Machine Learning

|A Quick Dive into Dataset Structures

Prof. Dr. Mostafa Elhosseini

https://youtube.com/drmelhosseini

Agenda

- Get The Data

Get The Data

- The full Jupyter notebook is available at https://github.com/ageron/handson-ml3
- Download the data
- In typical environments your data would be available in a relational database (or some other common datastore) and spread across multiple tables/documents/files
- In this project, you will just download a single compressed file, housing.tgz, which contains a comma-separated value (CSV) file called **housing.csv** with all the data
- https://github.com/ageron/data/raw/main/housing.tgz
- It is important now to download and install Pandas library

A Quick Dive

```
housing.head()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity
0	-122.23	37.88	41.0	880.0	129.0	322.0	126.0	8.3252	452600.0	NEAR BAY
1	-122.22	37.86	21.0	7099.0	1106.0	2401.0	1138.0	8.3014	358500.0	NEAR BAY
2	-122.24	37.85	52.0	1467.0	190.0	496.0	177.0	7.2574	352100.0	NEAR BAY
3	-122.25	37.85	52.0	1274.0	235.0	558.0	219.0	5.6431	341300.0	NEAR BAY
4	-122.25	37.85	52.0	1627.0	280.0	565.0	259.0	3.8462	342200.0	NEAR BAY

A Quick Dive

```
[5]  # prompt: Using dataframe housing: get size

     housing.shape
```

```
(20640, 10)
```

- There are 20,640 instances in the dataset
- fairly small by Machine Learning standards
- There are 10 attributes
- Each row represents one district

A Quick Dive

- The **info()** method is useful to get a quick description of the data

```
housing.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

- There are 20,640 instances in the dataset
- total_bedrooms attribute has only 20,433 nonnull values
- 207 districts are missing this feature
- All attributes are numerical, except the ocean_proximity field

A Quick Dive

- You can find out what categories exist and how many districts belong to each category by using the **value_counts()** method

```
housing["ocean_proximity"].value_counts()
```

```
<1H OCEAN     9136
INLAND        6551
NEAR OCEAN    2658
NEAR BAY      2290
ISLAND           5
Name: ocean_proximity, dtype: int64
```

- you probably noticed that the values in that column were repetitive, which means that it is probably a **categorical attribute**

