

Eric Pitman Summer Workshop in Computational Science



3. Descriptive Statistics



CENTER FOR **COMPUTATIONAL RESEARCH**



University at Buffalo
The State University of New York

Descriptive Statistics



Explore a dataset:

- What's in the dataset?
- What does it mean?
- What if there's *a lot* of it?

Basic Statistical Functions in R



Wanted: measures of the center and the spread of our numeric data.

- `mean()`
- `median()`
- `range()`
- `var()` and `sd()` **# variance, standard deviation**
- `summary()` **# combination of measures**

mean()



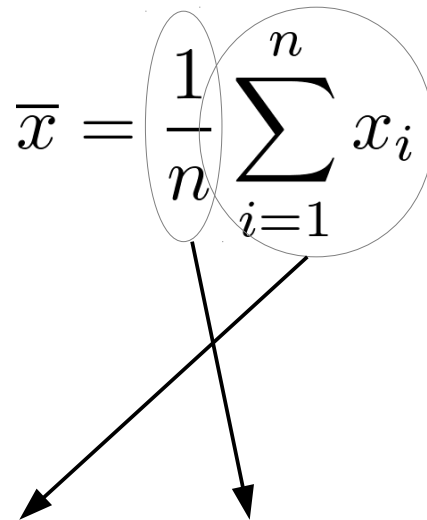
A measure of the data's “most typical” value.

- Arithmetic mean == average
- Divide sum of values by number of values

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

mean()

A measure of the data's “most typical” value.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$


```
> f <- c(3, 2, 4, 1)
```

```
> mean(f)    # == sum(f)/length(f) == (3+2+4+1)/4
```

```
[1] 2.5
```

median()

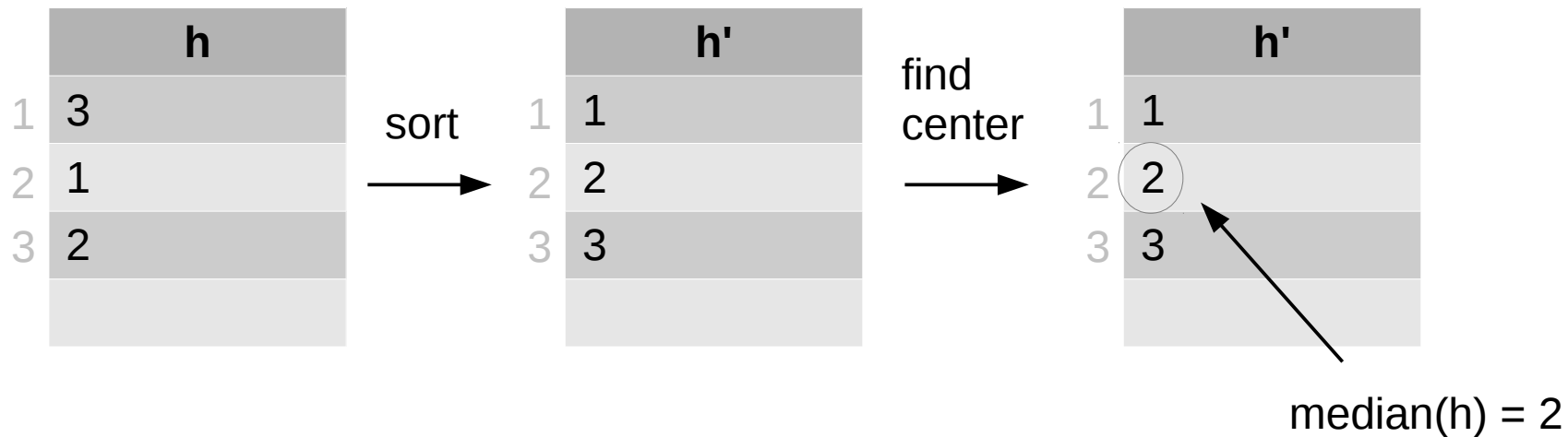


A measure of the data's center value. To find it:

