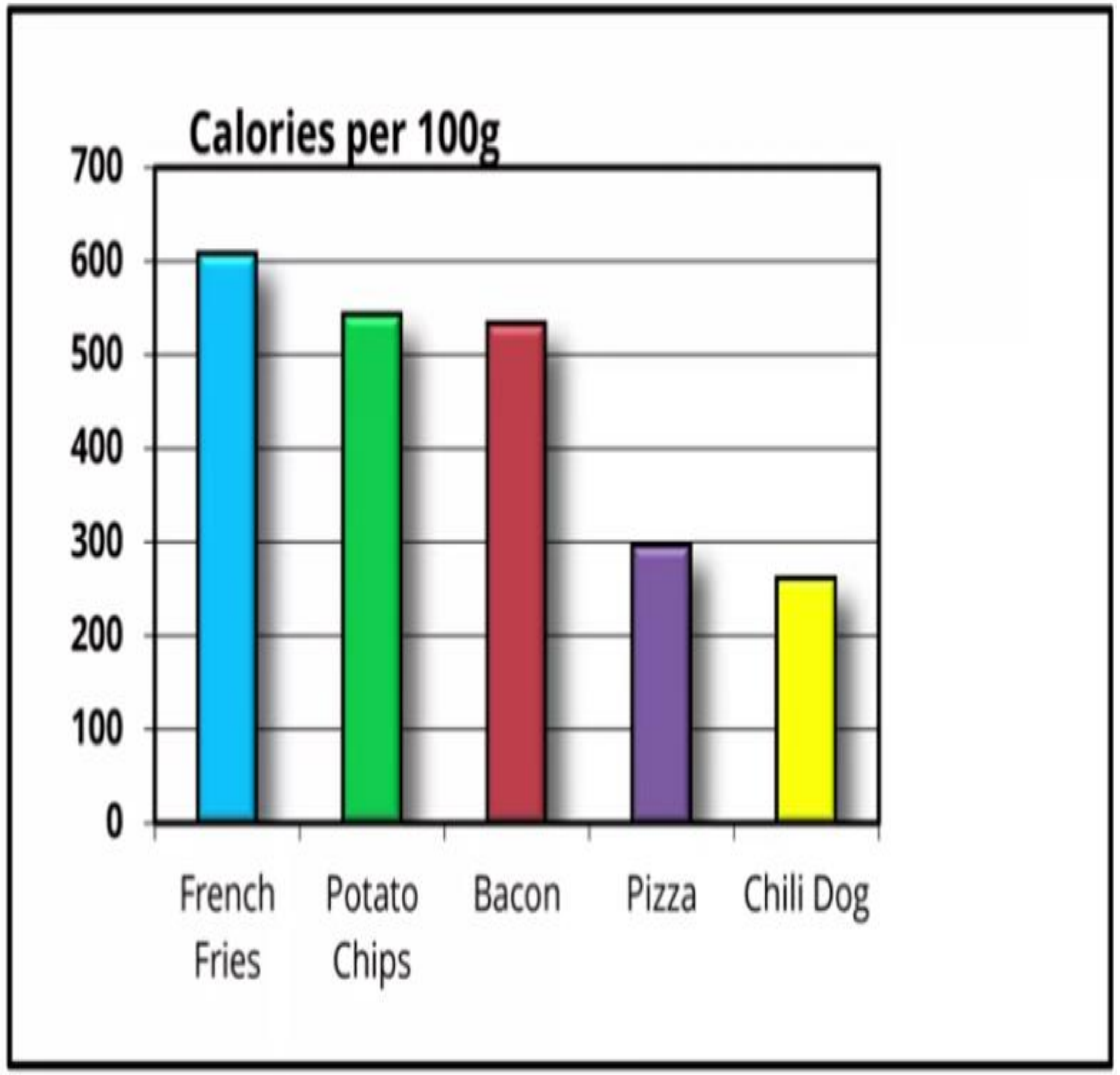
Data ink Ratio Maximization Example



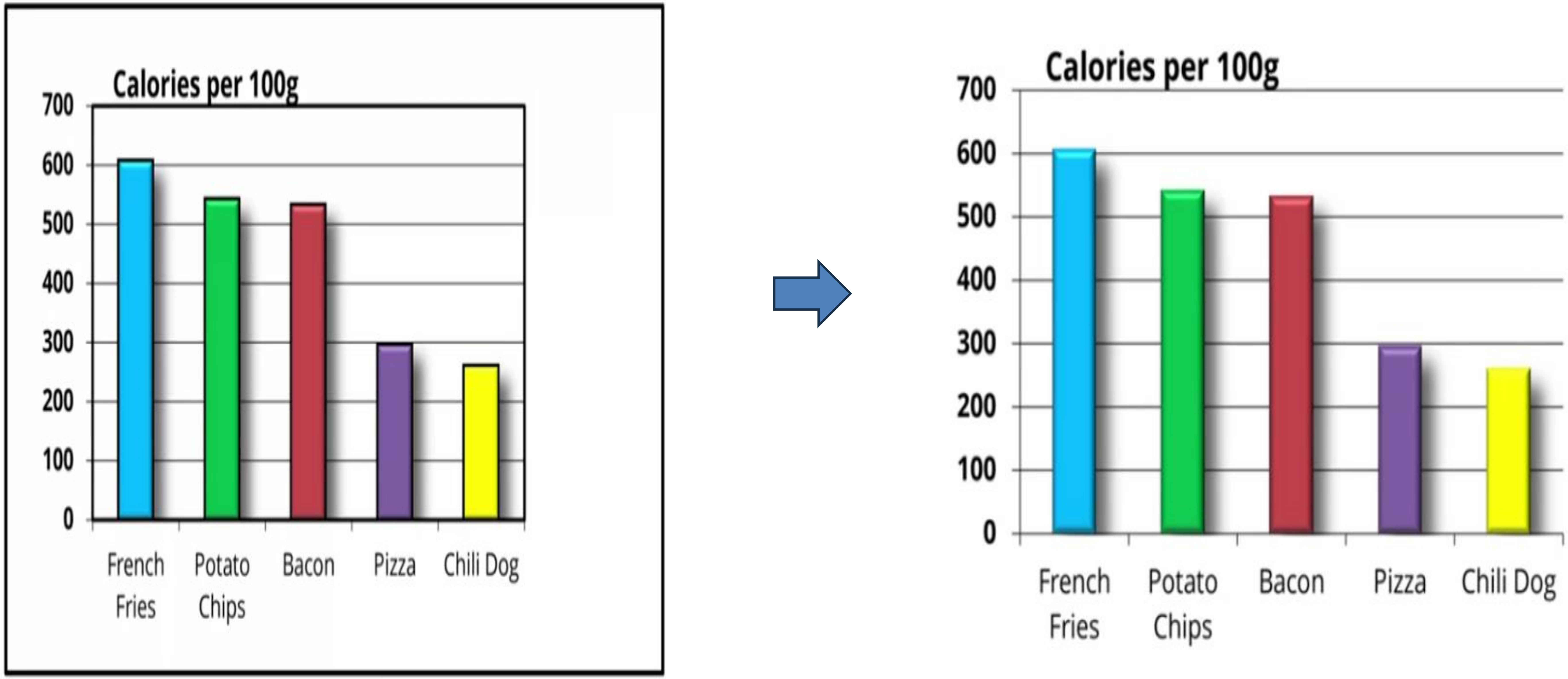


Data ink Ratio Maximization Example

