Part 1.5 | Day 7 | Panel Data (Wide Format)

Two Formats, Same Data

Panel data can be stored in two ways:

- **Long format:** Each observation is a separate row, with a column identifying the group
- **Wide format:** Each time period is a separate column

Same information, different shapes. Different shapes make different tasks easier.

Wide Format: Coffee Consumption (kg per capita)

Code	1999	2004	2009	2014	2019
FRA	5.5	4.7	5.3	5.4	5.5
DEU	7.1	7.6	6.5	6.4	6.3
JPN	3.0	3.3	3.3	3.5	3.6
GBR	2.3	2.5	3.1	2.7	3.4
USA	4.2	4.3	4.2	4.5	5.0

In long format, years are *values* in a column. In wide format, years become *column names*.

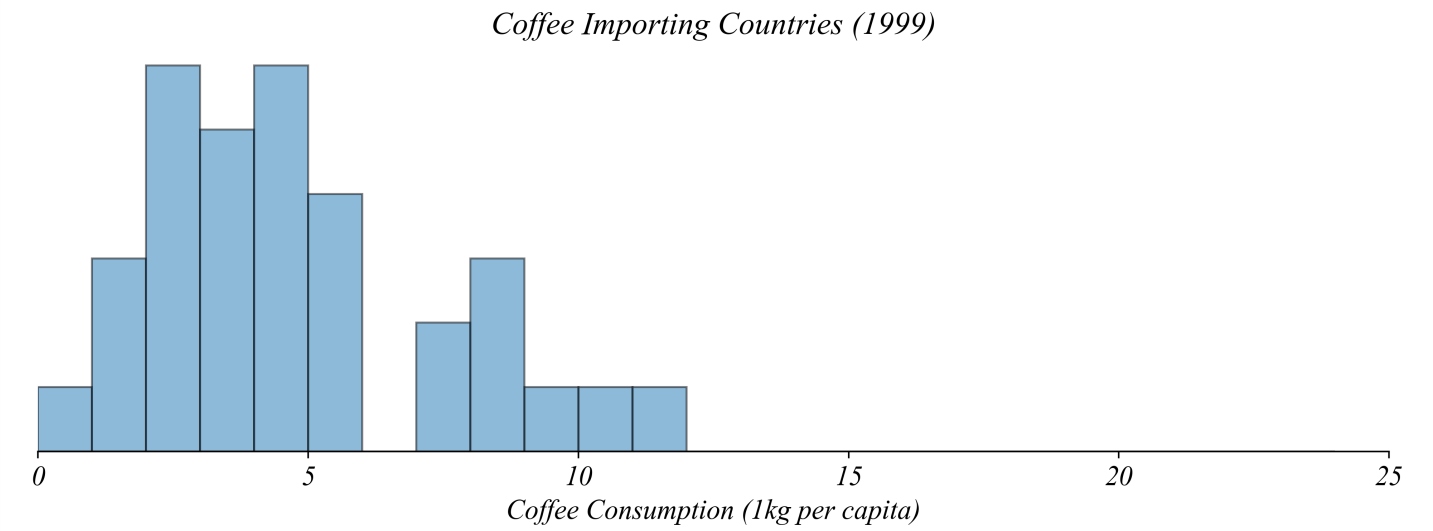
Long Format: Coffee Consumption

Code	Year	Consumption
FRA	1999	5.5
DEU	1999	7.1
JPN	1999	3.0
GBR	1999	2.3
USA	1999	4.2
FRA	2004	4.7
DEU	2004	7.6
JPN	2004	3.3
GBR	2004	2.5
USA	2004	4.3
FRA	2009	5.3
DEU	2009	6.5

