

# Exploring Data

Numerical Summaries

**STAT5002**

*The University of Sydney*

Mar 2025



THE UNIVERSITY OF  
**SYDNEY**

# Exploring Data

## Topic 1: Data & Graphical Summaries

What type of data do we have & how can we visualise it?

## Topic 2: Numerical Summaries

What are the main features of the data?

# Outline

## Centre

- Sample mean
- Sample median
- Robustness and comparisons

## Spread



- Standard deviation
- Interquartile range

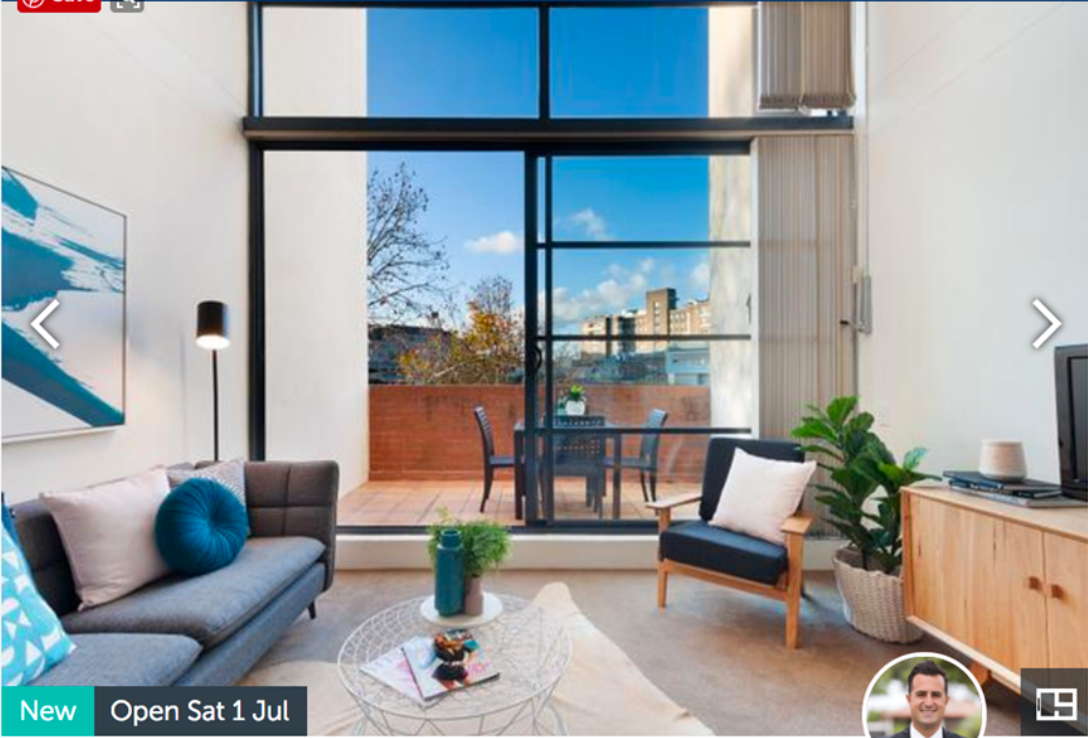
## Write functions in R

# Data story



How much does a property in Newtown cost?

cobden & hayson






New Open Sat 1 Jul



**Buyers Guide \$600-\$650k**  
Auction Sat 22 Jul  
205w/138 Carillon Avenue, Newtown, NSW 2042  
🏠 1 🚗 1 🚗 1

 Save   Details >

# Data on Newtown property sales

- Data is taken from [domain.com.au](https://domain.com.au):
  - ➡ All properties sold in Newtown (NSW 2042) between April-June 2017.
  - ➡ The variable `Sold` has price in \$1000s.

```
1 data <- read.csv("data/NewtownJune2017.csv", header = T)
2 head(data, n = 2)
```

	Property	Type	Agent	Bedrooms	Bathrooms	Carspots	Sold
1	19 Watkin Street	Newtown House	RayWhite	4	1	1	1975
2	30 Pearl Street	Newtown House	RayWhite	2	1	0	1250

Date

1	23/6/17
2	23/6/17

# Structure of Newtown data

```
1 dim(data)
```

```
[1] 56 8
```

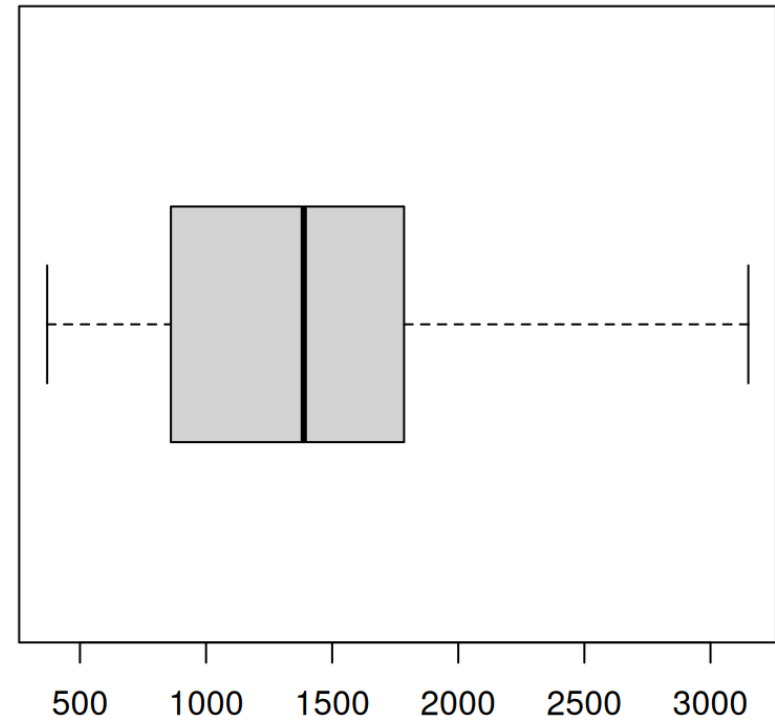
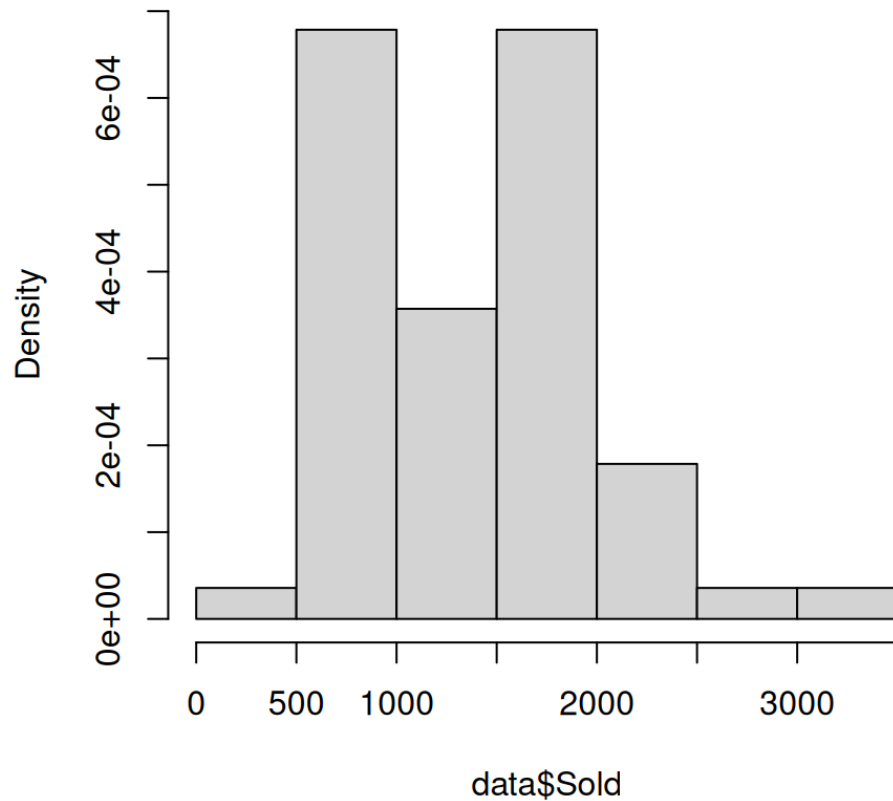
```
1 str(data)
```

```
'data.frame': 56 obs. of 8 variables:
 $ Property : chr "19 Watkin Street Newtown" "30 Pearl Street Newtown" "26 John Street Newtown" "23/617-623
King Street Newtown" ...
 $ Type : chr "House" "House" "House" "Apartment" ...
 $ Agent : chr "RayWhite" "RayWhite" "Belle" "RayWhite" ...
 $ Bedrooms : int 4 2 2 1 1 5 1 1 1 3 ...
 $ Bathrooms: int 1 1 1 1 1 1 1 1 1 2 ...
 $ Carspots : int 1 0 0 1 1 1 0 1 1 0 ...
 $ Sold : int 1975 1250 1280 780 650 2100 675 740 625 1950 ...
 $ Date : chr "23/6/17" "23/6/17" "17/6/17" "17/6/17" ...
```

