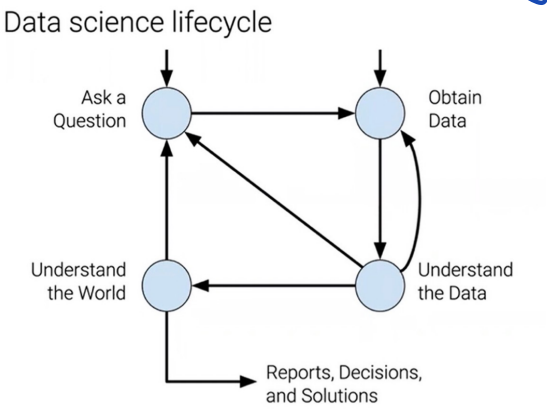
Lecture 2 Exploratory Data Analysis

§1 数据科学周期 Data science lifecycle



例：体育分析

① Sports analytics --- ask a question

1. How to combine players in a team such that the teamwork is the most effective?

2. When and where should a player shoot a ball?

3. How to defend the opposing team?

② Sports analytics --- obtain data

a camera system that collects data 25 times per second. Its aim is to follow the ball and all players on court. SportVU provides statistics such as real-time player and ball positioning through software and statistical algorithms.

STATS SportVU System

Currently, STATS is the Official Tracking partner of the NBA.

③ Sports analytics --- understand the data

- Things casual fans know: passes, shots, rebounds;
- Things casual players know: post-ups, pick-and-rolls, isolations;
- Things only professionals know: down screens, wide pins

④ Sports analytics --- understand the world

Effective field goal percentage:
[(All Field Goals Made) + 0.5(3P Field Goals Made)]/(All Field Goal Attempts)*

⑤ Sports analytics --- decisions and solutions

- Should the manager pay millions to hire a player?
- What combination of players would a team prefer?
- Is it better to pay one good shooter who takes good shots or a few good players who take okay shots but overall cost less?

§2 探索性数据分析 Exploratory Data Analysis (EDA)

1. 探索性数据分析 (exploratory data analysis)

1° 也被称为图形数据分析 (graphical data analysis)

2° EDA是一个迭代循环.

(1) 产生与数据有关的问题.

(2) 通过 visualizing, transforming and modelling your data 寻找答案.

(3) 重新完善问题/产生新问题

2. 目标

理解你的数据

3. 达成途径

提出问题

- Your goal during EDA is to develop an understanding of your data. The easiest way to do this is to use questions as tools to guide your investigation. When you ask a question, the question focuses your attention on a specific part of your dataset and helps you decide which graphs, models, or transformations to make.

两类重要问题：

1° 变量发生了什么样的变化（variation）？

2° 变量间发生了什么样的协变（covariation）？

- There is no rule about which questions you should ask to guide your research. However, two types of questions will always be useful for making discoveries within your data. You can loosely word these questions as:
  - What type of variation occurs within my variables?
  - What type of covariation occurs between my variables?

4. 一些术语（terminologies）

1° 变量（variable）：一个可测量的数量，质量或性质。

2° 值（value）：描述一个 variable 在测量时的状态。可能会随着测量变化。

3° 观测值（observation）：一组相似条件下的测量。（通常在同一时间对同一对象的观测会测量所有值）。observation 会包含多个 values，每个 value 对应一个 variable。observation 也被称为数据点（data point）

4° 表格数据（tubular data）：一组 value，每个 value 由一个 variable 和一个 observation 确定。Tubular data 是 tidy 的若每个 value 都位于自己的单元格（cell）中，每个 variable 位于自己的列（column）中，每个 observation 都位于自己的行（row）中。

- A **variable** is a quantity, quality, or property that you can measure.
- A **value** is the state of a variable when you measure it. The value of a variable may change from measurement to measurement.
- An **observation** is a set of measurements made under similar conditions (you usually make all of the measurements in an observation at the same time and on the same object). An observation will contain several values, each associated with a different variable. I'll sometimes refer to an observation as a data point.
- **Tabular data** is a set of values, each associated with a variable and an observation. Tabular data is *tidy* if each value is placed in its own "cell", each variable in its own column, and each observation in its own row.

5. 探索性数据分析：数据整洁 Tidy / clean data

- You can represent the same underlying data in multiple ways. The next example shows the same data organized in four different ways. Each dataset shows the same values of four variables: country, year, population, and cases, but each dataset organizes the values in a different way.

Table1

```
#> # A tibble: 6 x 4
#>   country     year  cases population
#>   <chr>      <int>  <int>     <int>
#> 1 Afghanistan  1999    745   19987071
#> 2 Afghanistan  2000   2666   20595360
#> 3 Brazil       1999  37737  172006362
#> 4 Brazil       2000  80488  174504898
#> 5 China        1999 212258 1272915272
#> 6 China        2000 213766 1280428583
```

Table2

```
#> # A tibble: 12 x 4
#>   country     year type       count
#>   <chr>      <int> <chr>      <int>
#> 1 Afghanistan  1999 cases        745
#> 2 Afghanistan  1999 population 19987071
#> 3 Afghanistan  2000 cases       2666
#> 4 Afghanistan  2000 population 20595360
#> 5 Brazil       1999 cases       37737
#> 6 Brazil       1999 population 172006362
#> # ... with 6 more rows
```

Table3

```
#> # A tibble: 6 x 3
#>   country     year rate
#> * <chr>      <int> <chr>
#> 1 Afghanistan  1999 745/19987071
#> 2 Afghanistan  2000 2666/20595360
#> 3 Brazil       1999 37737/172006362
#> 4 Brazil       2000 80488/174504898
#> 5 China        1999 212258/1272915272
#> 6 China        2000 213766/1280428583
```

Table 4a and 4b

table4a # cases

```
#> # A tibble: 3 x 3
#>   country     `1999` `2000`
#> * <chr>        <int>  <int>
#> 1 Afghanistan    745   2666
#> 2 Brazil       37737  80488
#> 3 China       212258 213766
```

table4b # population

```
#> # A tibble: 3 x 3
#>   country         `1999`     `2000`
#> * <chr>            <int>      <int>
#> 1 Afghanistan   19987071   20595360
#> 2 Brazil       172006362  174504898
#> 3 China       1272915272 1280428583
```

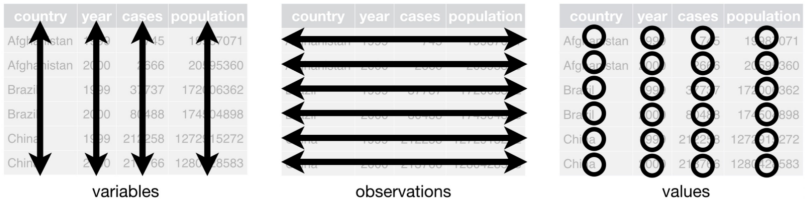
1. 数据整洁的条件

(1) 每个 value 都位于自己的单元格 (cell) 中

(2) 每个 variable 都位于自己的列 (column) 中

(3) 每个 observation 都位于自己的行 (row) 中

- In this example, only table1 is tidy. It's the only representation where each column is a variable.