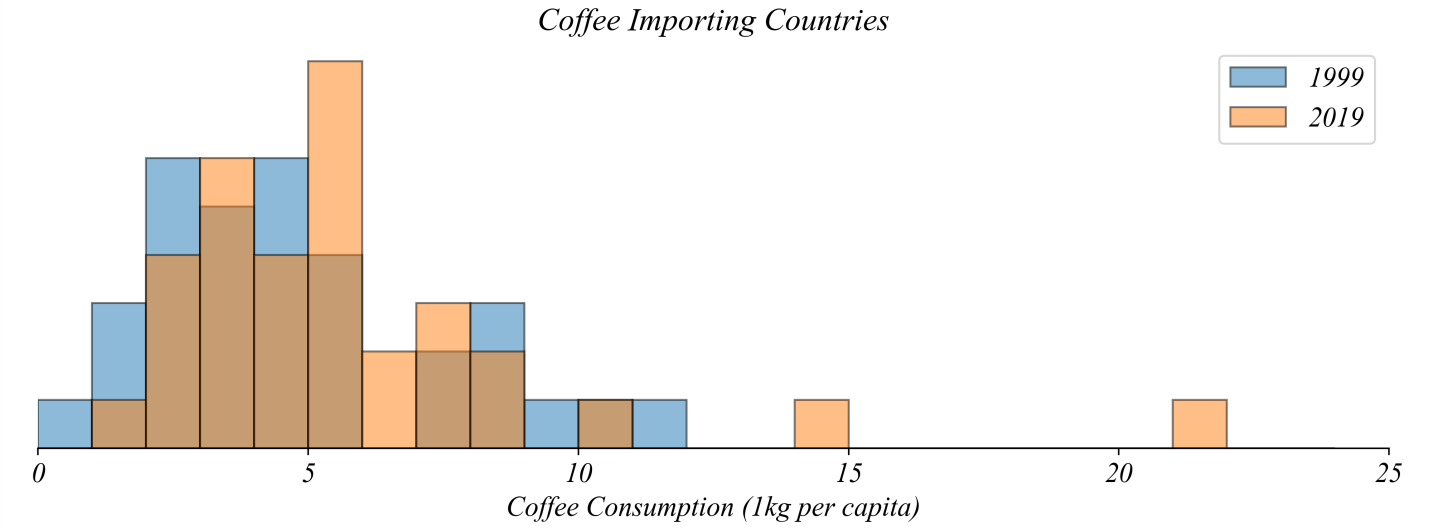
Comparing Distributions: Histograms

The world seems to be drinking more coffee than ever. But does the data on coffee consumption confirm this? This data contains coffee consumption in kilograms per capita of 34 coffee importers over the span of two decades.

Let's plot the histogram of country's coffee consumption for 1999.



This figure shows coffee consumption with a bin of 1kg. How did the consumption change between 1999 and 2019? To start the investigation, we can add 2019 coffee consumption per capita to this histogram.



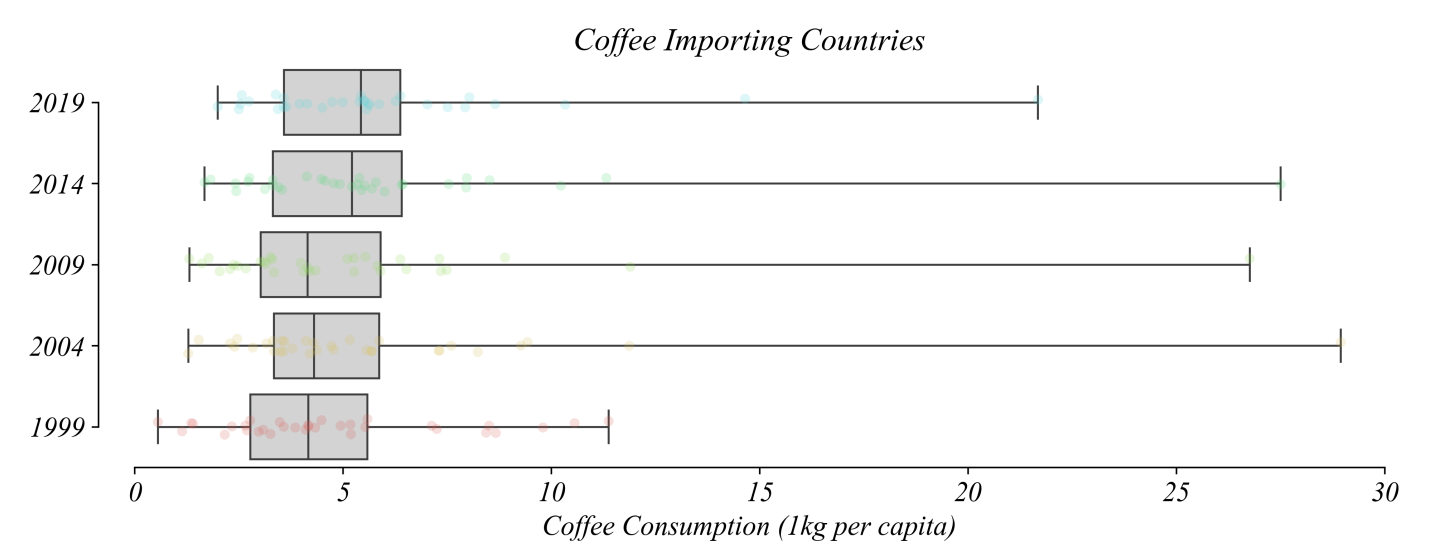
What can we conclude from the histograms?