# Graphical summaries

```
1 par(mfrow = c(1, 2))  
2 hist(data$Sold, freq = F)  
3 boxplot(data$Sold, horizontal = T)
```

**Histogram of data\$Sold**



# Numerical summaries

# Advantages of numerical summaries

- A numerical summary reduces all the data to one simple number (“statistic”).
  - ➡ This loses a lot of information.
  - ➡ However it allows easy communication and comparisons.
- Major features that we can summarise numerically are:
  - ➡ Maximum
  - ➡ Minimum
  - ➡ Centre [sample mean, median]
  - ➡ Spread [standard deviation, range, interquartile range]

## Note

Which summaries might be useful for talking about Newtown house prices?

- It depends!
- Reporting the centre without the spread can be misleading!

# Useful notation for data

- Observations of a single variable of size  $n$  can be represented by

$$x_1, x_2, \dots, x_n$$

- The ranked observations (ordered from smallest to largest) are

$$x_{(1)}, x_{(2)}, \dots, x_{(n)}$$

such that  $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$

- The sum of the observations are

$$\sum_{i=1}^n x_i$$

Sample mean

# Sample mean

The sample mean is the average of the data.

$$\text{sample mean} = \frac{\text{sum of data}}{\text{size of data}}$$

or

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

Note that the sample mean involves **all** of the data.

- The sample mean of all the properties sold in Newtown is:

```
1 mean(data$Sold)
```

```
[1] 1407.143
```

- Focusing specifically on houses with 4 bedrooms (large), the sample mean is:

```
1 mean(data$Sold[data$Type == "House" & data$Bedrooms == "4"])
```

```
[1] 2198.857
```

# Deviation from the mean

Given a data point  $x_i$ , its deviation from the sample mean  $\bar{x}$  is

$$D_i = x_i - \bar{x}$$

For example,

- 19 Watkin St sold for \$1950 (thousands).
  - ➡ This gives a gap of (\$1950-\$1407.143) = \$542.857 (thousands)
  - ➡ \$542.857 (thousands) **above** the sample mean
- 30 Pearl St sold for \$1250 (thousands).
  - ➡ This gives a gap of (\$1250-\$1407.143) = -\$157.143 (thousands)
  - ➡ \$157.143 (thousands) **below** the sample mean

# Sample mean as a balancing point

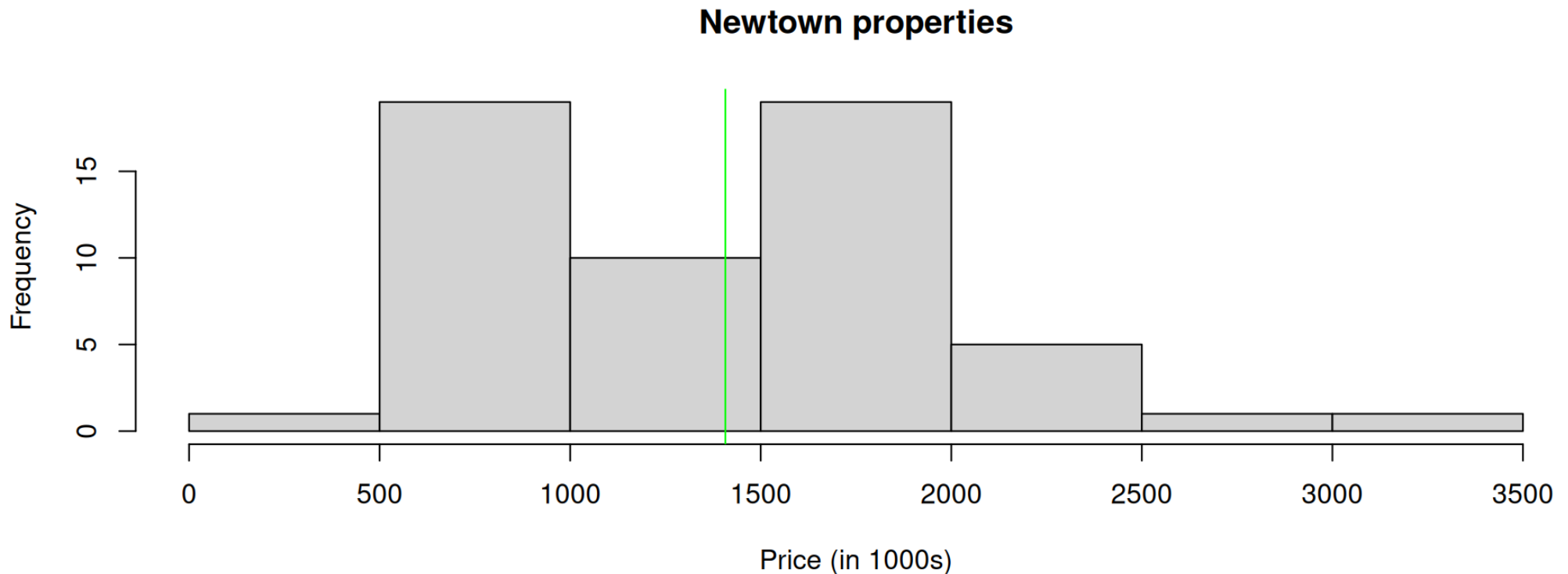
The sample mean is the point at which the data is **balanced** in the sense the sum of the **absolute deviations** for values to the left of the mean is the same as the sum of absolute deviations to the right of the mean.