2. 数据整洁的必要性

(1) general advantage: 有一个 consistent data structure. 更容易学习使用这一数据结构的工具，因为它们有潜在一致性.

(2) specific advantage: 允许 easier programming.

* Untidy data

- The principles of tidy data seem so obvious that you might wonder if you'll ever encounter a dataset that isn't tidy. Unfortunately, however, most data that you will encounter will be untidy. There are two main reasons:
  - Most people aren't familiar with the principles of tidy data, and it's hard to derive them yourself unless you spend a *lot* of time working with data.
  - Data is often organized to facilitate some use other than analysis. For example, data is often organized to make entry as easy as possible.

3. 使数据整洁

1° 步骤一：找出 variables 和 observations

2° 步骤二：解决两个常见问题

(1) 一个 variable 出现在多个列中

(2) 一个 observation 出现在多个行中

4. 常见问题

(1) 一些列的名称不是 variable，而是 value

Common problem 1

- A common problem is a dataset where some of the column names are not names of variables, but *values* of a variable.
- Take table4a: the column names 1999 and 2000 represent values of the year variable, the values in the 1999 and 2000 columns represent values of the cases variable, and each row represents two observations, not one.

- table4a # cases
- #> # A tibble: 3 x 3
- #> country `1999` `2000`
- #> * <chr> <int> <int>
- #> 1 Afghanistan 745 2666
- #> 2 Brazil 37737 80488
- #> 3 China 212258 213766

Tidying it

- To tidy a dataset like this, we need to **pivot** the offending columns into a new pair of variables. To describe that operation we need three parameters:
  - The set of columns whose names are values, not variables. In this example, those are the columns 1999 and 2000.
  - The name of the variable to move the column names to. Here it is year.
  - The name of the variable to move the column values to. Here it's cases.

- table4a # cases
- #> # A tibble: 3 x 3
- #> country `1999` `2000`
- #> * <chr> <int> <int>
- #> 1 Afghanistan 745 2666
- #> 2 Brazil 37737 80488
- #> 3 China 212258 213766

Tidying it

- table4a %>%
- pivot_longer(c(`1999`, `2000`), names_to = "year", values_to = "cases")
- #> # A tibble: 6 x 3
- #> country year cases
- #> <chr> <chr> <int>
- #> 1 Afghanistan 1999 745
- #> 2 Afghanistan 2000 2666
- #> 3 Brazil 1999 37737
- #> 4 Brazil 2000 80488
- #> 5 China 1999 212258
- #> 6 China 2000 213766

Common problem 2

- We can use pivot_longer() to tidy table4b in a similar fashion. The only difference is the variable stored in the cell values:

```
table4b %>%
  pivot_longer(c(`1999`, `2000`), names_to =
"year", values_to = "population")
#> # A tibble: 6 x 3
#>  country  year  population
#>  <chr>  <chr>  <int>
#> 1 Afghanistan 1999  19987071
#> 2 Afghanistan 2000  20595360
#> 3 Brazil  1999  172006362
#> 4 Brazil  2000  174504898
#> 5 China  1999  1272915272
#> 6 China  2000  1280428583
```

(2) 一个 observation 分布在多个行中.

Common problem 3

- pivot_wider() is the opposite of pivot_longer(). You use it when an observation is scattered across multiple rows.
- For example, take table2: an observation is a country in a year, but each observation is spread across two rows.

```
table2
#> # A tibble: 12 x 4
#>  country  year type  count
#>  <chr>  <int> <chr>  <int>
#> 1 Afghanistan 1999 cases  745
#> 2 Afghanistan 1999 population 19987071
#> 3 Afghanistan 2000 cases  2666
#> 4 Afghanistan 2000 population 20595360
#> 5 Brazil  1999 cases  37737
#> 6 Brazil  1999 population 172006362
#> # ... with 6 more rows
```

Tidying it

- To tidy this up, we only need two parameters:
  - The column to take variable names from. Here, it's type.
  - The column to take values from. Here it's count.
- Once we've figured that out, we can use pivot_wider().

```
table2
#> # A tibble: 12 x 4
#>   country     year type         count
#>   <chr>      <int> <chr>        <int>
#> 1 Afghanistan  1999 cases          745
#> 2 Afghanistan  1999 population  19987071
#> 3 Afghanistan  2000 cases         2666
#> 4 Afghanistan  2000 population  20595360
#> 5 Brazil       1999 cases         37737
#> 6 Brazil       1999 population 172006362
#> # ... with 6 more rows
```

Tidying it

```
table2 %>%
  pivot_wider(names_from = type, values_from = count)
#> # A tibble: 6 x 4
#>   country     year  cases population
#>   <chr>      <int>  <int>      <int>
#> 1 Afghanistan  1999   745   19987071
#> 2 Afghanistan  2000  2666   20595360
#> 3 Brazil       1999  37737  172006362
#> 4 Brazil       2000  80488  174504898
#> 5 China        1999 212258 1272915272
#> 6 China        2000 213766 1280428583
```

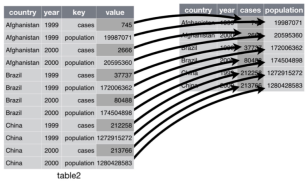


table2

Common problem 4

一列内包含了多个variables.

- table3 has a different problem: we have one column (rate) that contains two variables (cases and population). To fix this problem, we'll need the separate() function. You'll also learn about the complement of separate(): unite(), which you use if a single variable is spread across multiple columns.