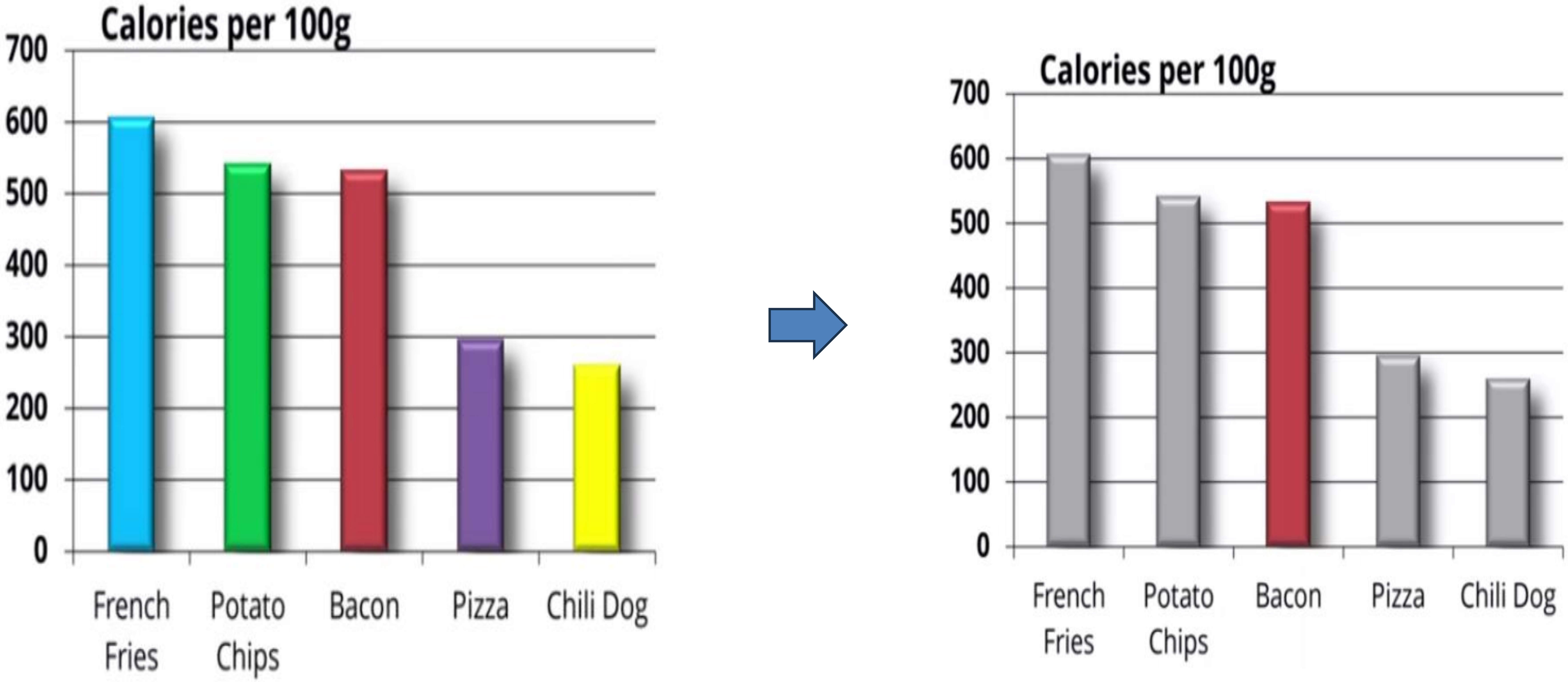




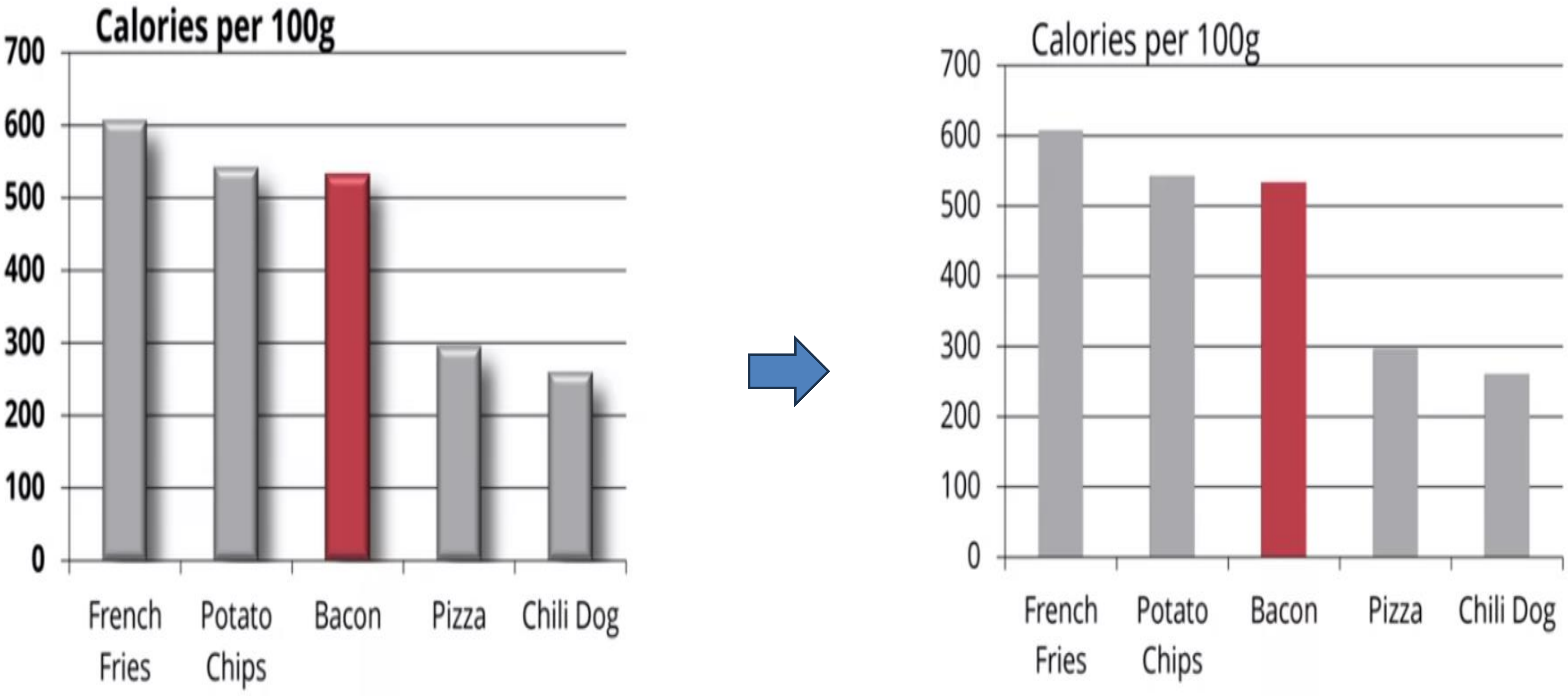
Data ink Ratio Maximization Example