$$\sum_{x_i < \bar{x}} |x_i - \bar{x}| = \sum_{x_i > \bar{x}} |x_i - \bar{x}|$$

# Sample mean on the histogram

However, sample mean may **not** be balancing point of a histogram, the area to the left of the mean may not be the same as the area to the right of the mean.

```
1 hist(data$Sold, main = "Newtown properties", xlab = "Price (in 1000s)")
2 abline(v = mean(data$Sold), col = "green")
```

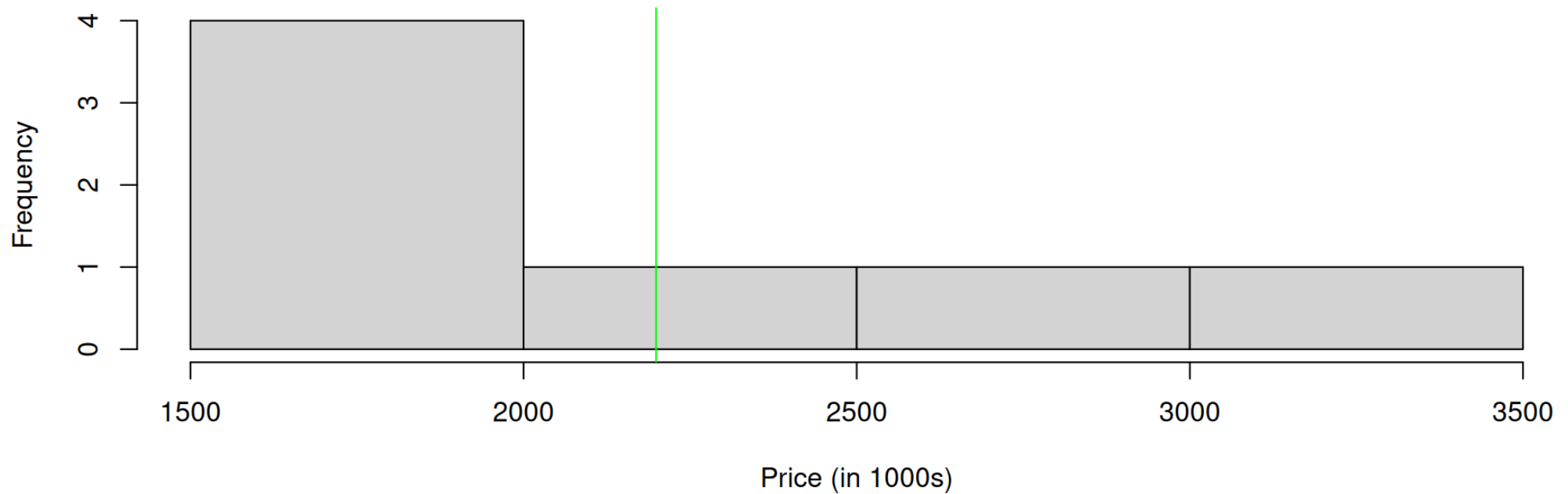


# Skewed data

When the data is skewed, this effect is more significant.

```
1 hist(data$Sold[data$Type = "House" & data$Bedrooms = "4"], main = "Newtown 4 Bedrooms",  
2     xlab = "Price (in 1000s)")  
3 abline(v = mean(data$Sold[data$Type = "House" & data$Bedrooms = "4"]), col = "green")
```

**Newtown 4 Bedrooms**



Sample median

# Sample median

The sample median  $\tilde{x}$  is the **middle data point**, when the observations are ordered from smallest to largest.

- For an odd sized number of observations:

$$\text{sample median} = \text{the unique middle point} = x_{(\frac{n+1}{2})}$$

- For an even sized number of observations:

$$\text{sample median} = \text{average of the 2 middle points} = \frac{x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)}}{2}$$

# Ordering observations

The ordered observations are:

```
1 sort(data$Sold)

[1] 370 625 645 650 675 692 720 740 740 755 770 780 812 860 861
[16] 920 935 955 955 999 1100 1240 1250 1280 1309 1315 1370 1375 1400 1460
[31] 1553 1575 1590 1600 1600 1600 1605 1662 1701 1710 1750 1780 1790 1806 1850
[46] 1940 1950 1975 2000 2100 2200 2235 2300 2410 2810 3150

1 length(data$Sold)

[1] 56
```

As we have  $n = 56$  observations (even), the sample median is found between the  $(\frac{n}{2}) = 28\text{th}$  and  $(\frac{n}{2} + 1) = 29\text{th}$  prices, or  $\frac{1375+1400}{2} = 1387.5$ .

- The sample median of all the properties sold in Newtown is:

```
1 median(data$Sold)
```

```
[1] 1387.5
```

- Focusing specifically on houses with 4 bedrooms (large), the sample median is:

```
1 median(data$Sold[data$Type == "House" & data$Bedrooms == "4"])
```

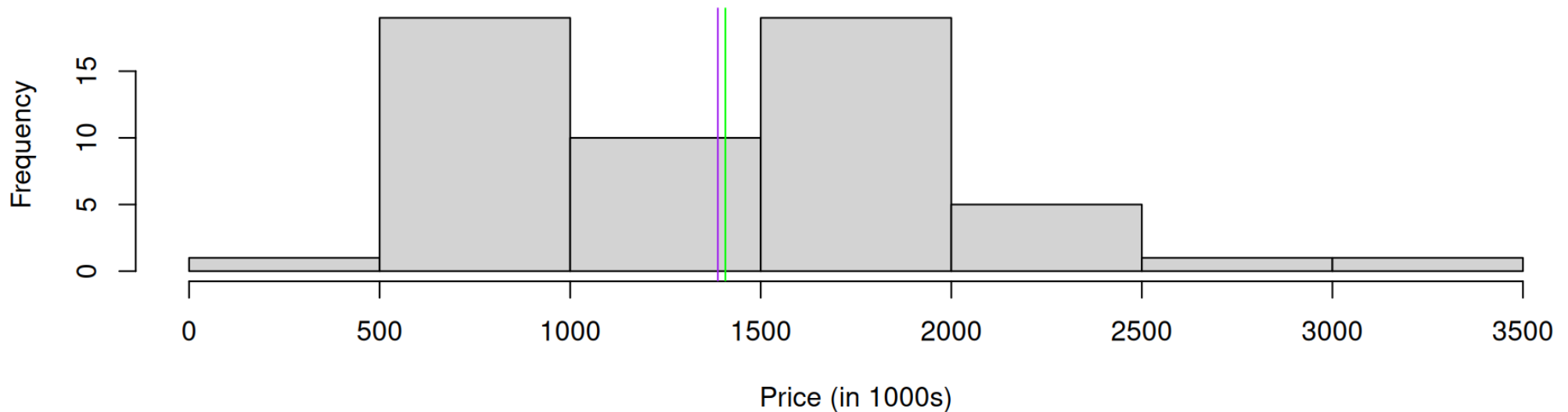
```
[1] 1975
```

# Sample median on the histogram

- The sample median is the **half way point** on the histogram - i.e., 50% of the houses sold are below and above \$1.3875 million.

```
1 hist(data$Sold, xlab = "Price (in 1000s)")
2 abline(v = mean(data$Sold), col = "green") # create a green line for the mean
3 abline(v = median(data$Sold), col = "purple") # create a purple line for the median
```