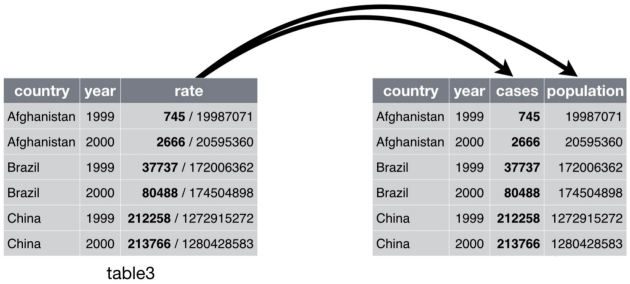
```
table3
#> # A tibble: 6 x 3
#>   country     year rate
#> *   <chr>      <int> <chr>
#> 1 Afghanistan  1999 745/19987071
#> 2 Afghanistan  2000 2666/20595360
#> 3 Brazil       1999 37737/172006362
#> 4 Brazil       2000 80488/174504898
#> 5 China        1999 212258/1272915272
#> 6 China        2000 213766/1280428583
```

Tidying it

- separate() pulls apart one column into multiple columns, by splitting wherever a separator character appears.
- The rate column contains both cases and population variabl es, and we need to split it into two variables. separate() takes the name of the column to separate, and the names of the columns to separate into.

```
table3 %>%
  separate(rate, into = c("cases", "population"))
#> # A tibble: 6 x 4
#>   country     year cases  population
#>   <chr>      <int> <chr>  <chr>
#> 1 Afghanistan  1999 745    19987071
#> 2 Afghanistan  2000 2666   20595360
#> 3 Brazil       1999 37737  172006362
#> 4 Brazil       2000 80488  174504898
#> 5 China        1999 212258 1272915272
#> 6 China        2000 213766 1280428583
```

Tidying it



country	year	rate
Afghanistan	1999	**745** / 19987071
Afghanistan	2000	**2666** / 20595360
Brazil	1999	**37737** / 172006362
Brazil	2000	**80488** / 174504898
China	1999	**212258** / 1272915272
China	2000	**213766** / 1280428583

country	year	cases	population
Afghanistan	1999	**745**	19987071
Afghanistan	2000	**2666**	20595360
Brazil	1999	**37737**	172006362
Brazil	2000	**80488**	174504898
China	1999	**212258**	1272915272
China	2000	**213766**	1280428583

table3

Tidying it

- Look carefully at the column types: you'll notice that cases and population are character columns. This is the default behavior in separate(): it leaves the type of the column as is. Here, however, it's not very useful as those really are numbers. We can ask separate() to try and convert to better types using convert = TRUE:

```
table3 %>%
  separate(rate, into = c("cases", "population"))
#> # A tibble: 6 x 4
#>   country     year cases  population
#>   <chr>      <int> <chr>  <chr>
#> 1 Afghanistan  1999 745    19987071
#> 2 Afghanistan  2000 2666   20595360
#> 3 Brazil       1999 37737  172006362
#> 4 Brazil       2000 80488  174504898
#> 5 China        1999 212258 1272915272
#> 6 China        2000 213766 1280428583
```

5. 数值缺失 missing value

1° 数值缺失的两种形式：

1) Explicitly（显式）：用 NA 标出

2) Implicitly（隐式）：没有在数据中出现

- There are two missing values in this dataset:
  - The return for the fourth quarter of 2015 is explicitly missing, because the cell where its value should be instead contains NA.
  - The return for the first quarter of 2016 is implicitly missing, because it simply does not appear in the dataset.

```
stocks <- tibble(
  year   = c(2015, 2015, 2015, 2015, 2016, 2016, 2016),
  qtr    = c(   1,    2,    3,    4,    2,    3,    4),
  return = c(1.88, 0.59, 0.35,   NA, 0.92, 0.17, 2.66)
)
```

2° 改变数据表的呈现形式可使隐式缺失变为显式

Missing values

- An explicit missing value is the presence of an absence; an implicit missing value is the absence of a presence.
- The way that a dataset is represented can make implicit values explicit. For example, we can make the implicit missing value explicit by putting years in the columns:

```
stocks %>%
  pivot_wider(names_from = year,
values_from = return)
#> # A tibble: 4 x 3
#>    qtr `2015` `2016`
#>  <dbl>  <dbl>  <dbl>
#> 1    1   1.88  NA
#> 2    2   0.59   0.92
#> 3    3   0.35   0.17
#> 4    4  NA      2.66
```

6. Untidy value 的原因

1° 多数人对数据整洁的原则不熟悉.

2° 数据都整理成便于使用而非便于分析的形式.

Untidy data

- The principles of tidy data seem so obvious that you might wonder if you'll ever encounter a dataset that isn't tidy. Unfortunately, however, most data that you will encounter will be untidy. There are two main reasons:
  - Most people aren't familiar with the principles of tidy data, and it's hard to derive them yourself unless you spend a *lot* of time working with data.
  - Data is often organized to facilitate some use other than analysis. For example, data is often organized to make entry as easy as possible.

7. Non-tidy value 的原因

1) 替代的数据结构有实质性的性能或空间优势.

2) 某些领域已经发展了自己的存储数据规定

- There are lots of useful and well-founded data structures that are not tidy data. There are two main reasons to use other data structures:
  - Alternative representations may have substantial performance or space advantages.
  - Specialized fields have evolved their own conventions for storing data that may be quite different to the conventions of tidy data.

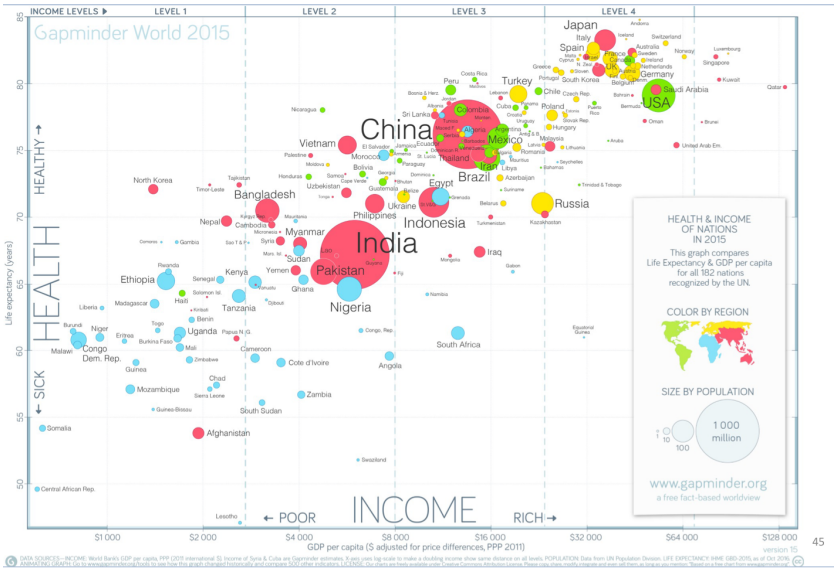
tidy data is not the only way

- Either of these reasons means you'll need something other than a tibble (or data frame). If your data does fit naturally into a rectangular structure composed of observations and variables, I think tidy data should be your default choice. But there are good reasons to use other structures; tidy data is not the only way.