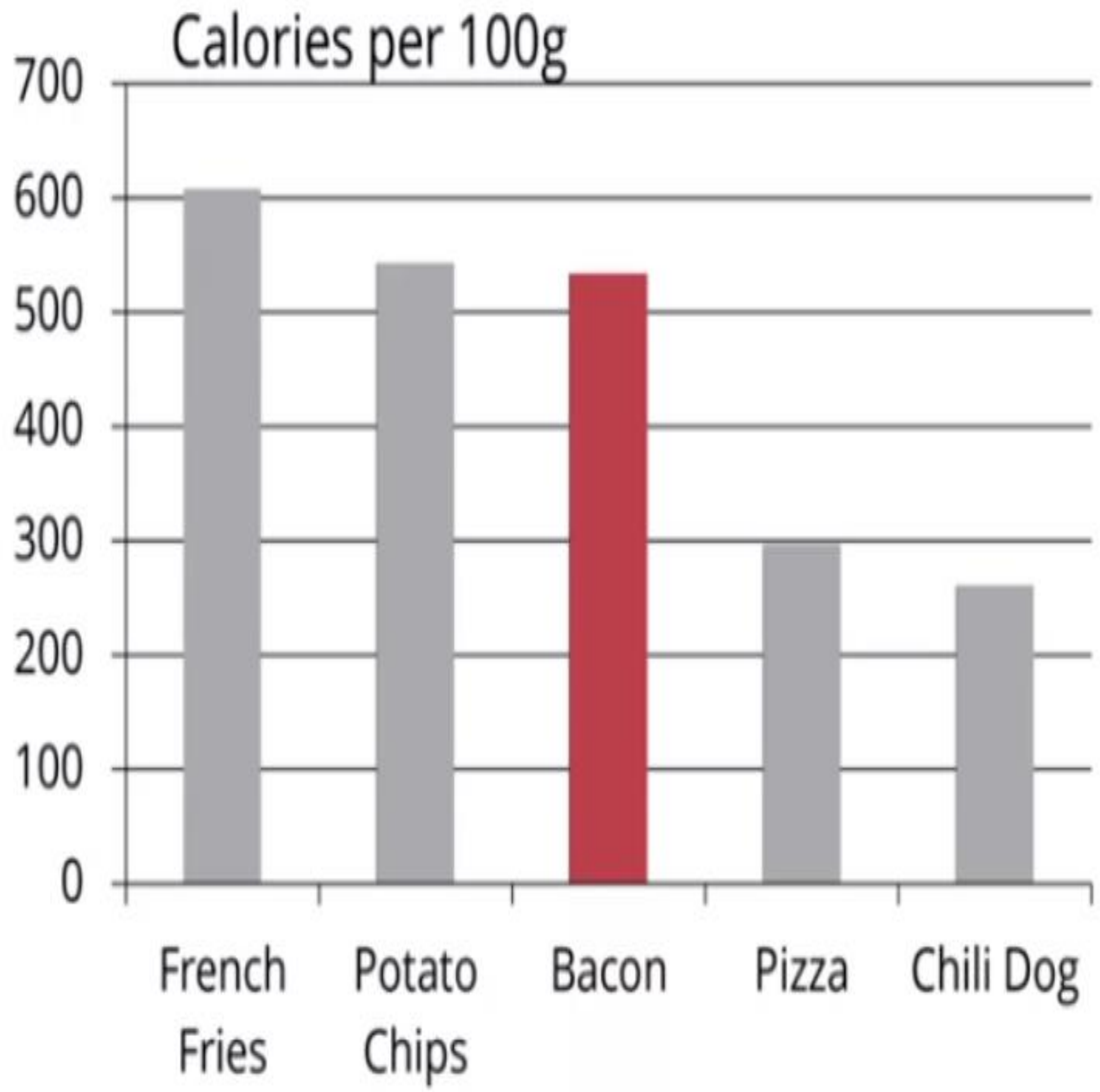
Data ink Ratio Maximization Example

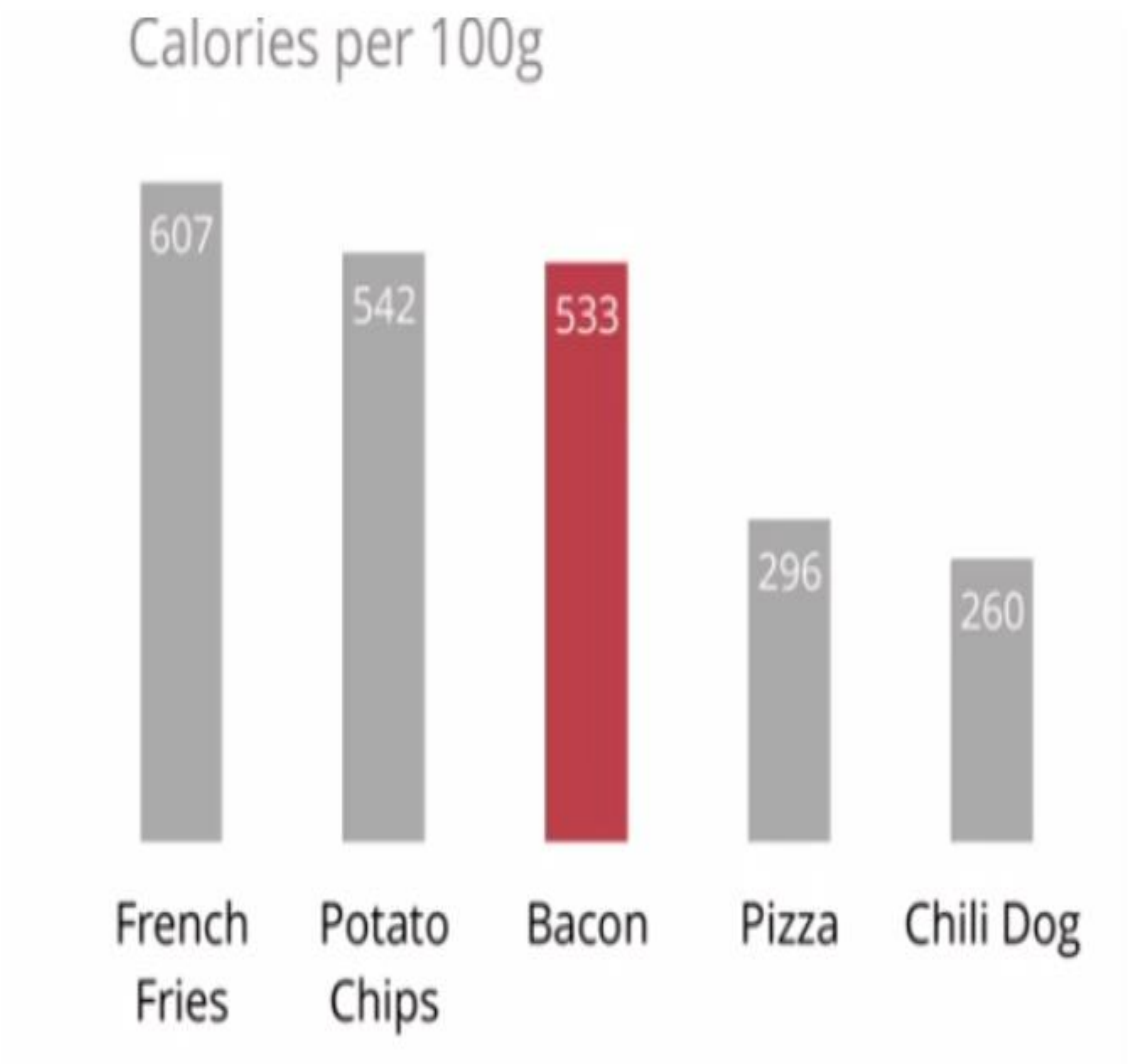


Data ink Ratio Maximization Example