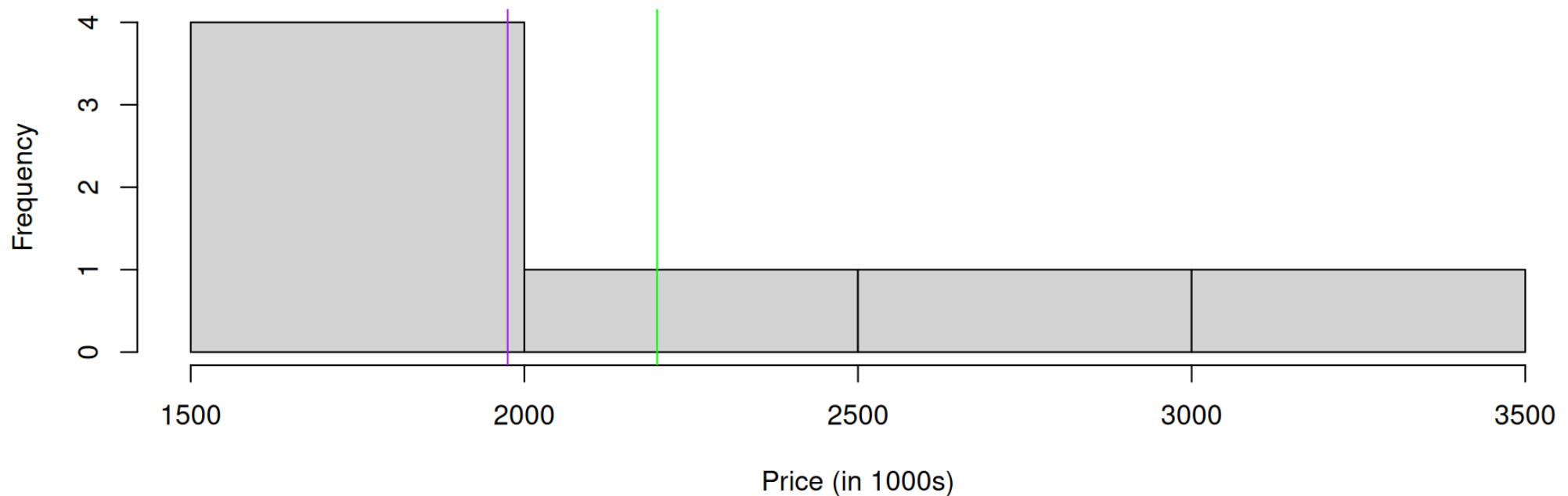
Histogram of data\$Sold



# Histogram for 4 Bedroom Houses

```
1 hist(data$Sold[data$Type = "House" & data$Bedrooms = "4"], main = "Newtown 4 Bedrooms",  
2       xlab = "Price (in 1000s)")  
3 abline(v = mean(data$Sold[data$Type = "House" & data$Bedrooms = "4"]), col = "green")  
4 abline(v = median(data$Sold[data$Type = "House" & data$Bedrooms = "4"]), col = "purple")
```

**Newtown 4 Bedrooms**



# Comparison between sample mean and median

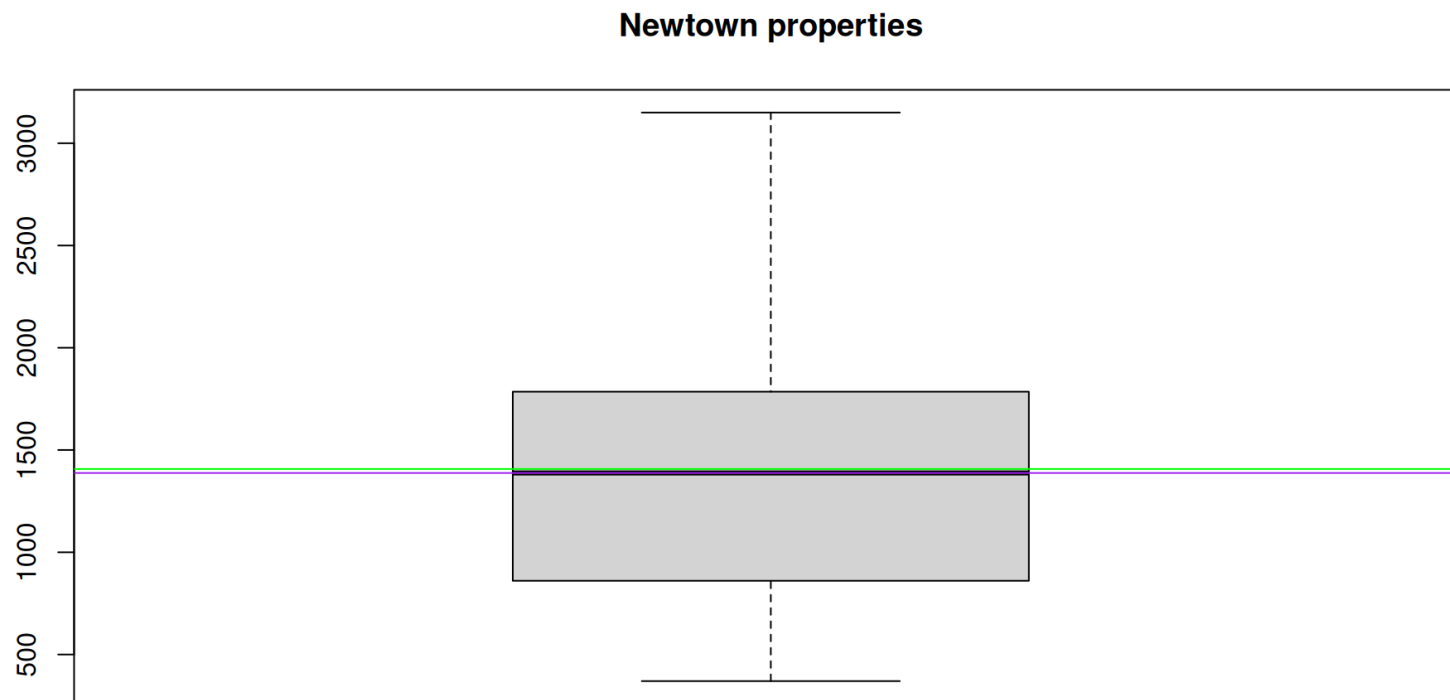
If you had to choose between reporting the sample mean or sample median for Newtown properties, which would you choose and why?

- For the full property portfolio, the sample mean and the sample median are fairly similar.
- For the 4 bedroom houses, the sample mean is higher than the sample median because it is being “pulled up” by some very expensive houses.
- For the average buyer, the sample median would be more useful as an indication of the sort of price needed to get into the market.
- For any agent selling houses in the area, the sample mean might be more useful in order to predict their average commissions!
- In practice, we can report both!