§4 探索性数据分析：数据可视化 data visualization

1. 数据可视化

- Data visualization is a very important part of data analysis. You can use it to explore your data. If you understand your data well, you'll have a better chance to find some insights. Finally, when you find any insights, you can use visualizations again to be able to share your findings with other people.



Basic data visualizations

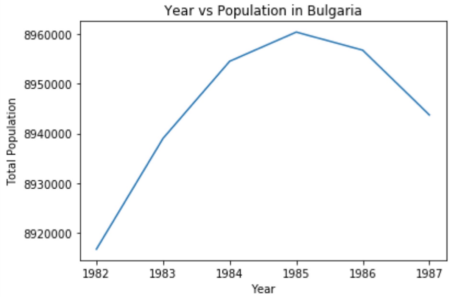
- There are many types of visualizations. Some of the most famous are: **line plot**, **scatter plot**, **histogram**, **box plot**, **bar chart**, and **pie chart**. But among so many options how do we choose the right visualization?
- There are many visualization packages in Python. One of the most famous is Matplotlib. (We don't focus on coding here!)

2. Line plot (折线图)

反映数据在时间间隔内的变化趋势 (trend). 是一个 time series.

Line Plot

- This type of plot is often used to visualize a trend in data over intervals of time - a **time series**.



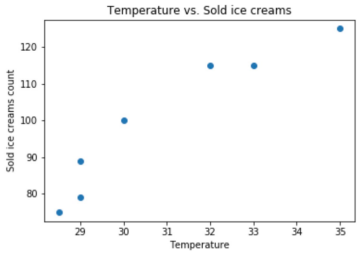
3. Scatter plot (散点图)

比较两个变量，展现趋势与相关性 (correlations)

* 图表上呈现相关性的变量A.B未必存在A影响B的关系，A.B可能同时受另一变量C影响才呈现相关性

Scatter plot

- This type of plot can be used to display trends or correlations. In data science, it shows how 2 variables compare.



48

4. Histogram (直方图)

展现数据数值的分布 (distribution of numeric data)

直方图的构建：1° 把 values 的值的范围 (range) 划分为一系列区间 (interval)

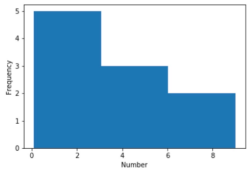
2° 统计 how many values fall into each interval.

3° intervals 也被称为 bins (区间).

Bins 是变量的连续 (consecutive) 且非重叠 (non-overlapping) 的 intervals. 它们必须相邻 (adjacent) 且通常大小相同 (of equal size)

Histogram

- Histogram is an accurate representation of the distribution of numeric data. To create a histogram, first, we divide the entire range of values into a series of intervals, and second, we count **how many values fall into each interval**. The intervals are also called **bins**. The bins are consecutive and non-overlapping intervals of a variable. They must be adjacent and are often of equal size.



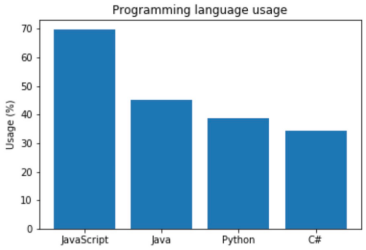
49

5. Bar chart (柱状图/条形图)

便于比较不同类别的给定数值 (given numeric value).

Bar chart

- Bar chart represents categorical data with rectangular bars. Each bar has a height corresponds to the value it represents. It's useful when we want to compare a given numeric value on different categories.



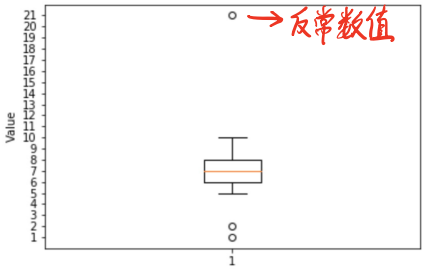
50

6. Box plot (箱形图)

从五个数值层面展示 values 的 distribution：minimum，first quartile，median，third quatile，maximum.

Box plot

- Box plot is a way to show the distribution of values based on the five-number summary: **minimum**, **first quartile**, **median**, **third quartile**, and **maximum**.



51

7. Pie chart (饼图)

展示数值的比例

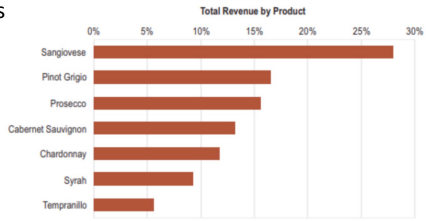
不建议使用！：1° 不利于同一饼图内不同 sections 间的比较.

2° 不利于不同饼图内 data 的比较.

3° 在很多层面可被 bar chart 替代.

Pie chart versus bar chart

- The main reason is that it's difficult to compare the sections of a given pie chart. Also, it's difficult to compare data across multiple pie charts. In many cases, they can be **replaced** by a **bar chart**.



§5 探索性数据分析：两种分析思路 (variation)

1. Variation

Variation 是 variables 的 value 随着测量而变化的趋势.(描述变量内的情况)

Variation 的产生：

1° 对同一连续变量 (continuous variables) 测量两次，得到不同结果.

2° 对恒定量 (如光速) 测量两次，误差会使结果不同.

3° 对不同时间的同一对象 / 同一时间的不同对象进行测量，同一类别变量 (categorical variables) 会变化.

2. 基于 variation 使数据可视化

可视化的方式由变量类型和目的决定

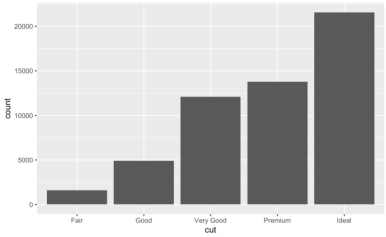
1° categorical variables 的分布

使用 bar chart.

- How you visualise the distribution of a variable will depend on whether the variable is categorical or continuous. A variable is **categorical** if it can only take one of a small set of values. To examine the distribution of a categorical variable, use a bar chart.

Bar chart

- The height of the bars displays how many observations occurred with each x value.



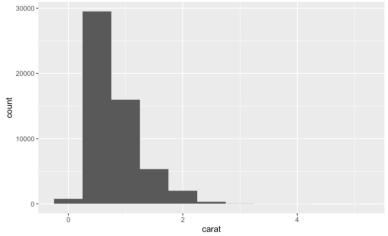
59

2° continuous variables 的分布
使用 histogram

- A variable is **continuous** if it can take any of an infinite set of ordered values. Numbers and date-times are two examples of continuous variables. To examine the distribution of a continuous variable, use a histogram.