A Quick Dive

- The **describe()** method shows a summary of the numerical attributes

```
housing.describe()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
count	20640.000000	20640.000000	20640.000000	20640.000000	20433.000000	20640.000000	20640.000000	20640.000000	20640.000000
mean	-119.569704	35.631861	28.639486	2635.763081	537.870553	1425.476744	499.539680	3.870671	206855.816909
std	2.003532	2.135952	12.585558	2181.615252	421.385070	1132.462122	382.329753	1.899822	115395.615874
min	-124.350000	32.540000	1.000000	2.000000	1.000000	3.000000	1.000000	0.499900	14999.000000
25%	-121.800000	33.930000	18.000000	1447.750000	296.000000	787.000000	280.000000	2.563400	119600.000000
50%	-118.490000	34.260000	29.000000	2127.000000	435.000000	1166.000000	409.000000	3.534800	179700.000000
75%	-118.010000	37.710000	37.000000	3148.000000	647.000000	1725.000000	605.000000	4.743250	264725.000000
max	-114.310000	41.950000	52.000000	39320.000000	6445.000000	35682.000000	6082.000000	15.000100	500001.000000

A Quick Dive

- The **std** row shows the standard deviation, which measures **how dispersed** the values are.
- The 25%, 50%, and 75% rows show the corresponding percentiles:
- A percentile indicates the value below which a given percentage of observations in a group of observations falls.