# Sample mean and median on the boxplot

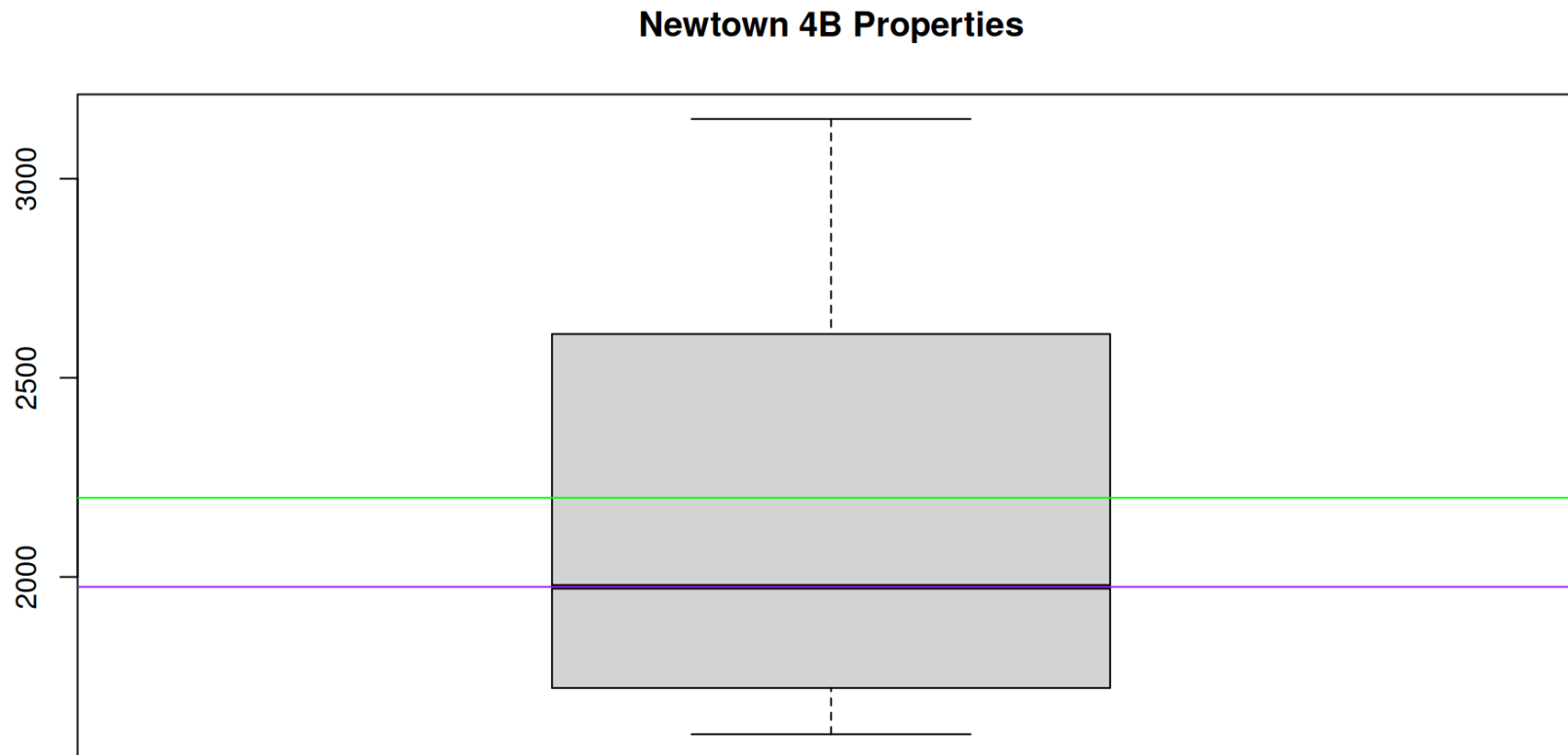
The sample median is the centre line on the boxplot.

```
1 boxplot(data$Sold, main = "Newtown properties")
2 abline(h = mean(data$Sold), col = "green")
3 abline(h = median(data$Sold), col = "purple")
```



# Boxplot for 4 Bedroom Houses

```
1 boxplot(data$Sold[data$Type == "House" & data$Bedrooms == "4"], main = "Newtown 4B Properties")
2 abline(h = mean(data$Sold[data$Type == "House" & data$Bedrooms == "4"]), col = "green")
3 abline(h = median(data$Sold[data$Type == "House" & data$Bedrooms == "4"]), col = "purple")
```



# Robustness and comparisons

The sample median is said to be **robust** and is a good summary for skewed data as it is not affected by **outliers** (extreme data values).

# Example

Recently a heritage building was sold for 13 million in Newtown.



## *i* Note

How would the sample mean and sample median change if it was added to the data?

- The sample mean would be a lot higher.
- The sample median would be a bit higher: it moves from the average of the 28th and 29th points to the 29th point.

# Adding in a large outlier

```
1 data2 = c(data$Sold, 13000)
2 sort(data2)

[1] 370 625 645 650 675 692 720 740 740 755 770 780
[13] 812 860 861 920 935 955 955 999 1100 1240 1250 1280
[25] 1309 1315 1370 1375 1400 1460 1553 1575 1590 1600 1600 1600
[37] 1605 1662 1701 1710 1750 1780 1790 1806 1850 1940 1950 1975
[49] 2000 2100 2200 2235 2300 2410 2810 3150 13000

1 mean(data2)

[1] 1610.526

1 median(data2)

[1] 1400
```

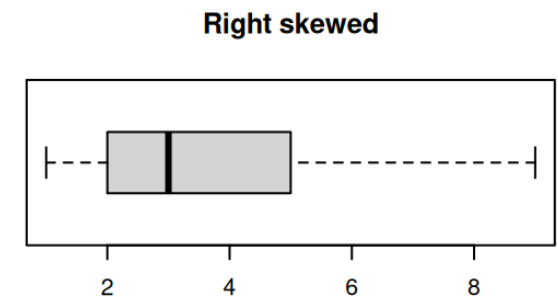
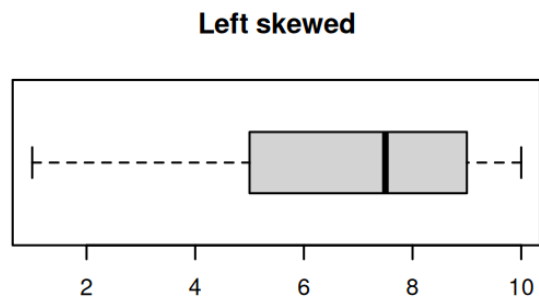
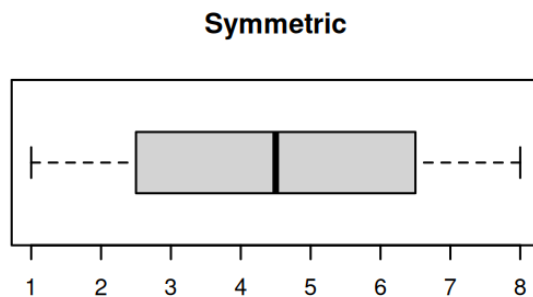
## Summary of changes

Change in data	Sample mean	Sample median
Original data	1407.143	1387.5
Extra property of 13000	1610.526	1400

# Skewness

The difference between the sample mean and the sample median can be an indication of the **shape** of the data.