- Some countries increased their per capita coffee consumption.
- No country exceeded 20 kg per capita in 1999, and one country exceeded 20 kg per capita in 2019.
- We don't know which country is represented by which bar, some countries might have decreased their coffee consumption, although we can't say for sure.

The histograms suggest a general increase in coffee consumption — but they're not great for comparison. Let's use a multi-boxplot instead.

Boxplots

A visualization type useful for comparing multiple distributions is a **box and whisker plot**, or **boxplot**. The boxplot can represent the same data by summarizing it.

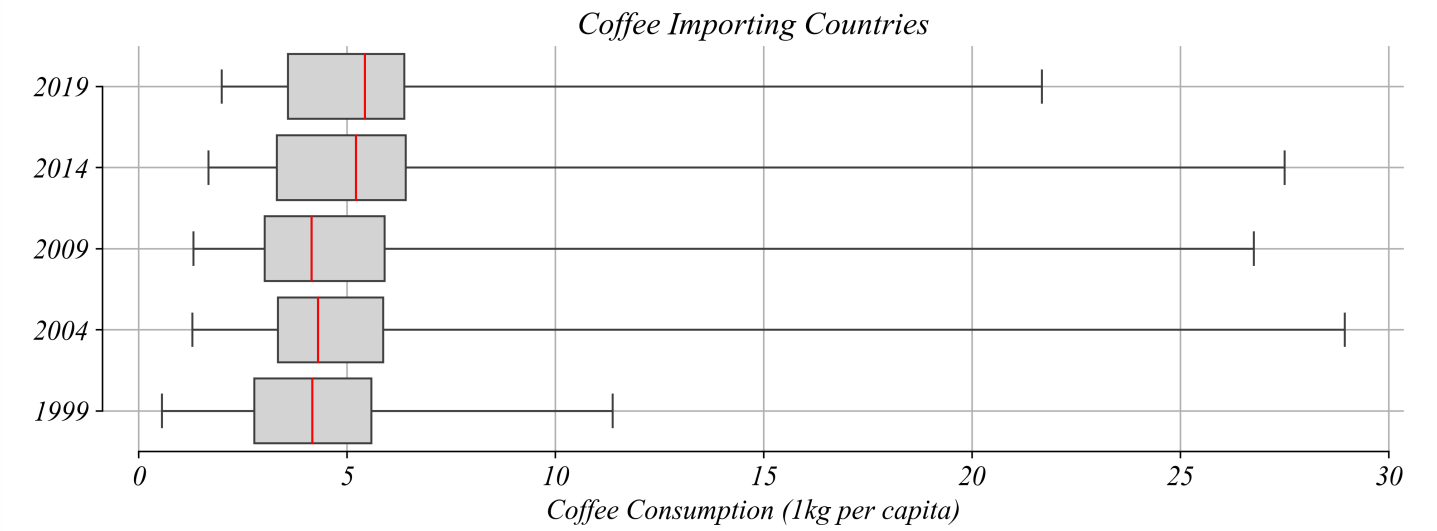


To aid our discussion, I'm adding in the countries scattered across the horizontal. Each point corresponds to a country and their coffee consumption on the horizontal. The vertical axis is 'jittered' to make it easy to see countries which are clumped together.