- Sort the contents of the data structure
- Compute the value at the center of the data:
 - For odd number of elements, take the center element's value.
 - For even number of elements, take mean around center.

median()

Odd number of values:



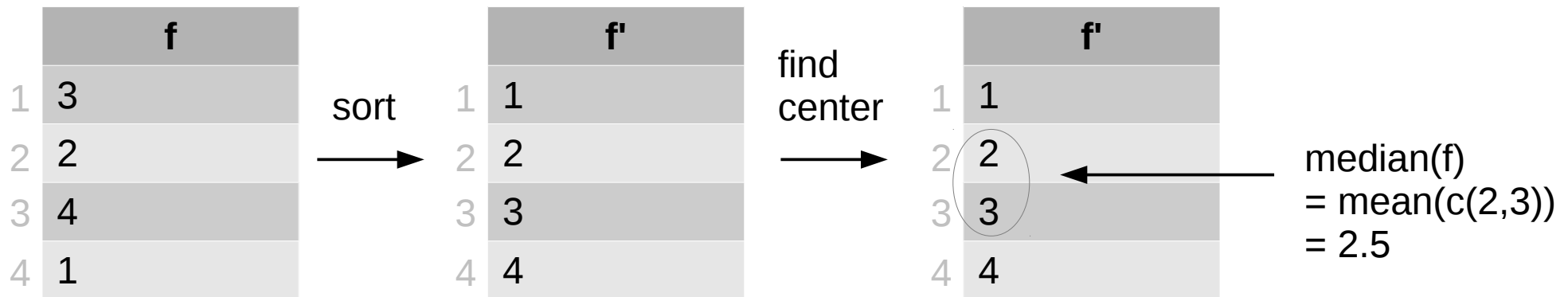
```
> h <- c(3, 1, 2)
```

```
> median(h)
```

```
[1] 2
```

median()

Even number of values: need to find mean()



```
> f <- c(3, 2, 4, 1)
```

```
> median(f)
```

```
[1] 2.50
```


range(): min() and max()



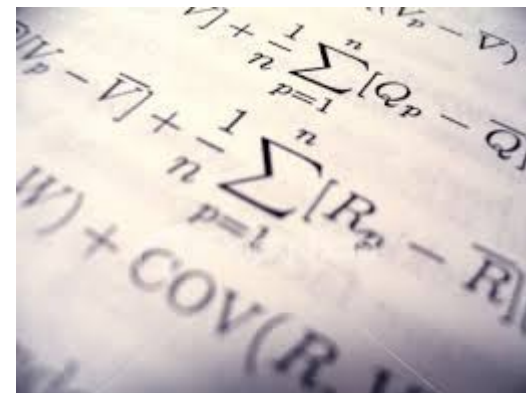
range() reports the minimum and maximum values found in the data structure.

```
> f <- c(3, 2, 4, 1)
```

```
> range(f) # reports min(f) and max(f)
```

```
[1] 1 4
```

var() and sd()



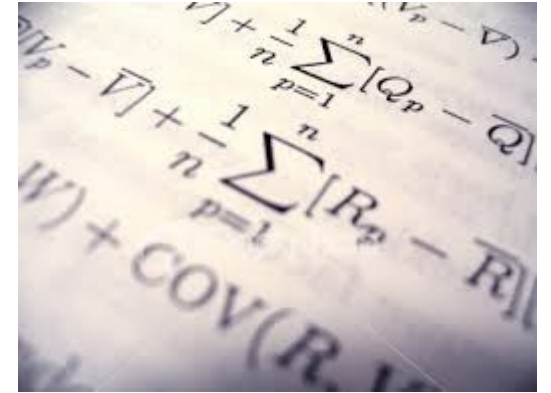
- *Variance*: a measure of the spread of the values relative to their mean:

$$Var = s_n^2 = \frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2 \quad \text{Sample variance}$$

- *Standard deviation*: square root of the variance

$$s_n = \sqrt{Var} \quad \text{Sample standard deviation}$$

R's summary() Function



Provides several useful descriptive statistics about the data:

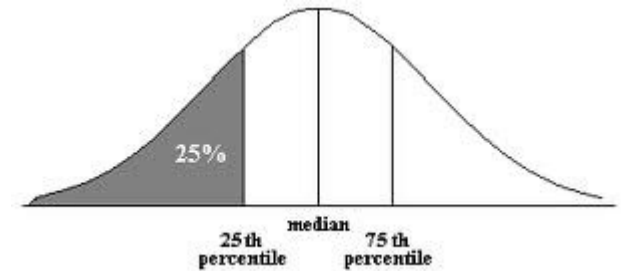
```
> g <- c(3, NA, 2, NA, 4, 1)
```

```
> summary(g)
```

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. | NA's |
|------|---------|--------|------|---------|------|------|
| 1.00 | 1.75 | 2.50 | 2.50 | 3.25 | 4.00 | 2 |

Quartiles: Sort the data set and divide it up into quarters...

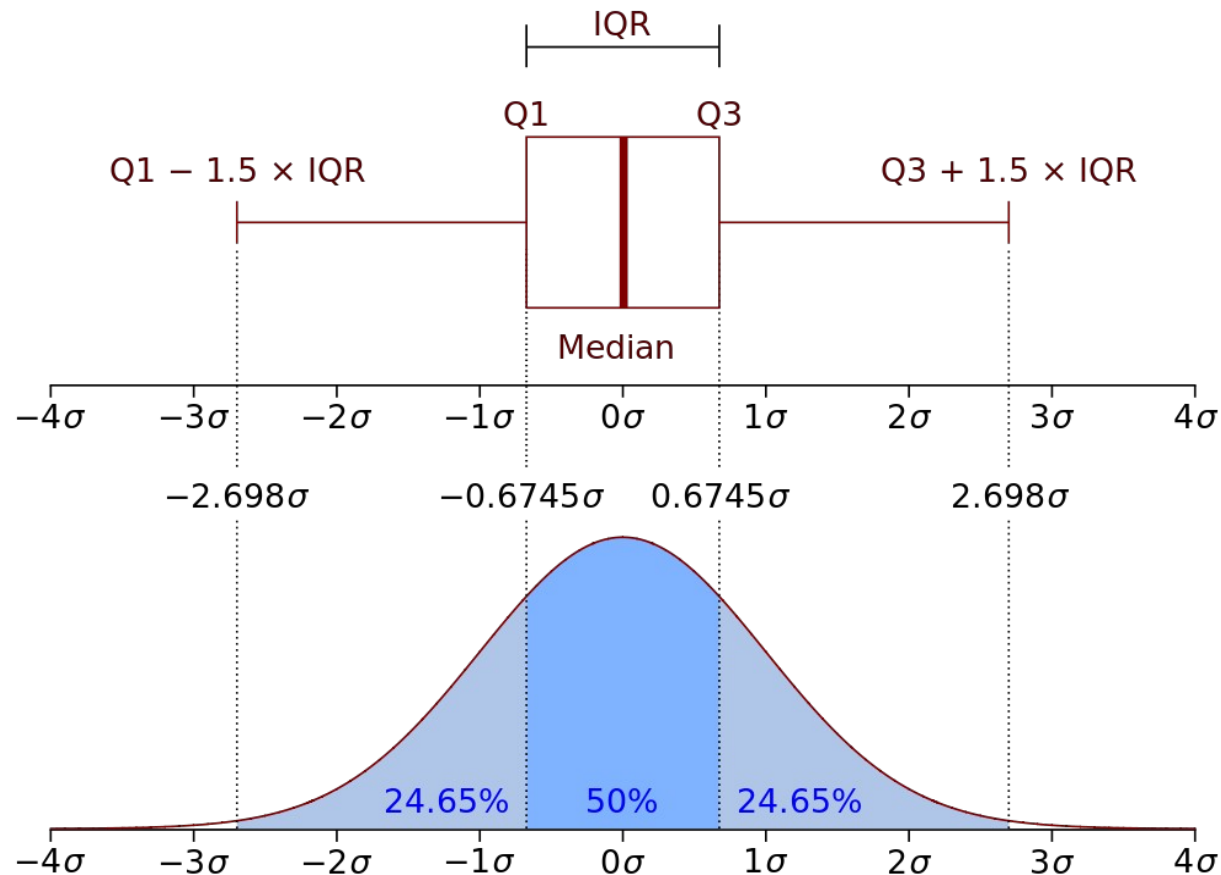
Quartiles



Quartiles are the *three points* that divide ordered data into four equal-sized groups:

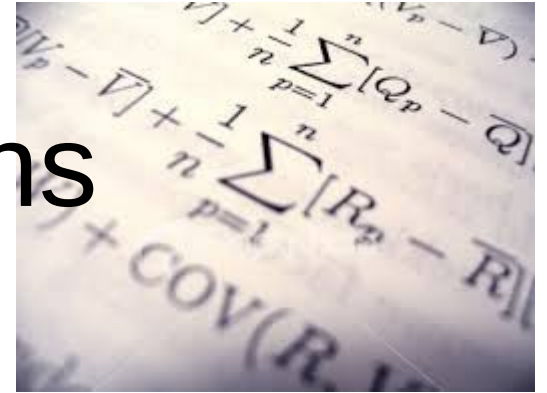
- Q1 marks the boundary just above the lowest 25% of the data
- Q2 (the *median*) cuts the data set in half
- Q3 marks the boundary just below the highest 25% of data

Quartiles



Boxplot and probability distribution function of Normal $N(0, 1\sigma^2)$ population

Summary: Basic Statistical Functions



- Characterize the center and the spread of our numeric data.
- Comparing these measures can give us a good sense of our dataset.

Statistics and Missing Data



If NAs are present, specify `na.rm=TRUE` to call:

- `mean()`
- `median()`
- `range()`
- `sum()`
- ...and some other functions

R disregards NAs, then proceeds with the calculation.

Diamonds Data



50,000 diamonds, for example:

| | carat | cut | color | clarity | depth | table | price | x | y | z |
|---|-------|---------|-------|---------|-------|-------|-------|------|------|------|
| 1 | 0.23 | Ideal | E | SI2 | 61.5 | 55 | 326 | 3.95 | 3.98 | 2.43 |
| 2 | 0.21 | Premium | E | SI1 | 59.8 | 61 | 326 | 3.89 | 3.84 | 2.31 |
| 3 | 0.23 | Good | E | VS1 | 56.9 | 65 | 327 | 4.05 | 4.07 | 2.31 |

What can we learn about these data?

Diamonds Data



This dataset is part of the `ggplot2` library.

To make the dataset accessible from the command line, just load the library:

```
> library("ggplot2")
```

```
> str(diamonds)
```

```
'data.frame': 53940 obs. of 10 variables:
```

```
$ carat   : num  0.23 0.21 0.23 0.29 0.31 ...
```

Diamonds Data summary()



Information provided by summary() depends on the type of data, by column:

| carat | cut | color | price |
|----------------|-----------------|---------|---------------|
| Min. :0.2000 | Fair : 1610 | D: 6775 | Min. : 326 |
| 1st Qu.:0.4000 | Good : 4906 | E: 9797 | 1st Qu.: 950 |
| Median :0.7000 | Very Good:12082 | F: 9542 | Median : 2401 |
| Mean :0.7979 | Premium :13791 | G:11292 | Mean : 3933 |
| 3rd Qu.:1.0400 | Ideal :21551 | H: 8304 | 3rd Qu.: 5324 |
| Max. :5.0100 | | I: 5422 | Max. :18823 |
| | | J: 2808 | |