- For symmetric data, the sample mean and sample median are the same:  $\bar{x} = \tilde{x}$ .
- For left skewed data (the most frequent data are concentrated on the right, with a left tail), the sample mean is smaller than the sample median:  $\bar{x} < \tilde{x}$ .
- For right skewed data (the most frequent data are concentrated on the left, with a right tail), the sample mean is larger than the sample median:  $\bar{x} > \tilde{x}$ .



# Which is optimal for describing centre?

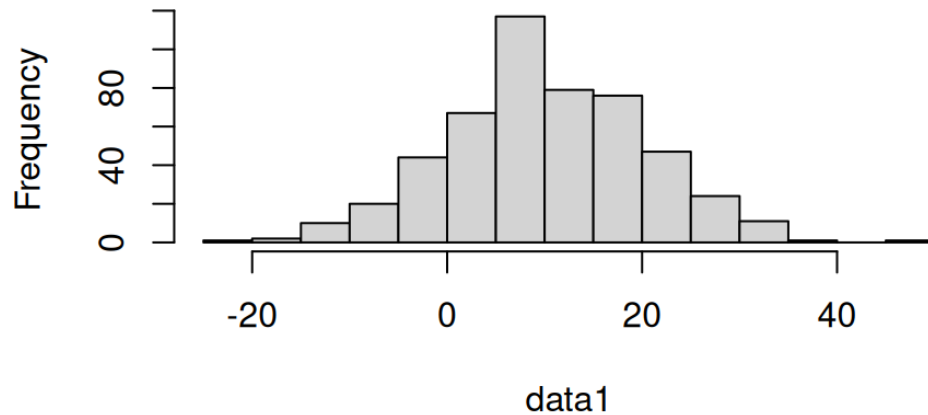
- Both have strengths and weaknesses depending on the nature of the data.
- Sometimes neither gives a sensible sense of location, for example if the data is **bimodal**.
- As the **sample median is robust**, it is preferable for data which is skewed or has many outliers, like Sydney house prices.
- The **sample mean** is helpful for data which is **basically symmetric**, with not too many outliers, and for theoretical analysis.

# Limitations of both?

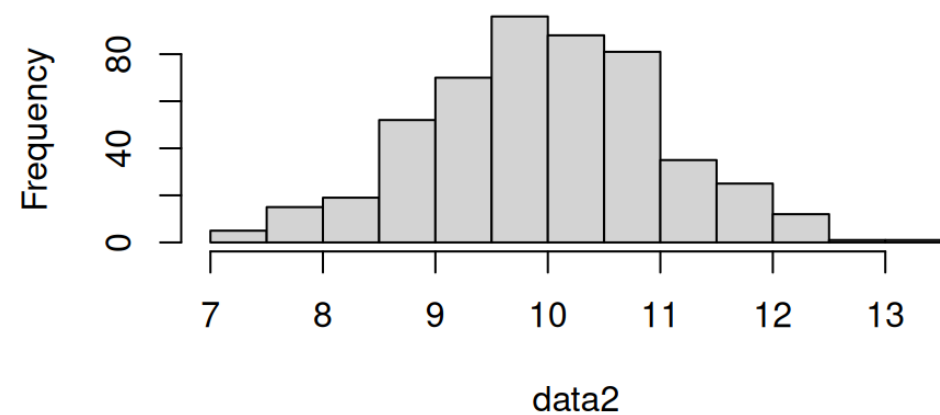
- Both the sample mean and sample median allow easy comparisons.
- However, they need to be paired with a measure of **spread**.
- In the following example, the sample means are the same, but the data are very different.

Or, consider two data sets  $\{-1, 0, 1\}$  and  $\{-100, 0, 100\}$ .

**Histogram of data1**



**Histogram of data2**



```
[1] 10.22644  9.95406
```

# Standard deviation

# How to measure spread?

For each property sold, we could calculate the **deviation** (or the gap) from the sample mean,  $D_i = x_i - \bar{x}$ , between the house and the sample mean \$1407 (thousands).

Property	Sold	Gap	Conclusion
19 Watkin Street	\$1950 (thousands)	1950-1407=543	More than half a million dollars more expensive than the average house price
30 Pearl St	\$1250 (thousands)	1250-1407=-157	Cheaper than the average house price

# Deviations in the dataset

```
1 gaps = data$Sold - mean(data$Sold)
2 gaps
```

```
[1] 567.857143 -157.142857 -127.142857 -627.142857 -757.142857
[6] 692.857143 -732.142857 -667.142857 -782.142857 542.857143
[11] -32.142857 167.857143 -408.142857 -452.142857 -547.142857
[16] 197.857143 182.857143 -167.142857 -1037.142857 532.857143
[21] -687.142857 -452.142857 -487.142857 442.857143 192.857143
[26] -652.142857 -7.142857 145.857143 1402.857143 192.857143
[31] 792.857143 372.857143 398.857143 293.857143 -98.142857
[36] -307.142857 -472.142857 -762.142857 52.857143 -37.142857
[41] -715.142857 1742.857143 1002.857143 -637.142857 254.857143
[46] 827.857143 592.857143 382.857143 342.857143 302.857143
[51] 192.857143 -546.142857 -667.142857 -92.142857 892.857143
[56] -595.142857
```

```
1 max(gaps)
```

```
[1] 1742.857
```

## Note

What are the biggest and smallest deviations?

How do we **summarise** all the deviations into **1 number** ("spread")?

# 1st attempt: The mean gap

We could calculate the **average** of the deviations.

$$\text{mean deviation} = \text{sample mean}(\text{data} - \text{sample mean}(\text{data}))$$

```
1 round(mean(gaps))
```

```
[1] 0
```

## Note

What's the problem?

Note: It will always be 0.

- From the definition, the mean deviation must be 0, as the mean is the **balancing point** of the deviations.
- The mean deviation is

$$\frac{\sum_{i=1}^n D_i}{n} = \frac{\sum_{i=1}^n (x_i - \bar{x})}{n} = \frac{\sum_{i=1}^n x_i}{n} - \frac{n\bar{x}}{n} = 0.$$

## Better option: Standard deviation

Standard deviation is a measure of spread that is based on the average.

First define the **root mean square** (RMS).

- The RMS measures the **average** of a set of numbers, regardless of the signs.
- The steps are: *Square* the numbers, then *Mean* the result, then *Root* the result.

$$\text{RMS}(\text{numbers}) = \sqrt{\text{sample mean}(\text{numbers}^2)}$$

- So effectively, the *Square* and *Root* operations “reverse” each other.
- RMS retain the same unit as the unit of the sample mean.

- Applying RMS to the deviations, we get

$$\text{RMS of deviations} = \sqrt{\text{sample mean (deviations}^2)} = \sqrt{\frac{\sum_{i=1}^n D_i^2}{n}}$$

- To avoid the cancellation of the deviations, another possible method is to consider the average of the absolute values of the deviations:

$$\text{mean absolute deviation (MAD)} = \frac{\sum_{i=1}^n |D_i|}{n}.$$

However, MAD is much harder to analyse.