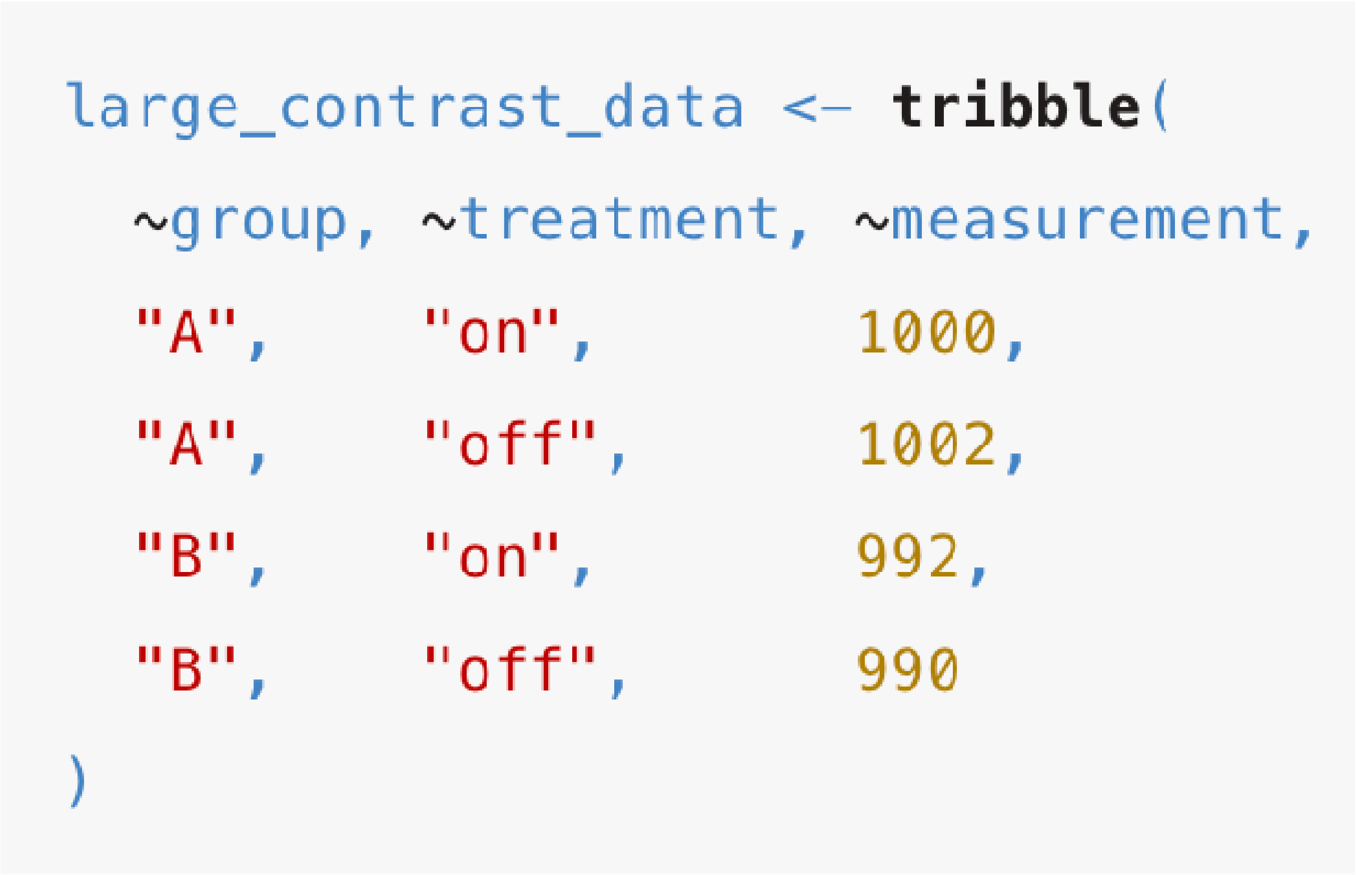


Data ink Ratio Maximization Example

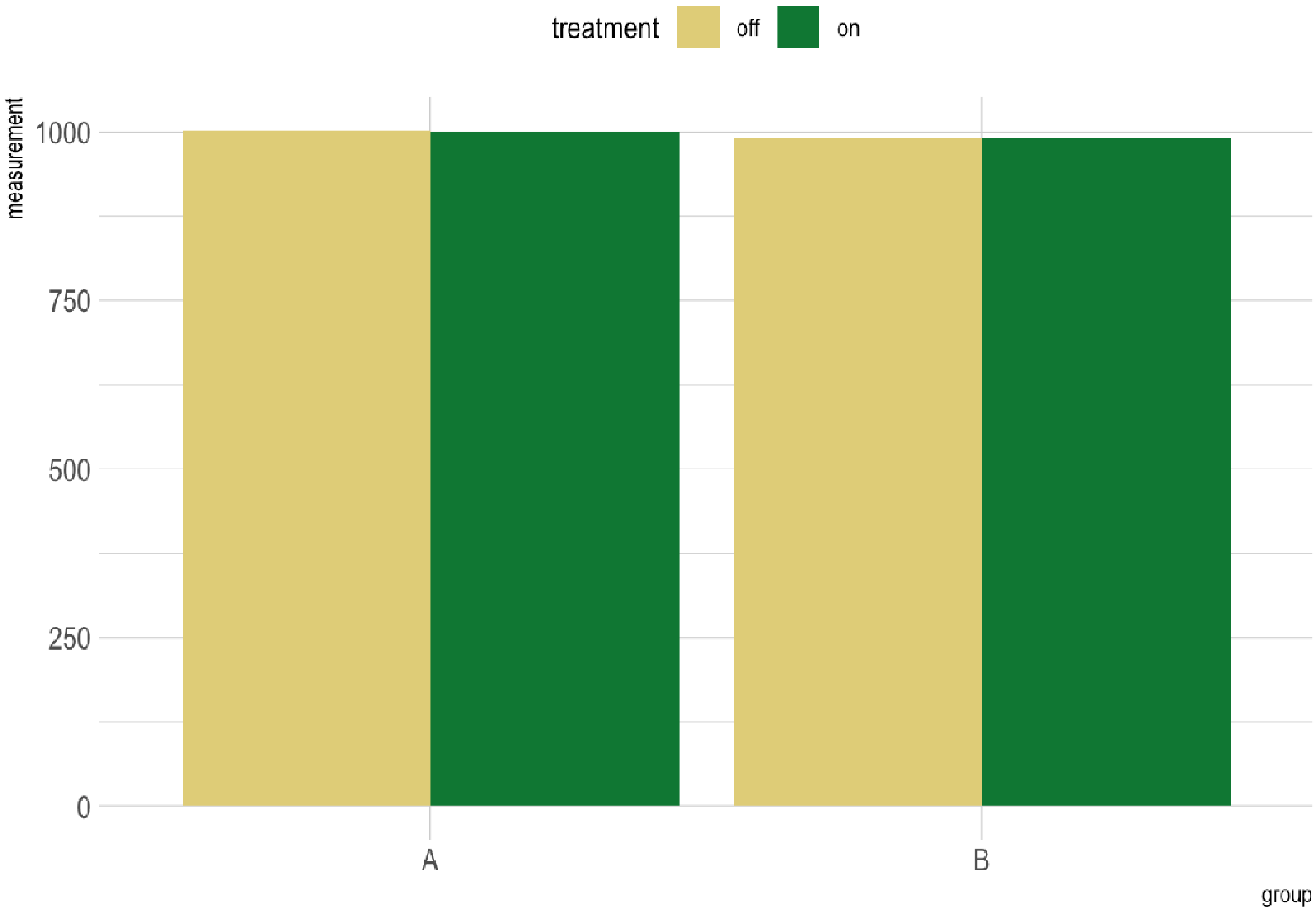