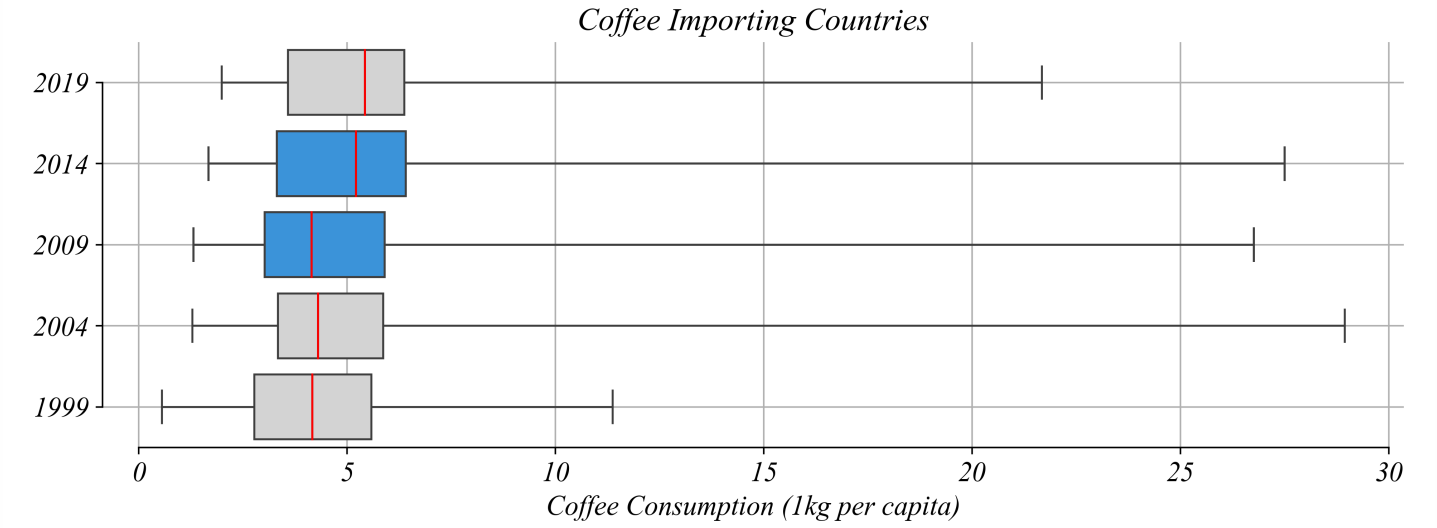
Boxplots visually summarize the data — but their real power lies in the ease of comparisons between distributions. Next, we'll use boxplots to analyze the changes in coffee consumption between 1999 and 2019.

Comparing Boxplots

Now that we understand what boxplots represent, we'll analyze coffee consumption data in smaller time increments. Each boxplot represents data from a single year. For convenience, we'll use horizontal boxplots.



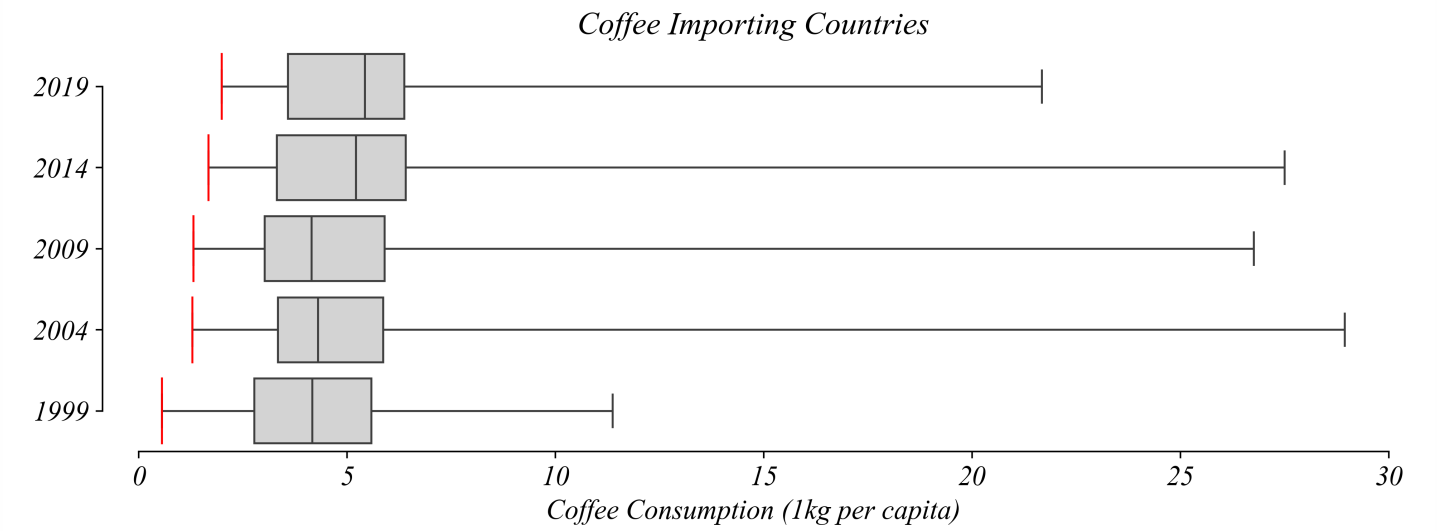
Based on the boxplots, when did the typical coffee consumption (median) increase the most?