# Standard deviation in terms of RMS

## Population Standard deviation

- The standard deviation measures the **spread** of the data.

$$SD_{pop} = \text{RMS of (deviations from the mean)}$$

- Formally,  $SD_{pop} = \sqrt{\text{Mean of (deviations from the mean)}^2} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$

```
1 sqrt(mean(gaps^2))
```

```
[1] 593.7166
```

# Standard deviation in R?

It is easy to calculate in R.

```
1 sd(data$Sold)
```

```
[1] 599.0897
```

## Note

But why is this slightly different?

# Adjusting the standard deviation

- There are **two** different formulas for the standard deviation, depending on whether the data is the **population** or a **sample**.
- The `sd` command in R always gives the **sample** version, as we most commonly have samples.
- Formally,  $SD_{pop} = \sqrt{\frac{1}{n} \sum_{i=1}^n D_i^2}$  and  $SD_{sample} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n D_i^2}$ , where  $D_i = x_i - \bar{x}$  is the deviation.

```
1 sd(data$Sold) * sqrt(55/56) # adjust by sqrt((n-1)/n), it calculates the population SD.
```

```
[1] 593.7166
```

```
1 gaps = data$Sold - mean(data$Sold) # calculate the gaps
2 sqrt(mean(gaps^2)) # calculates the population SD.
```

```
[1] 593.7166
```

Why does the sample SD use the adjustment  $\sqrt{(n-1)/n}$ ?

- It is an **unbiased estimator** of the standard deviation (beyond the scope of this unit, will be covered in Year 2)
- Estimating the sample mean uses all of the  $n$  data points. The sum (or the mean) of  $n$  deviations is zero

$$\sum_{i=1}^n D_i = \sum_{i=1}^n (x_i - \bar{x}) = 0.$$

This means, given the first  $n - 1$  deviations, we know the  $n$ -th deviation, because

$$\left( \sum_{i=1}^{n-1} D_i \right) + D_n = 0 \quad \implies \quad D_n = - \sum_{i=1}^{n-1} D_i.$$

Hence, there are only  $n - 1$  effective pieces of information in the deviations.

## Summary: population and sample

Summary	Formula	In R
Population or Sample mean	Sample Mean (Average)	<code>mean(data)</code>
Population standard deviation $SD_{pop}$	RMS of gaps from the sample mean	<code>sd(data)*sqrt((n-1)/n)</code>
Sample standard deviation $SD_{sample}$	Adjusted RMS of gaps from the sample mean	<code>sd(data)</code>

- The population standard deviation is always smaller than a sample standard deviation, ( $\{pop\}$   $\{sample\}$ ), why? Extra variability due to sampling.
- Note for large sample sizes, the difference becomes negligible.

# How to tell the difference?

- It can be tricky to work out whether your data is a population or sample!
- Look at the information about the data story and the research questions.
  - ⇒ If we are just interested in the Newtown property prices during April-June 2017, then the **data** is the whole **population**.
  - ⇒ If we are studying the property prices during April-June 2017 as a window into more general property prices (for the rest of the year or for the Inner West area) , then the **data** could be considered a **sample**.
- Population SD and sample SD get closer with increasing sample size  $n$ .

# Variance

The squared standard deviation is called the **variance**. Similar to the sample SD and the population SD, there are two versions of the variance

$$\text{Var}_{\text{sample}} = \text{SD}_{\text{sample}}^2 \quad \text{and} \quad \text{Var}_{\text{pop}} = \text{SD}_{\text{pop}}^2.$$

- For summarising spread, we often prefer SD, as it has the same unit as the data points and the mean.
- In some situations, e.g., dealing with random variables and understanding the property of sample mean, using the variance can be much simpler.

# Standard units (“Z score”)

Standard units of a data point = how many standard deviations is it below or above the mean

$$\text{standard units} = \frac{\text{data point} - \text{mean}}{\text{SD}}$$

This means that

$$\text{data point} = \text{mean} + \text{SD} \times \text{standard units}$$

It gives the relative location of a data point in the data set. It also have other benefits in data modelling (see later lectures).

# Comparing 2 data points

To compare 2 data points, we can compare the standard units.

Property	Sold	Standard units	Conclusion
19 Watkin Street	\$1950 (thousands)	$\frac{1950-1407}{599} = 0.91$	Almost 1 SD higher than the average house price
30 Pearl St	\$1250 (thousands)	$\frac{1250-1407}{599} = -0.26$	0.26 SDs cheaper than the average house price

So 19 Watkin is a more unusual purchase than 30 Pearl St, relative to the mean.

# Interquartile range

# Interquartile range (IQR)

The IQR is another measure of spread by **ordering** the data.

**IQR = range of the middle 50% of the data**

More formally, **IQR** =  $Q_3 - Q_1$ , where

- $Q_1$  is the 25-th percentile (1st quartile) and  $Q_3$  is the 75-th percentile (3rd quartile).
- The median is the 50-th percentile, or 2nd quartile  $\tilde{x} = Q_2$ .
- p-th percentile: there are p% of **ordered** data below the value of p-th percentile.

# Quantile, quartile, percentile

The set of ***q*-quantiles** divides the **ordered** data into ***q*** equal size sets (in terms of percentage of data).

**Percentile** is 100-quantile, so the set of percentiles divides the data into 100 equal parts.

The set of **quartiles** divides the data into four quarters.

```
1 summary(data$Sold)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
370.0	860.8	1387.5	1407.1	1782.5	3150.0

```
1 summary(data$Sold)[5] - summary(data$Sold)[2] # one way to calculate IQR
```

```
3rd Qu.  
921.75
```

```
1 IQR(data$Sold) # use the built-in function
```

```
[1] 921.75
```

So the range of the middle 50% of properties sold is almost a million dollars!

# Reporting

- Like the median, the IQR is **robust**, so it's suitable as a summary of spread for skewed data.
- We report in pairs: (mean,SD) or (median,IQR).

# IQR on the boxplot and outliers

- The IQR is the length of the box in the boxplot. It represents the span of the middle 50% of the houses sold.
- The **lower** and **upper thresholds** (expected minimum and maximum) are a distance of  $1.5IQR$  from the 1st and 3rd quartiles (by **Tukey**'s convention).

$$LT = Q_1 - 1.5IQR$$

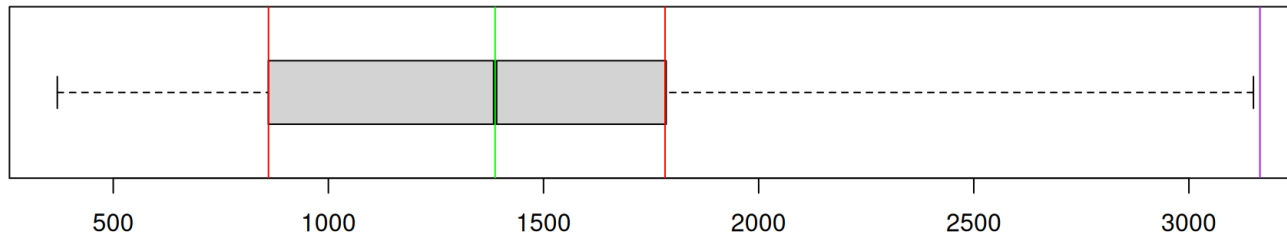
and

$$UT = Q_3 + 1.5IQR$$

- Data outside these thresholds is considered an **outlier** (“extreme reading”).

# Lower and Upper Thresholds on the Boxplot