Histogram

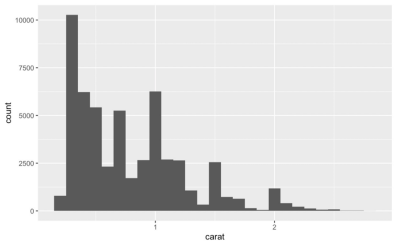
- A histogram divides the x-axis into equally spaced bins and then uses the height of a bar to display the number of observations that fall in each bin. In the graph, the tallest bar shows that almost 30,000 observations have a carat value between 0.25 and 0.75.



61

尽可能多地调整 binwidths (组距)

- You should always explore a variety of binwidths when working with histograms, as different binwidths can reveal different patterns. For example, here is how the graph above looks when we zoom into just the diamonds with a size of less than three carats and choose a smaller binwidth.

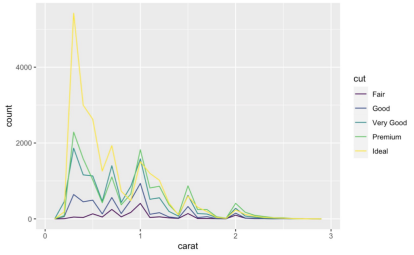


62

3° 在一个图内叠加 (overlay) 多个 histograms
使用 line plots

Line plots

- If you wish to overlay multiple histograms in the same plot, instead of displaying the counts with bars, use lines instead. It's much easier to understand overlapping lines than bars.



63

3、基于variation对可视化数据进行分析并提出问题

- Now that you can visualize variation, what should you look for in your plots? And what type of follow-up questions should you ask? I've put together a list below of the most useful types of information that you will find in your graphs, along with some follow-up questions for each type of information. The key to asking good follow-up questions will be to rely on your curiosity (What do you want to learn more about?) as well as your skepticism (How could this be misleading?).

1° typical value (特征值)

在 bar chart 和 histogram 中，

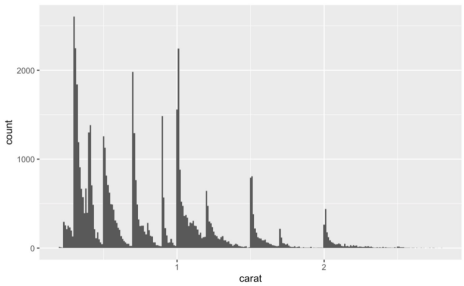
tall bars ⟶ common values of variables

short bars ⟶ less-common values of variables

Typical values

- In both bar charts and histograms, tall bars show the common values of a variable, and shorter bars show less-common values. Places that do not have bars reveal values that were not seen in your data. To turn this information into useful questions, look for anything unexpected:
  - Which values are the most common? Why?
  - Which values are rare? Why? Does that match your expectations?
  - Can you see any unusual patterns? What might explain them?

- Why are there more diamonds at whole carats and common fractions of carats?
- Why are there more diamonds slightly to the right of each peak than there are slightly to the left of each peak?
- Why are there no diamonds bigger than 3 carats?



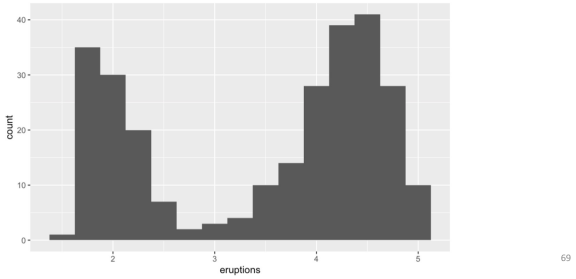
2° clusters (相似值/群集)

clusters of similar values 表明你的数据中存在 subgroups (子组)

Clusters

- Clusters of similar values suggest that subgroups exist in your data. To understand the subgroups, ask:
  - How are the observations within each cluster similar to each other?
  - How are the observations in separate clusters different from each other?
  - How can you explain or describe the clusters?
  - Why might the appearance of clusters be misleading?

- The histogram below shows the length (in minutes) of 272 eruptions of the Old Faithful Geyser in Yellowstone National Park.



- Many of the questions above will prompt you to explore a relationship *between* variables, for example, to see if the values of one variable can explain the behavior of another variable. We'll get to that shortly.

3° unusual values (异常值)

Outliers (异常值) 是不同寻常的 observation，是不符合大体模式的 pattern.

来源：

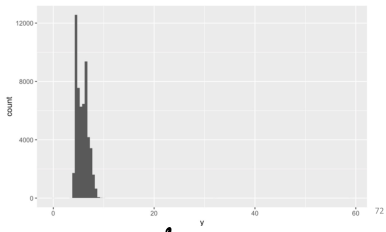
(1) data entry errors

(2) important new science.

异常值的发现：

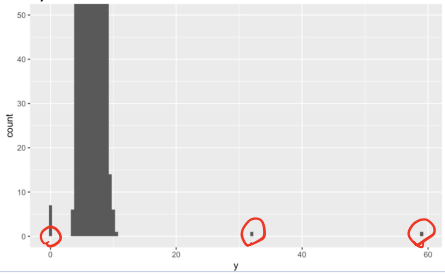
(1) 对于大量数据，很难在 histogram 中发现 outliers.

- When you have a lot of data, outliers are sometimes difficult to see in a histogram. For example, take the distribution of the y variable from the diamonds dataset. The only evidence of outliers is the unusually wide limits on the x-axis.



(2) common bins 中有太多 observation，rare bins 又太短，为便于观察，我们可 缩放 y 轴.

- There are so many observations in the common bins that the rare bins are so short that you can't see them. To make it easy to see the unusual values, we need to zoom to small values of the y-axis.



7

Unusual values

- This allows us to see that there are three unusual values: 0, ~30, and ~60. We pluck them out with dplyr:
- The y variable measures one of the three dimensions of these diamonds, in mm. We know that diamonds can't have a width of 0mm, so these values must be incorrect. We might also suspect that measurements of 32mm and 59mm are implausible: those diamonds are over an inch long, but don't cost hundreds of thousands of dollars!

```
unusual <- diamonds %>%
  filter(y < 3 | y > 20) %>%
  select(price, x, y, z) %>%
  arrange(y)
unusual
#> # A tibble: 9 x 4
#>   price     x     y     z
#>   <int> <dbl> <dbl> <dbl>
#> 1  5139  0     0     0
#> 2  6381  0     0     0
#> 3 12800  0     0     0
#> 4 15686  0     0     0
#> 5 18034  0     0     0
#> 6  2130  0     0     0
#> 7  2130  0     0     0
#> 8  2075  5.15  31.8  5.12
#> 9 12210  8.09  58.9  8.06
```

74

异常值的处理：

对有无 outlier 的情况分别重复分析.