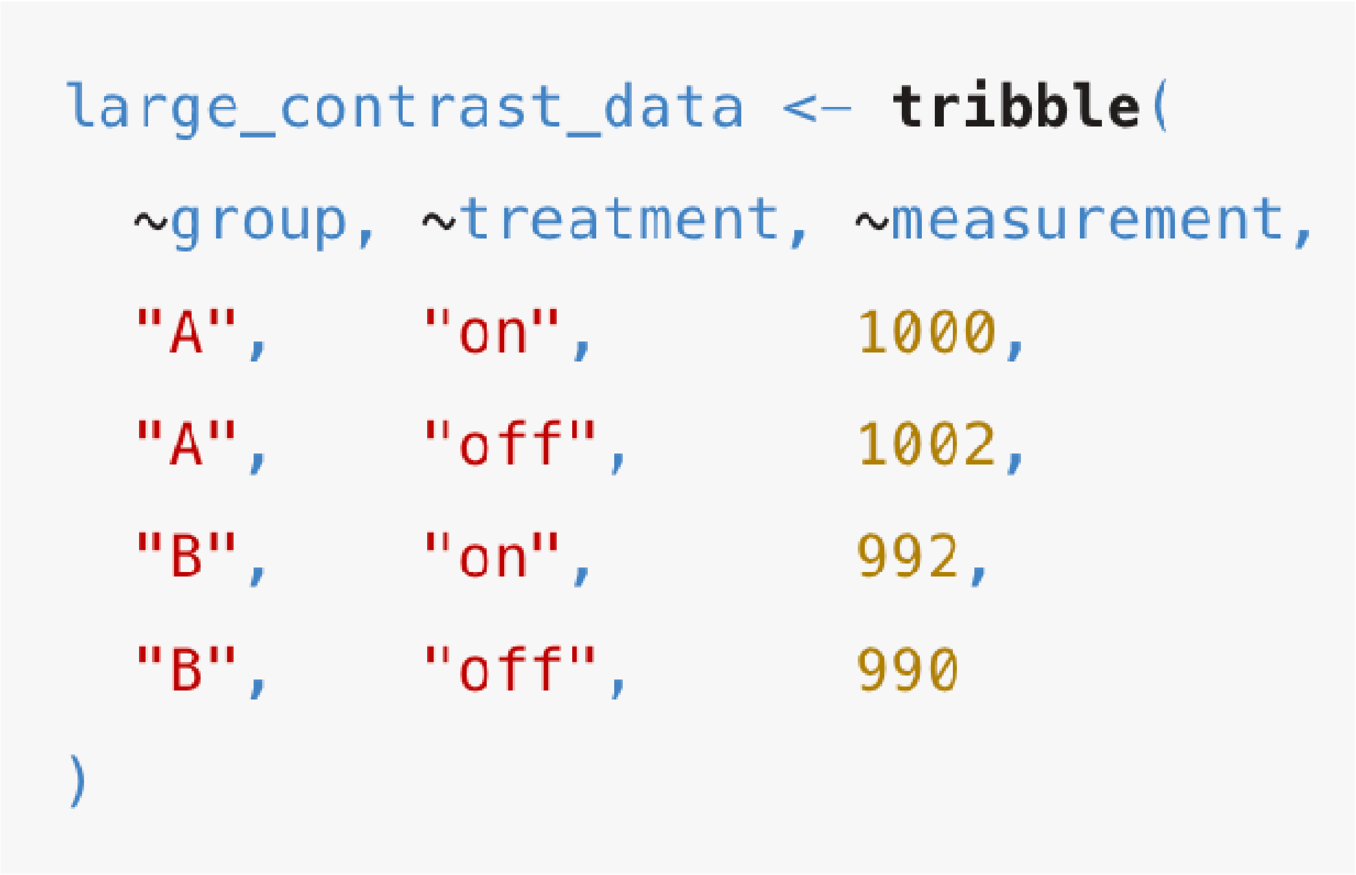
EXAMPLE OF UNINFORMATIVE PLOTTING



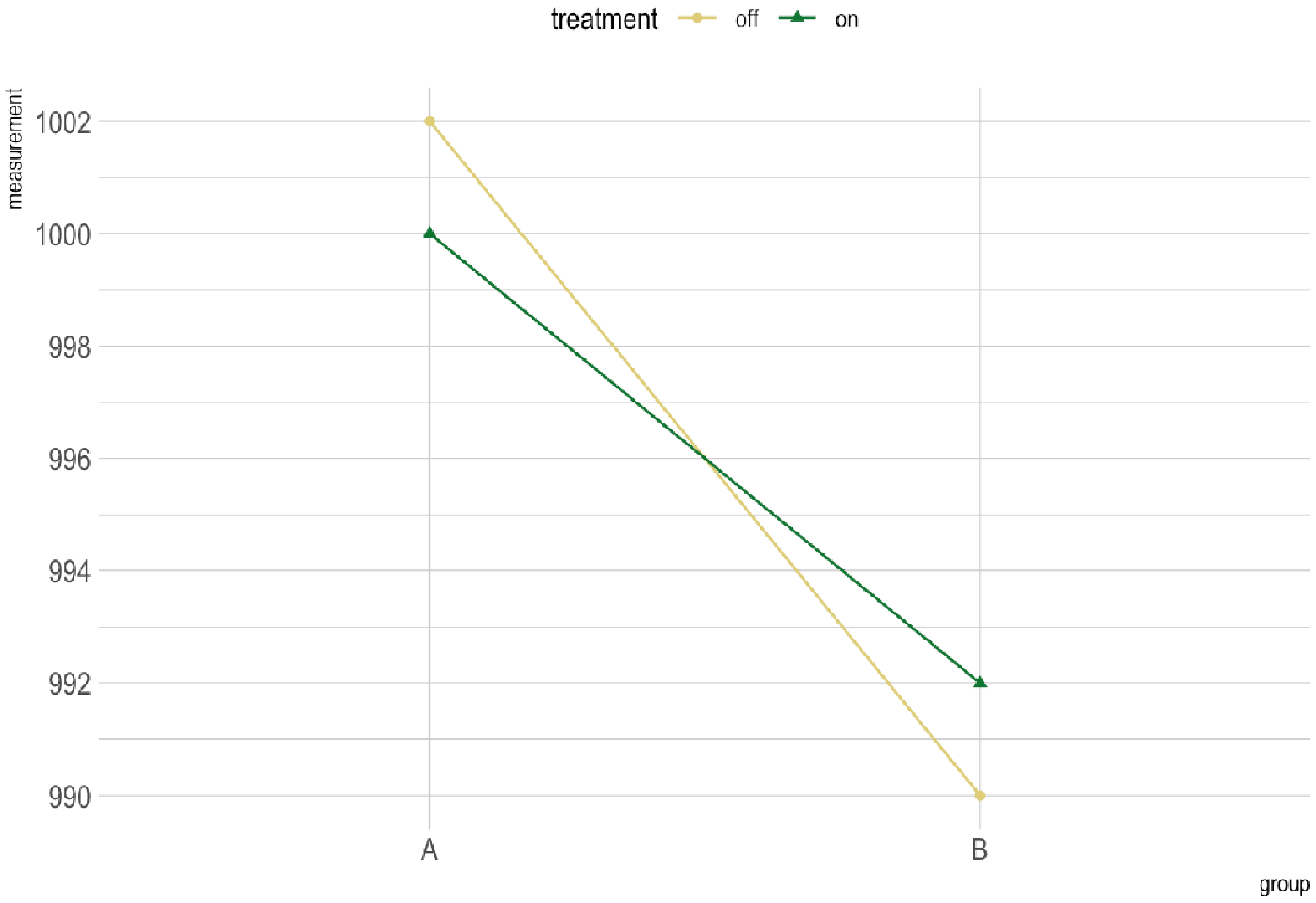
```
large_contrast_data <- tribble(
  ~group, ~treatment, ~measurement,
  "A",    "on",       1000,
  "A",    "off",      1002,
  "B",    "on",       992,
  "B",    "off",      990
)
```