- For example, 25% of the districts have a housing_median_age lower than 18, while 50% are lower than 29 and 75% are lower than 37.
- These are often called the 25th percentile (or 1st quartile), the median, and the 75th percentile (or 3rd quartile).

A Quick Dive

- **Looking at a graph** often gives you more of a feel for a variable and its distribution than looking at the raw data
- A **histogram** is a graph that uses bars to portray the **frequencies or the relative frequencies** of the possible outcomes for a quantitative variable

A Quick Dive

- A **histogram** provides a versatile way to picture the distribution of a large data set.
  - Divide the range of the data into intervals (bins) of equal width.
  - Count the number of observations in each interval, creating a frequency table.
  - On the horizontal axis, label the values or the endpoints of the intervals.
  - Draw a bar over each value or interval with a height equal to its frequency (or percentage), values of which are marked on the vertical axis.
  - Label and title appropriately.

A Quick Dive

- A **histogram** provides a versatile way to picture the distribution of a large data set.
  - Divide the range of the data into intervals (bins) of equal width.
  - Count the number of observations in each interval, creating a frequency table.
  - On the horizontal axis, label the values or the endpoints of the intervals.
  - Draw a bar over each value or interval with a height equal to its frequency (or percentage), values of which are marked on the vertical axis.
  - Label and title appropriately.

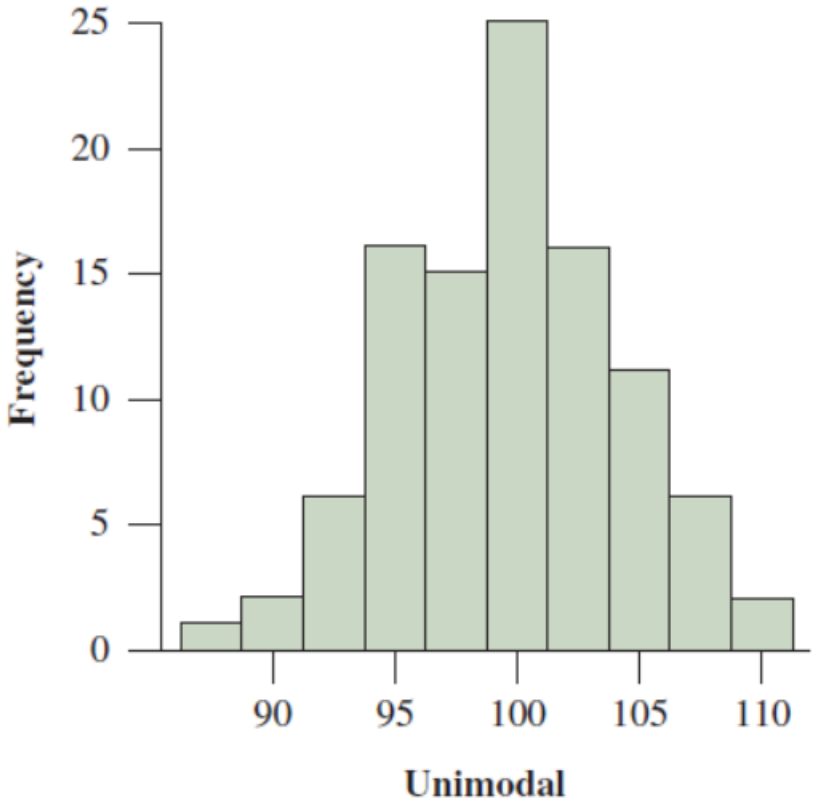
Interpreting Histograms

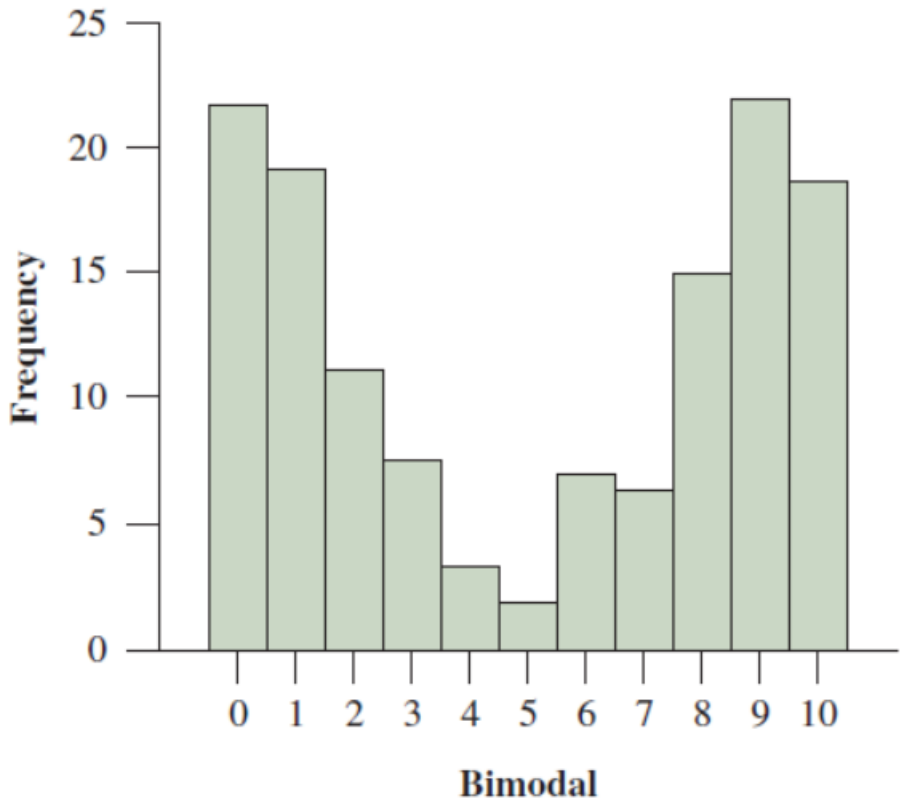
- Overall pattern consists of the **center**, **spread**, and **shape**.
  - Assess where distribution is centered by finding the **median** (50% of data below median and 50% of data above).
  - Assess the spread of a distribution.
  - Shape of a distribution: roughly symmetric, skewed to the right, or skewed to the left

Shape

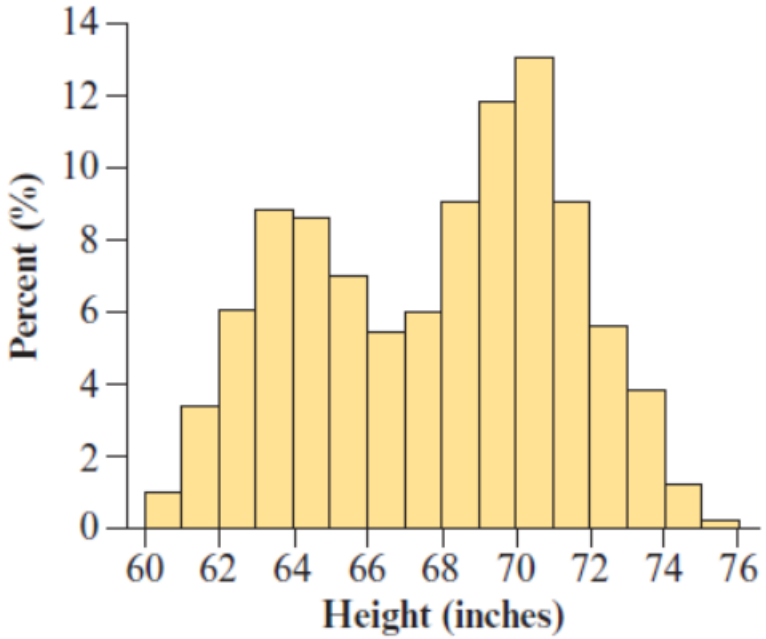
- Does the distribution have a single mound or peak?
  - If so, the distribution is called **unimodal**, and the value that occurs most often is called the **mode**.
  - A distribution with **two distinct mounds** is called **bimodal**
  - A bimodal distribution can result, for example, when a population is **polarized** on a **controversial** issue
  - A bimodal distribution can also result when the observations come from two different groups
  - histogram of the height of students in your school might show two peaks, one for females and one for males

Shape


Frequency
25
20
15
10
5
0
90
95
100
105
110
Unimodal


Frequency