(1) 若对结果只有 minimal effect，可用 missing value (缺失值) 替代.

替代时有两种 options.

① 去除含 strange value 的整行 (不建议!)

② 仅将 unusual value 替换为 missing value.

(2) 若对结果有 substantial effect，则需找出导致它们的原因.

- It's good practice to repeat your analysis with and without the outliers. If they have minimal effect on the results, and you can't figure out why they're there, it's reasonable to replace them with missing values, and move on. However, if they have a substantial effect on your results, you shouldn't drop them without justification. You'll need to figure out what caused them (e.g. a data entry error) and disclose that you removed them in your write-up.

Missing values

- If you've encountered unusual values in your dataset, and simply want to move on to the rest of your analysis, you have two options.
  - Drop the entire row with the strange values: (I don't recommend this option because just because one measurement is invalid, doesn't mean all the measurements are. Additionally, if you have low quality data, by time that you've applied this approach to every variable you might find that you don't have any data left!)
  - Instead, I recommend replacing the unusual values with missing values.

§6 探索性数据分析：两种分析思路（covariation）

1. covariation

covariation是两或多个 variables 以相关模式发生变化的趋势，(描述变量间的情况)

发现 covariation 的最佳方式：对多个数据可视化

2. 基于variation 使数据可视化与分析

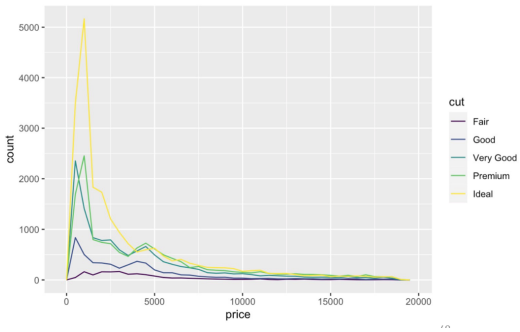
可视化的方式由变量类型和目的决定

1° 一个 categorical variable 和一个 continuous variable

(1) 使用 line plot (通常横轴为 continuous variable)

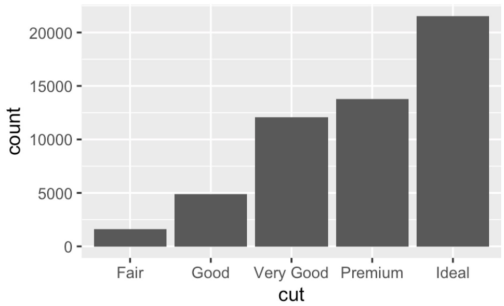
A categorical and continuous variable

- It's common to want to explore the distribution of a continuous variable broken down by a categorical variable.
- For example, let's explore how the price of a diamond varies with its quality:



但不同种类的钻数也存在差异

- It's hard to see the difference in distribution because the overall counts differ so much:

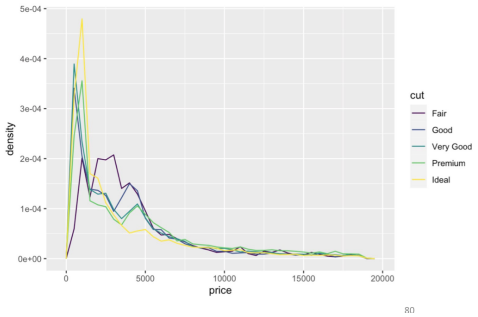


可将 y 轴由 count 替换为 density

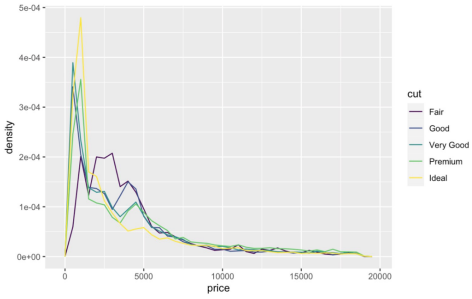
这是 count standard

每个 frequency polygon (频数多边形) 以下的面积为 1.

- To make the comparison easier we need to swap what is displayed on the y-axis. Instead of displaying count, we'll display **density**, which is the count standardised so that the area under each frequency polygon is one.



- There's something rather surprising about this plot it appears that fair diamonds (the lowest quality) have the highest average price!



(2) 使用 box plot (通常纵轴为 continuous variable)

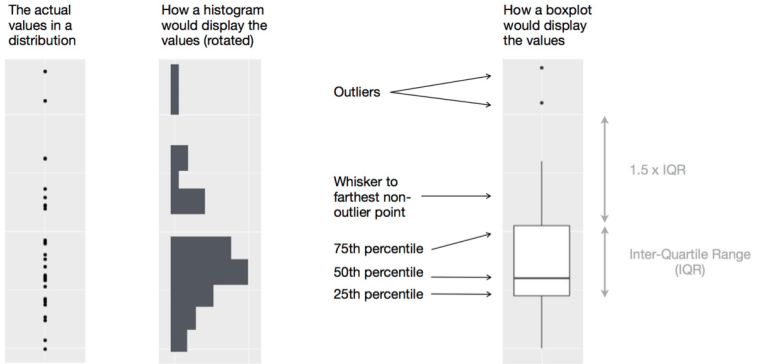
① IQR:
从 distribution 的 25th percentile 到 75th persentile 是一个 box,
这段距离被称为 interquartile range (IQR) (四分位距)

② median (中位数)
由 box 中间的直线表示.

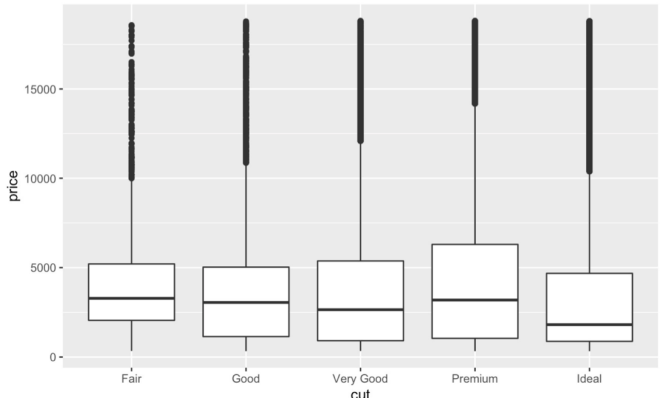
③ outliner points
超出 box 边缘 1.5 倍 IQR 的 visual points

Box plot

- Another alternative to display the distribution of a continuous variable broken down by a categorical variable is the boxplot. A **boxplot** is a type of visual shorthand for a distribution of values that is popular among statisticians. Each boxplot consists of:
  - A box that stretches from the 25th percentile of the distribution to the 75th percentile, a distance known as the interquartile range (IQR). In the middle of the box is a line that displays the median, i.e. 50th percentile, of the distribution. These three lines give you a sense of the spread of the distribution and whether or not the distribution is symmetric about the median or skewed to one side.
  - Visual points that display observations that fall more than 1.5 times the IQR from either edge of the box. These outlying points are unusual so are plotted individually.
  - A line (or whisker) that extends from each end of the box and goes to the farthest non-outlier point in the distribution.