```
1 boxplot(data$Sold, horizontal = T)
2 iqr = quantile(data$Sold)[4] - quantile(data$Sold)[2]
3 abline(v = median(data$Sold), col = "green")
4 abline(v = quantile(data$Sold)[2], col = "red")
5 abline(v = quantile(data$Sold)[4], col = "red")
6 abline(v = quantile(data$Sold)[2] - 1.5 * iqr, col = "purple")
7 abline(v = quantile(data$Sold)[4] + 1.5 * iqr, col = "purple")
```

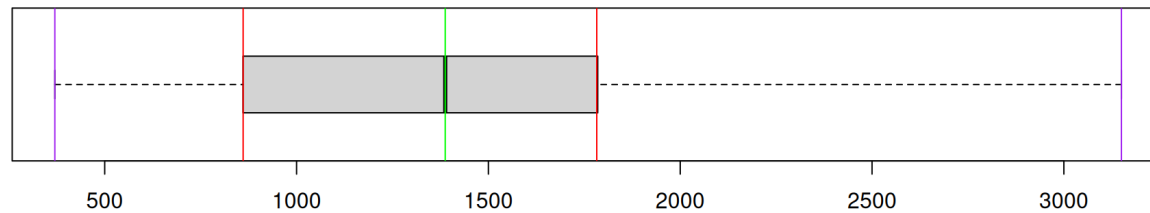


Note the lower threshold is not shown...why?

- The lower threshold should be the same difference from Q1 as the upper threshold is from Q3.
- So the lower threshold would be  $< 0$ , which is not possible as we are dealing with house prices.

# Thresholds can be outside of the data's range

```
1 boxplot(data$Sold, horizontal = T)
2 abline(v = median(data$Sold), col = "green")
3 abline(v = quantile(data$Sold)[2], col = "red")
4 abline(v = quantile(data$Sold)[4], col = "red")
5 abline(v = max(min(data$Sold), quantile(data$Sold)[2] - 1.5 * iqr), col = "purple")
6 abline(v = min(max(data$Sold), quantile(data$Sold)[4] + 1.5 * iqr), col = "purple")
```



To make the LT and UT staying within the range of data, R uses the convention

$$LT = \max(\min(x), Q_1 - 1.5IQR)$$

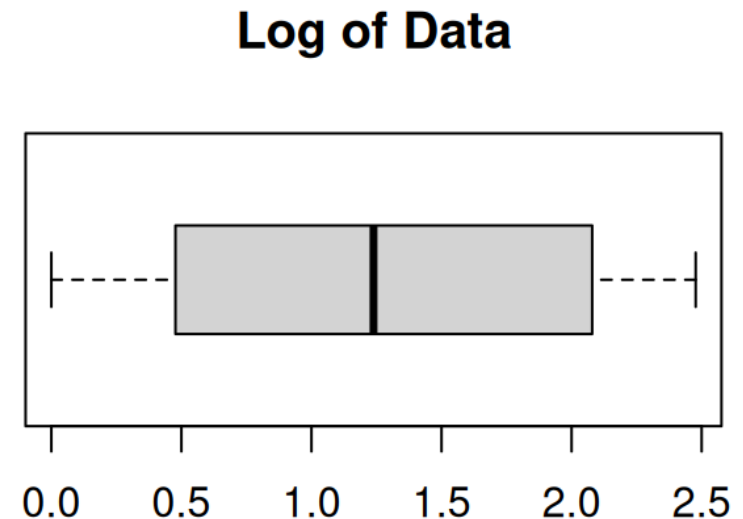
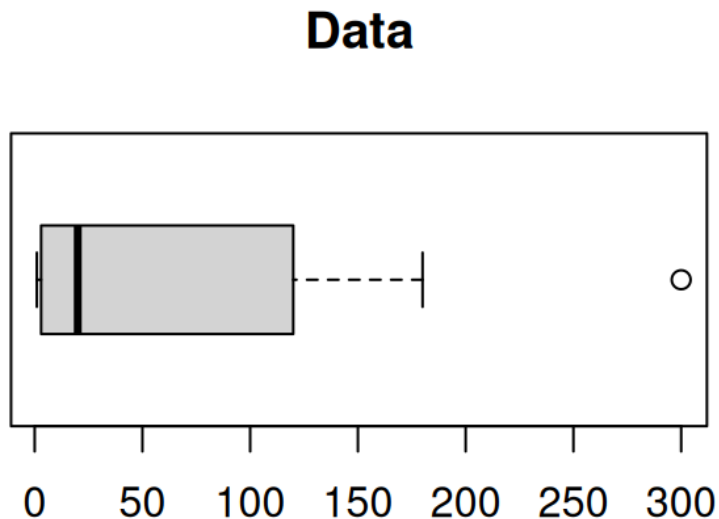
and

$$UT = \min(\max(x), Q_3 + 1.5IQR)$$

# Dealing with outliers

Sometimes outliers indicate that a better model is needed. We may remove outliers by transforming the data. For example, a right skewed data set with outliers can be transformed into the logarithmic scale.

```
1 w = c(1, 2, 3, 4, 10, 30, 60, 120, 180, 300)
2 w1 = log(w, 10)
3 par(mfrow = c(1, 2))
4 boxplot(w, main = "Data", horizontal = T)
5 boxplot(w1, main = "Log of Data", horizontal = T)
```



Write a function in R

# How to write a function in R

A function in R is one of the most used objects. For example, `mean`, `median`, `sd` are all R functions. It is very important to understand the purpose and syntax of R functions and knowing how to create or use them.

To declare a user-defined function in R, we use the keyword `function`.

```
1 function_name ← function(parameter1, parameter2) {  
2   # function body  
3   c = parameter1 + parameter2  
4   # return the outputs  
5   return(c)  
6 }
```

Here we declared a function with name `function_name`, the function takes inputs `parameter1`, `parameter2` and returns an output `c`. It can take any number of inputs but **only one** outputs.

# Example

Here we want to write a function in R that calculates the sample mean and sample standard deviation

```
1 my_summary ← function(X) {  
2   # Write operations within the curly brackets  
3   m = sum(X)/length(X)  
4   s = sqrt(sum((X - m)^2)/(length(X) - 1))  
5   # put mean and sd in a vector, then return the vector as a single output  
6   return(c(m, s))  
7 }
```

Then we can reuse all the operations defined in the function.

```
1 w = c(1, 2, 3, 4, 10, 30, 60, 120, 180, 300) # a data vector  
2 my_summary(w) # our function
```

```
[1] 71.0000 100.5651
```

```
1 c(mean(w), sd(w)) # R built-in function
```

```
[1] 71.0000 100.5651
```

# Summary

## Centre

- Sample mean
- Sample median
- Robustness and comparisons

## Spread

- Standard deviation
- Interquartile range

## Write functions in R