EXAMPLE OF INFORMATIVE HYPOTHESIS-DRIVEN PLOTTING

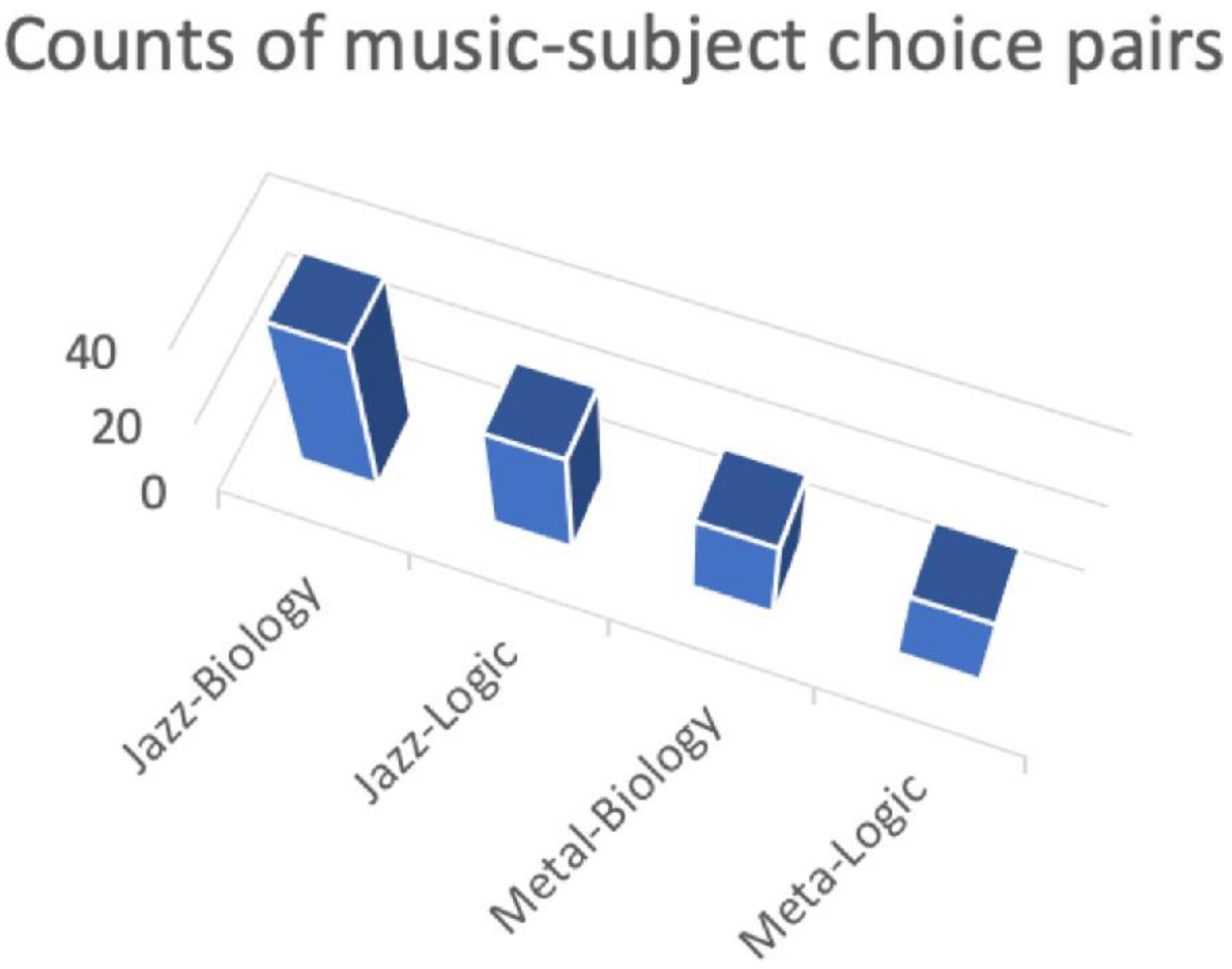


```r
large_contrast_data <- tribble(
  ~group, ~treatment, ~measurement,
  "A",    "on",        1000,
  "A",    "off",       1002,
  "B",    "on",        992,
  "B",    "off",       990
)
```



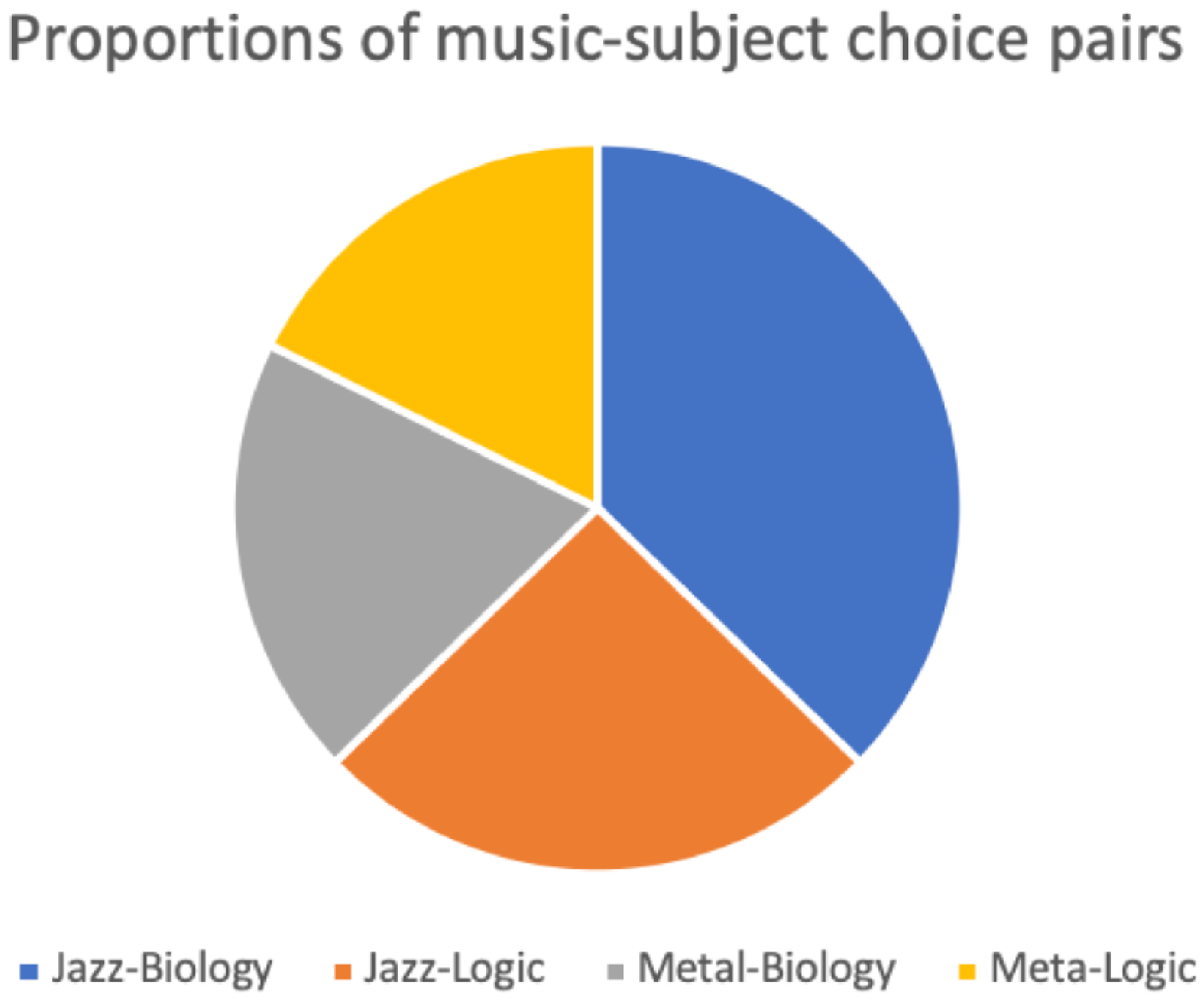
EXAMPLE OF UNINFORMATIVE PLOTTING

```
## # A tibble: 4 x 3
##   JM    LB          n
##   <chr> <chr>   <int>
## 1 Jazz  Biology    38
## 2 Jazz  Logic      26
## 3 Metal Biology    20
## 4 Metal Logic      18
```