25
20
15
10
5
0
0
1
2
3
4
5
6
7
8
9
10
Bimodal

Shape

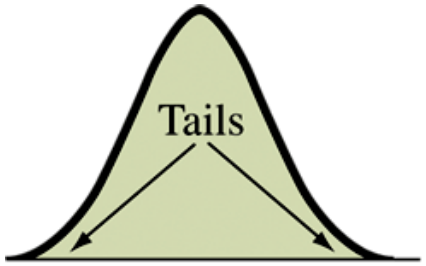


Shape

- What is the shape of the distribution? The shape of a **unimodal** distribution is often described as **symmetric**, or **skewed**
- A distribution is **symmetric** if the side of the distribution below a central value is a mirror image of the side above that central value.
- The distribution is **skewed** if one side of the distribution stretches out longer than the other side
- Do the data cluster together, or is there a gap such that one or more observations noticeably deviate from the rest — **outlier**

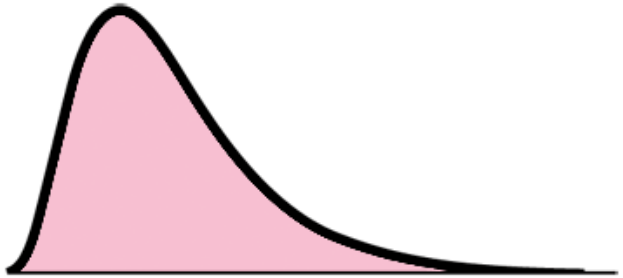
Shape

Symmetric Distributions: if both left and right sides of the histogram are mirror images of each other.



Symmetric

Skewed to the left

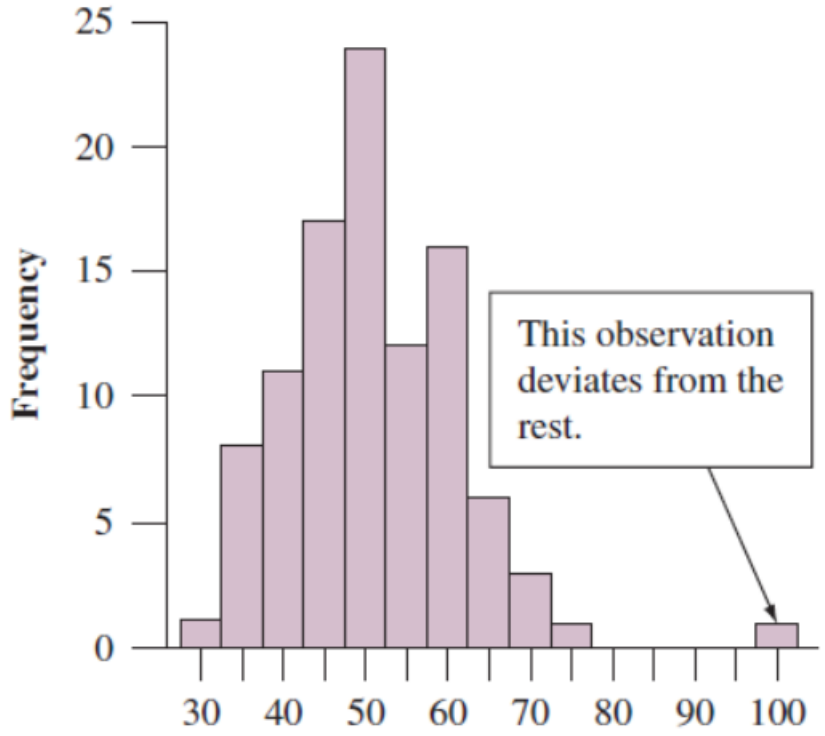


Skewed to the right

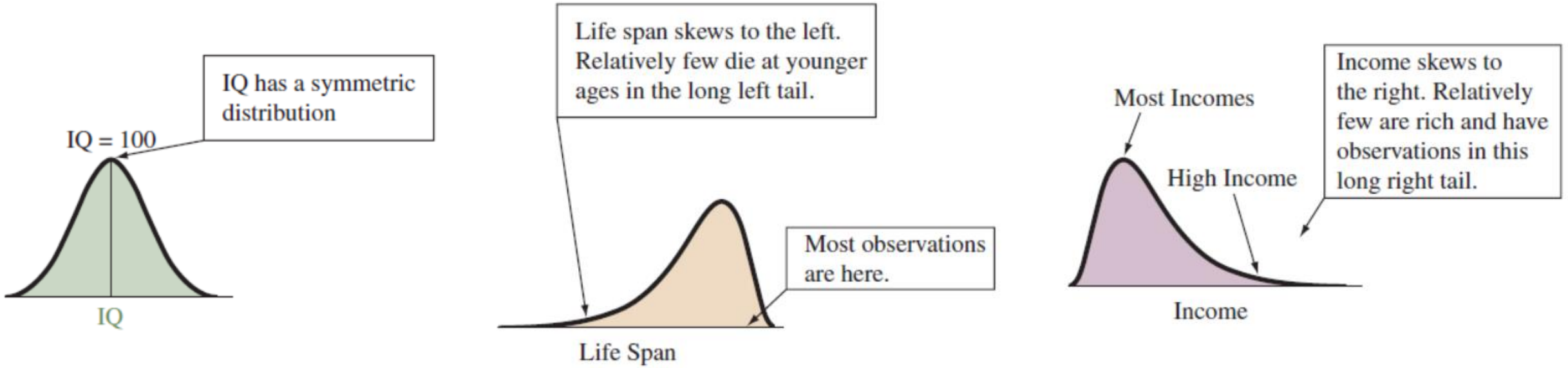
A distribution is **skewed to the left** if the left tail is longer than the right tail.

A distribution is **skewed to the right** if the right tail is longer than the left tail.

Shape


25
20
15
10
5
0
Frequency
This observation deviates from the rest.
30
40
50
60
70
80
90
100

Shape



with small data sets, the shape of the distribution might not be so obvious

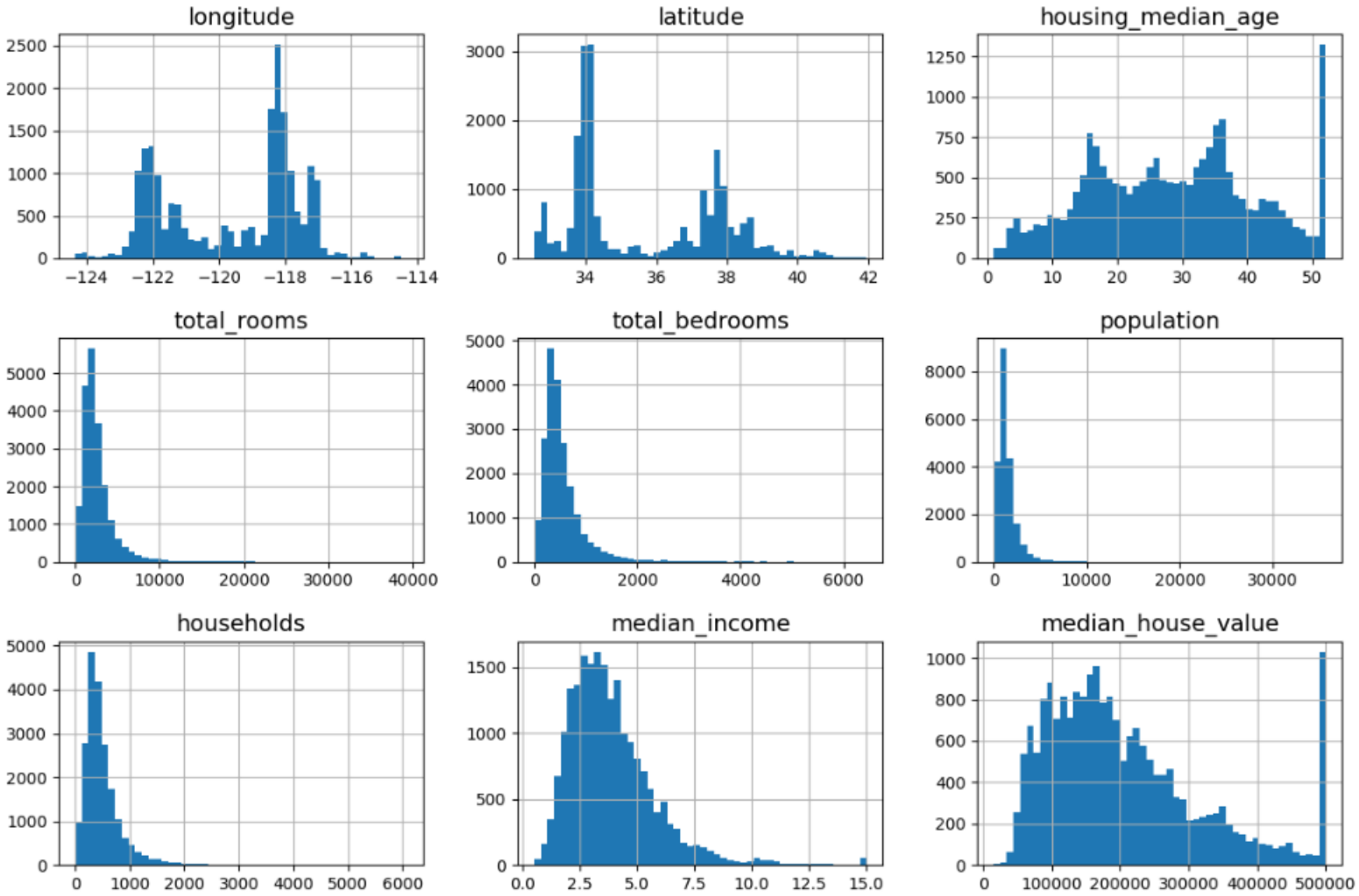
A Quick Dive