numeric data:
statistical summary

categorical (factor) data:
counts

Diamond Price with Size: Scatter Plot

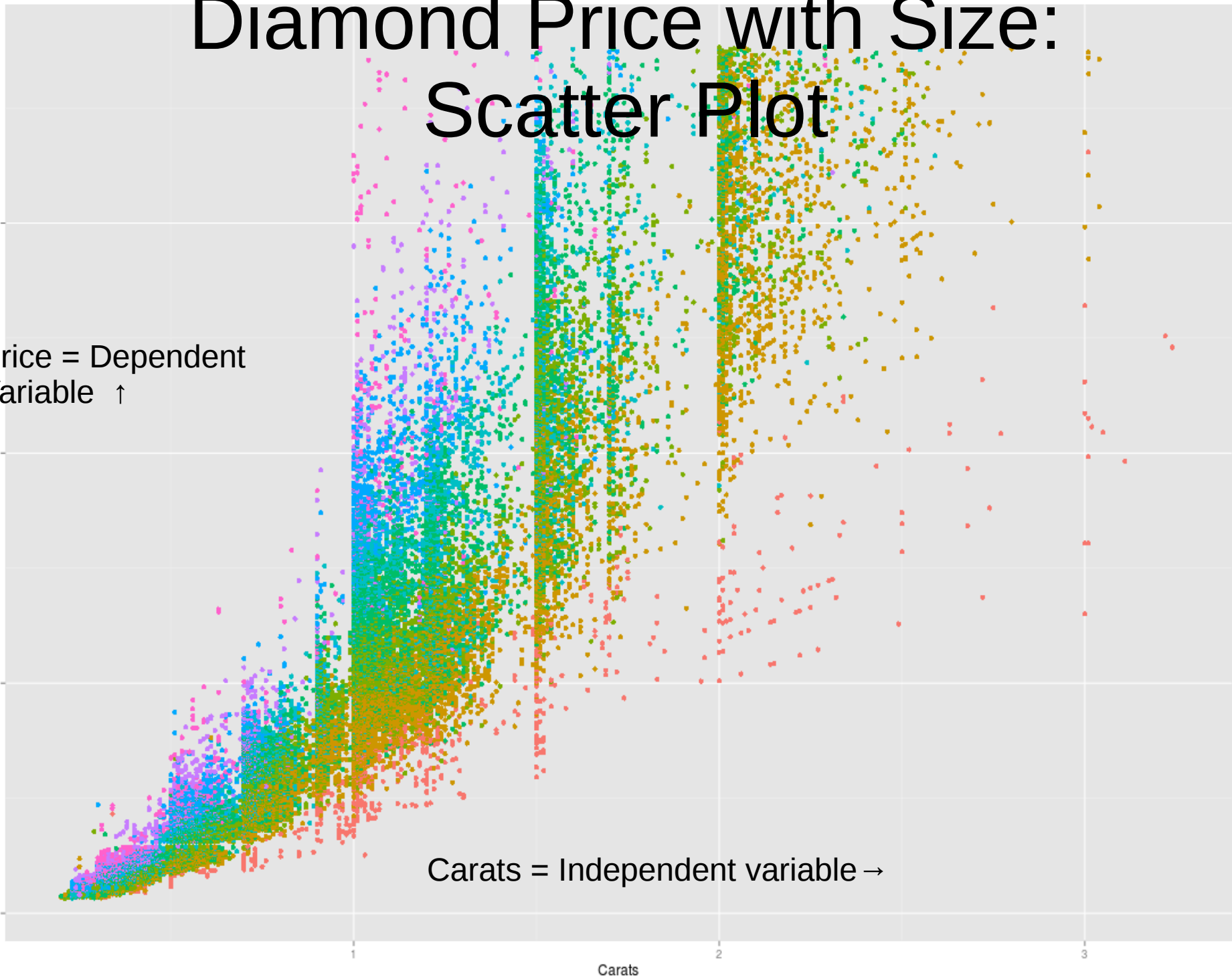
Price in Dollars

Price = Dependent
Variable ↑

Carats = Independent variable →

clarity

- I1
- SI2
- SI1
- VS2
- VS1
- WVS2
- WVS1
- IF



table() Function

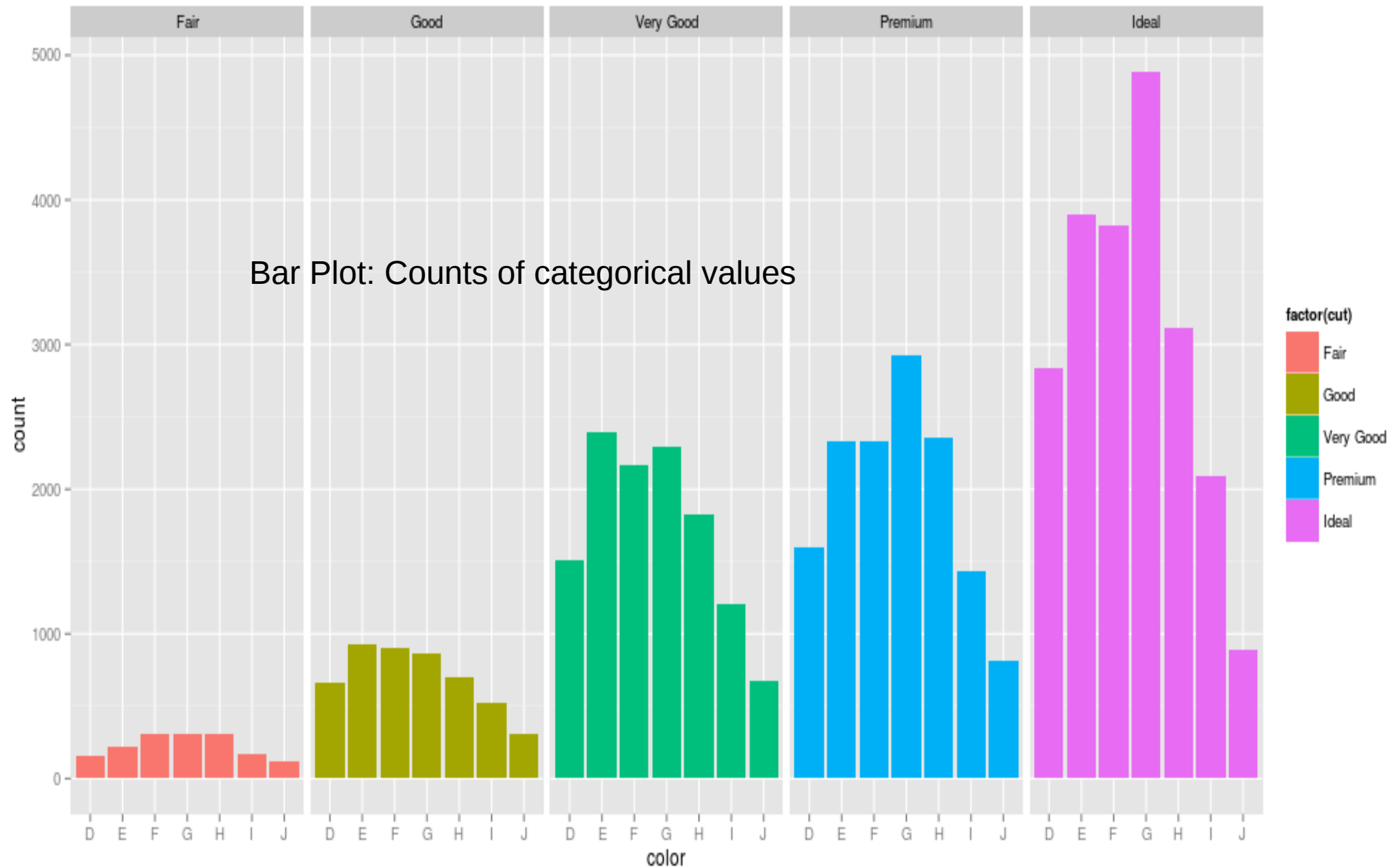


Contingency table: counts of categorical values for selected columns

```
> table(diamonds$cut, diamonds$color)
```

| | D | E | F | G | H | I | J |
|-----------|------|------|------|------|------|------|-----|
| Fair | 163 | 224 | 312 | 314 | 303 | 175 | 119 |
| Good | 662 | 933 | 909 | 871 | 702 | 522 | 307 |
| Very Good | 1513 | 2400 | 2164 | 2299 | 1824 | 1204 | 678 |
| Premium | 1603 | 2337 | 2331 | 2924 | 2360 | 1428 | 808 |
| Ideal | 2834 | 3903 | 3826 | 4884 | 3115 | 2093 | 896 |