Counts of music-subject choice pairs

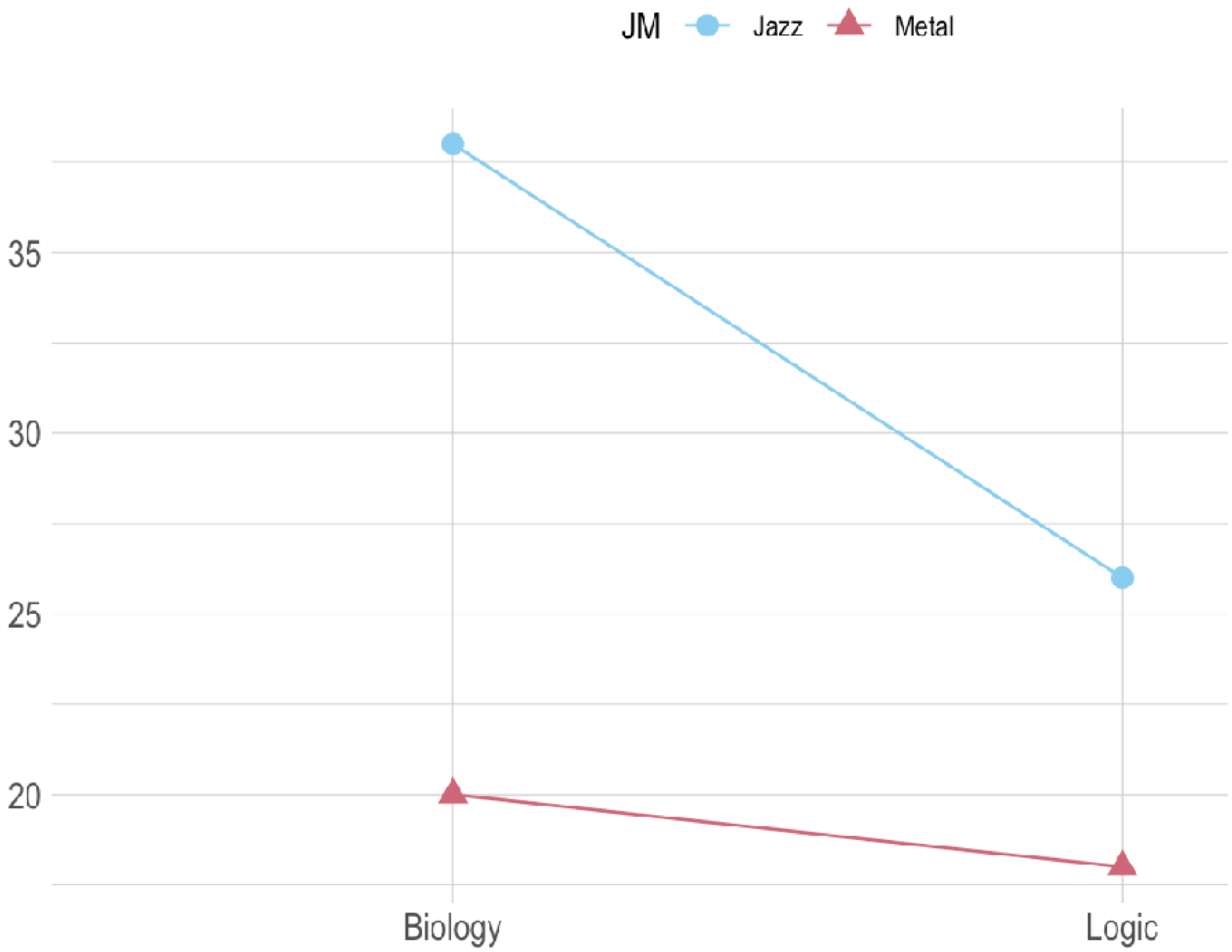
EXAMPLE OF UNINFORMATIVE PLOTTING

```
## # A tibble: 4 x 3
##   JM    LB          n
##   <chr> <chr>   <int>
## 1 Jazz  Biology    38
## 2 Jazz  Logic      26
## 3 Metal Biology    20
## 4 Metal Logic      18
```



EXAMPLE OF INFORMATIVE HYPOTHESIS-DRIVEN PLOTTING

```
## # A tibble: 4 x 3
##   JM    LB          n
##   <chr> <chr>   <int>
## 1 Jazz  Biology    38
## 2 Jazz  Logic      26
## 3 Metal Biology    20
## 4 Metal Logic      18
```


JM
● Jazz
▲ Metal
Biology
Logic