- Let's take a look at the distribution of price by cut.

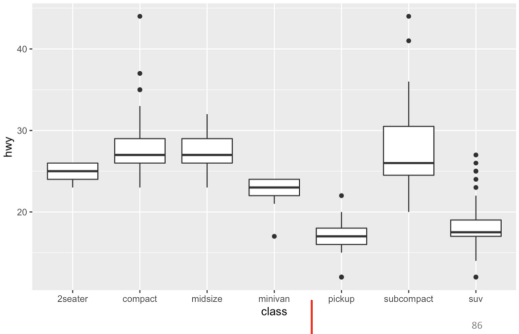


关于分布的信息更少，但 box plot 更紧凑（compact）

- We see much less information about the distribution, but the boxplots are much more compact so we can more easily compare them (and fit more on one plot). It supports the counterintuitive finding that better quality diamonds are cheaper on average!

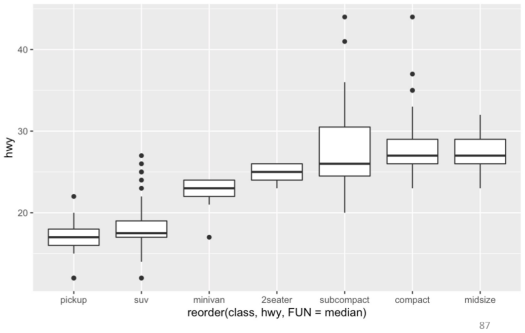
在这个例子中，切工（cut）是一个 ordered factor.
很多 categorical variable 没有这种 intrinsic order（内在顺序），
我们可以 reorder 它们使信息呈现更高效.

- Cut is an ordered factor: fair is worse than good, which is worse than very good and so on.
- Many categorical variables don't have such an intrinsic order, so you might want to reorder them to make a more informative display.



86

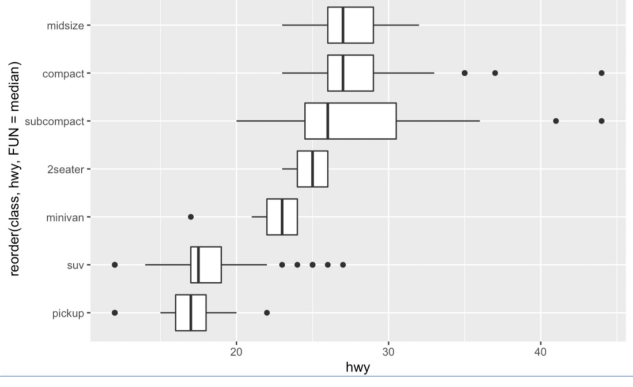
- Cut is an ordered factor: fair is worse than good, which is worse than very good and so on.
- Many categorical variables don't have such an intrinsic order, so you might want to reorder them to make a more informative display.



87

若 variable names 太长，可把图表旋转 90°

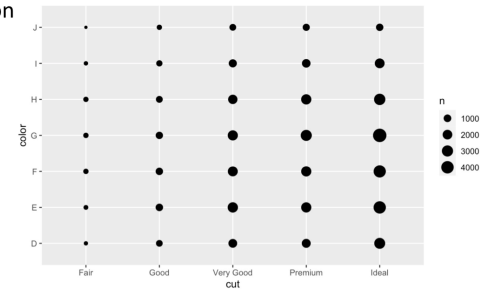
- If you have long variable names, you can also flip it 90°.



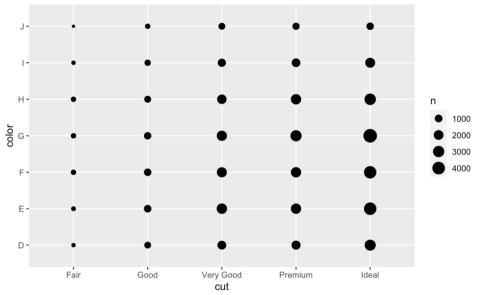
2° 两个 categorical variables
需要计算每组变量组合的 count.

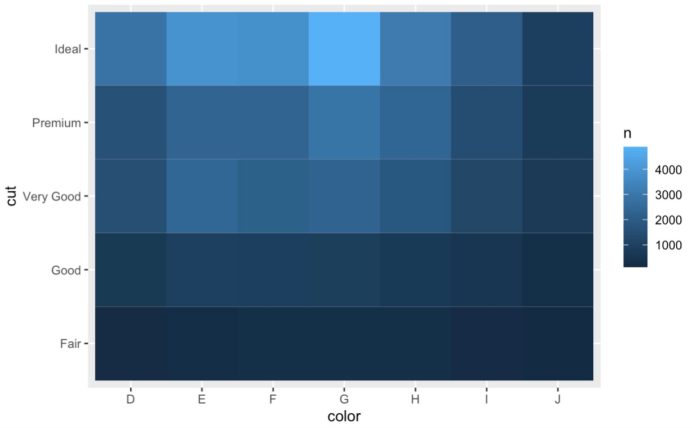
Two categorical variables

- To visualise the covariation between categorical variables, you'll need to count the number of observations for each combination.



- The size of each circle in the plot displays how many observations occurred at each combination of values. Covariation will appear as a strong correlation between specific x values and specific y values.





3° 两个 continuous variables

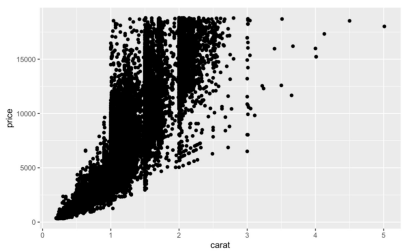
1) 使用 scatterplot

① scatterplot 的实用性会随着数据库的增加而减弱 (点开始 overplot (过度重叠))

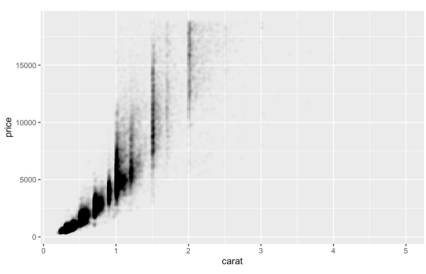
此时可以 add transparency (增加透明度)

Two continuous variables

- One great way to visualise the covariation between two continuous variables is to draw a scatterplot. You can see covariation as a pattern in the points.
- For example, you can see the relationship between the carat size and price of a diamond.