Diamond Color and Cut



Correlation



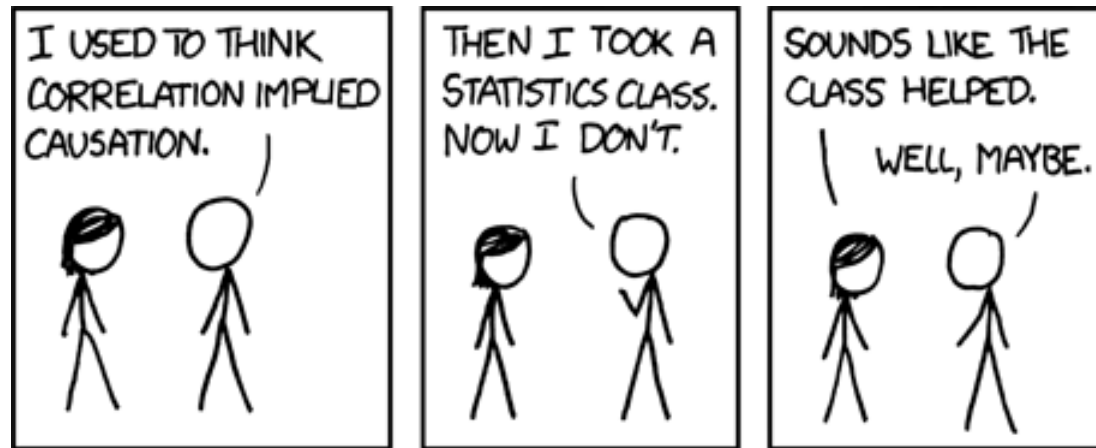
Do the two quantities X and Y vary together?

- Positively: $0 < \rho < 1$
- Or negatively: $-1 < \rho < 0$

$$\rho_{X,Y} = \text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

A pairwise, *statistical* relationship between quantities

Correlation



$$\rho_{X,Y} = \text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

NOTE: Correlation does not imply causation...

Looking for Correlations



diamonds data frame: 50,000 diamonds

- carat: weight of the diamond (0.2–5.01)
- table: width of top of diamond relative to widest point (43–95)
- price: price in US dollars
- x: length in mm (0–10.74)
- y: width in mm (0–58.9)
- z: depth in mm (0–31.8)

cor() Function



Look at pairwise, *statistical* relationships between numeric data:

```
> cor(diamonds[c(1,6:10)])
```

| | carat | table | price | x | y | z |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|
| carat | 1.0000000 | 0.1816175 | 0.9215913 | 0.9750942 | 0.9517222 | 0.9533874 |
| table | 0.1816175 | 1.0000000 | 0.1271339 | 0.1953443 | 0.1837601 | 0.1509287 |
| price | 0.9215913 | 0.1271339 | 1.0000000 | 0.8844352 | 0.8654209 | 0.8612494 |
| x | 0.9750942 | 0.1953443 | 0.8844352 | 1.0000000 | 0.9747015 | 0.9707718 |
| y | 0.9517222 | 0.1837601 | 0.8654209 | 0.9747015 | 1.0000000 | 0.9520057 |
| z | 0.9533874 | 0.1509287 | 0.8612494 | 0.9707718 | 0.9520057 | 1.0000000 |

-1.0: perfectly anticorrelated



0 : uncorrelated



1.0: perfectly correlated

Student Dataset Example



Now we can write some R to perform some descriptive statistics on our student data:

- Contingency table of school and handedness?
- Correlation between age and height?
- Some descriptive statistics about height (only comparing apples to apples!)

OK...but what's the problem here? Small dataset!

Interlude

Complete descriptive statistics exercises.



Open in the RStudio source editor:

`<workshop>/exercises/exercises-descriptive-statistics.R`