```
import matplotlib.pyplot as plt

# extra code – the next 5 lines define the default font sizes
plt.rc('font', size=14)
plt.rc('axes', labelsize=14, titlesize=14)
plt.rc('legend', fontsize=14)
plt.rc('xtick', labelsize=10)
plt.rc('ytick', labelsize=10)

housing.hist(bins=50, figsize=(12, 8))
save_fig("attribute_histogram_plots")  # extra code
plt.show()
```

A Quick Dive


longitude
latitude
housing_median_age
total_rooms
total_bedrooms
population
households
median_income
median_house_value

A Quick Dive

Looking at these histograms, you notice a few things:

- First, the median income attribute does not look like it is expressed in US dollars (USD)
- After checking with the team that collected the data, you are told that the data has been **scaled and capped** at 15 (actually, 15.0001) for higher median incomes, and at 0.5 (actually, 0.4999) for lower median incomes.
- **Why Capping**?

A Quick Dive

Data collectors sometimes cap numbers, or apply a maximum limit to certain values in a dataset, for various reasons.

- **Privacy and Anonymity**: In datasets containing sensitive information, capping can **help preserve individual privacy** by preventing the identification of specific entities or outliers that may be recognizable due to their extreme values.
- **Data Integrity**: Capping might be used to **remove outliers** that are believed to be errors or too atypical to be representative of the general population or phenomenon being studied.
- **Practical Limits**: In some cases, values are capped due to the nature of the data collection process or the practical limits of what is being measured (e.g., a survey with predefined maximum values).

A Quick Dive

Looking at these histograms, you notice a few things:

- The **housing median age** and the **median house** value were also **capped**. The latter may be a **serious problem** since it is your target attribute (your labels). Your machine learning algorithms may learn that prices never go beyond that limit
- You need to check with your client team (the team that will use your system's output) to see if this is a problem or not. If they tell you that they need precise predictions even beyond $500,000, then you have two options:

A Quick Dive

Looking at these histograms, you notice a few things:

- Collect proper labels for the districts whose labels were capped.
- Remove those districts from the training set (and also from the test set, since your system should not be evaluated poorly if it predicts values beyond $500,000).

A Quick Dive

Looking at these histograms, you notice a few things:

- These attributes have very different scales - Feature scaling.
- Many histograms are **skewed right**: This may make it a bit harder for some machine learning algorithms to detect patterns.
- Try **transforming** these attributes to have more symmetrical and **bell-shaped distributions**.

A Quick Dive

Looking at these histograms, you notice a few things:

- Total Rooms and Total Bedrooms: Both histograms exhibit a right-skewed distribution, where most of the data is concentrated on the left, with a long tail extending to the right.
- This indicates that most districts have a smaller number of rooms and bedrooms, with fewer districts having a very large number of rooms or bedrooms.