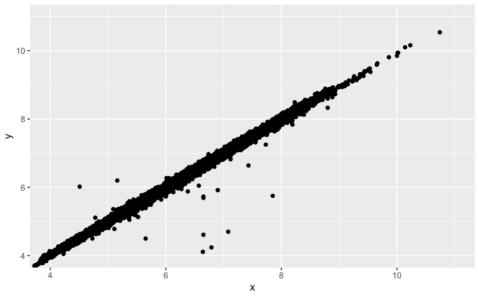


- Scatterplots become less useful as the size of your dataset grows, because points begin to overplot, and pile up into areas of uniform black (as above). One way to fix the problem is by adding transparency.



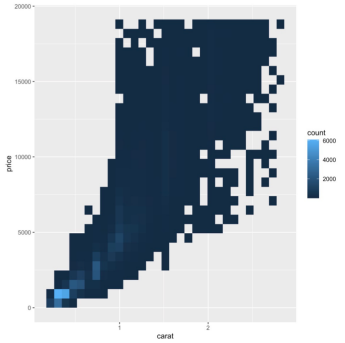
② 二维图能展现出一维图中无法发现的 outliers (一些不寻常的数值组合)

- Two dimensional plots reveal outliers that are not visible in one dimensional plots.
- For example, some points in the plot below have an unusual combination of x and y values, which make the points outliers even though their x and y values appear normal when examined separately.



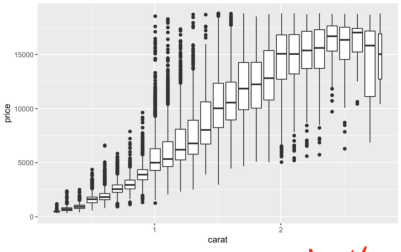
(2) 使用 bin
用颜色深浅反映落入每个 bin 中的 points 数.

- Another solution is to use bin: divide the coordinate plane into 2d bins and then use a fill color to display how many points fall into each bin.



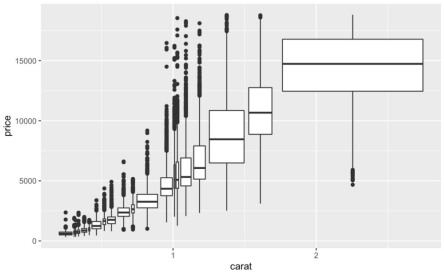
(3) 对其中一个 variable 进行 bin (分为不同区间), 使之可被认作 categorical variable.

- Another option is to bin one continuous variable so it acts like a categorical variable. Then you can use one of the techniques for visualising the combination of a categorical and a continuous variable that you learned about.
- For example, you could bin carat and then for each group, display a boxplot:



默认条件下, 无法区分不同区间中落入的点的个数, 因此可改变 box 的 width 使每个 box 中的点的个数相同.

- By default, boxplots look roughly the same (apart from number of outliers) regardless of how many observations there are, so it's difficult to tell that each boxplot summarises a different number of points.
- One way to show that is to make the width of the boxplot proportional to the number of points.



3. Pattern and model

1° Pattern (模式)

(1) 若两个变量间存在 systematic relationship, 它将在数据中以 pattern 的形式出现

(2) pattern 可以揭示 covariation, 并利用 covariation 减少 uncertainty.
若两变量 covary (共变), 则可利用一个变量的 value 预测另一个.
若协变是由 causal relationship (因果关系) 导致的, 则可以利用一个变量来控制另

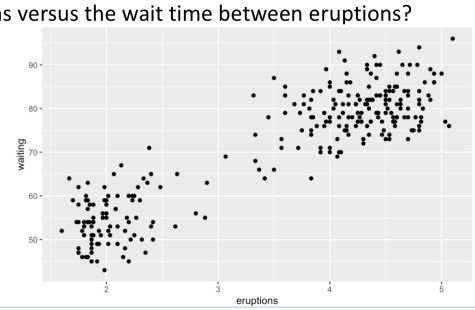
一个.

Patterns and models

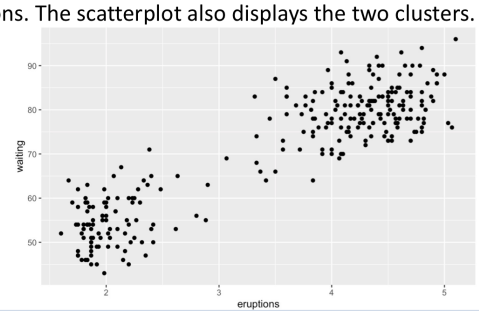
- Patterns in your data provide clues about relationships. If a systematic relationship exists between two variables it will appear as a pattern in the data. If you spot a pattern, ask yourself:
  - Could this pattern be due to coincidence (i.e. random chance)?
  - How can you describe the relationship implied by the pattern?
  - How strong is the relationship implied by the pattern?
  - What other variables might affect the relationship?
  - Does the relationship change if you look at individual subgroups of the data?

- Patterns provide one of the most useful tools for data scientists because they reveal covariation. If you think of variation as a phenomenon that creates uncertainty, covariation is a phenomenon that reduces it. If two variables covary, you can use the values of one variable to make better predictions about the values of the second. If the covariation is due to a causal relationship (a special case), then you can use the value of one variable to control the value of the second.

- What can you tell from this scatterplot of Old Faithful eruption lengths versus the wait time between eruptions?



- It shows a pattern: longer wait times are associated with longer eruptions. The scatterplot also displays the two clusters.

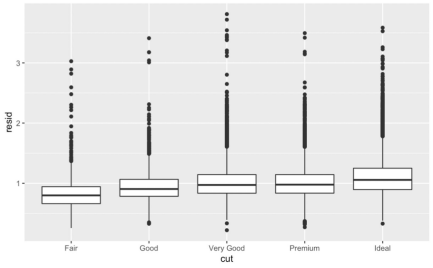


2° Model (模型)

Models 是从数据中 extract (提取) patterns 的工具.

- Models are a tool for extracting patterns out of data. For example, consider the diamonds data. It's hard to understand the relationship between cut and price, because cut and carat, and carat and price are tightly related. It's possible to use a model to remove the very strong relationship between price and carat so we can explore the subtleties that remain.

- Once you've removed the strong relationship between carat and price, you can see what you expect in the relationship between cut and price: relative to their size, better quality diamonds are more expensive.



- Can you now explain this plot? Why better quality diamonds seems cheaper?