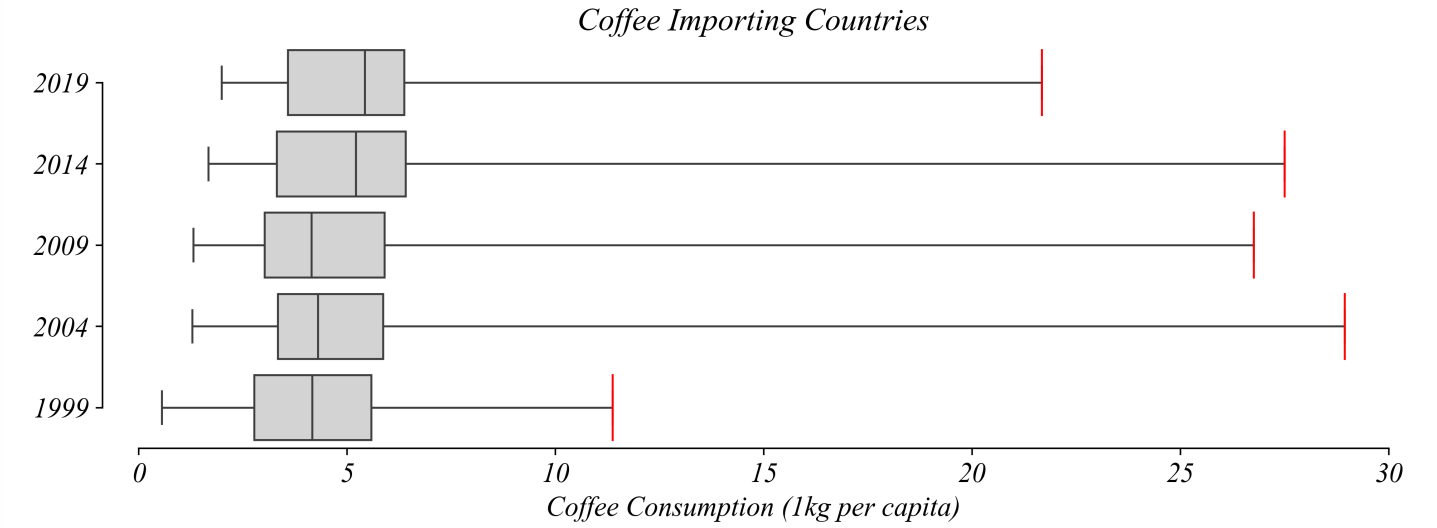
Between 2009 and 2014. The median consumption per capita — represented by the middle line in the box — stayed just below 5 kg until 2009, and then increased to above 5 kg.

The boxplots show that median consumption was more or less stable between 1999 and 2009, and then suddenly shifted by over 1 kg per capita. This would be much harder to notice by comparing five histograms.

What happened between all the visualized years in the box plot? The minimum consumption increased.

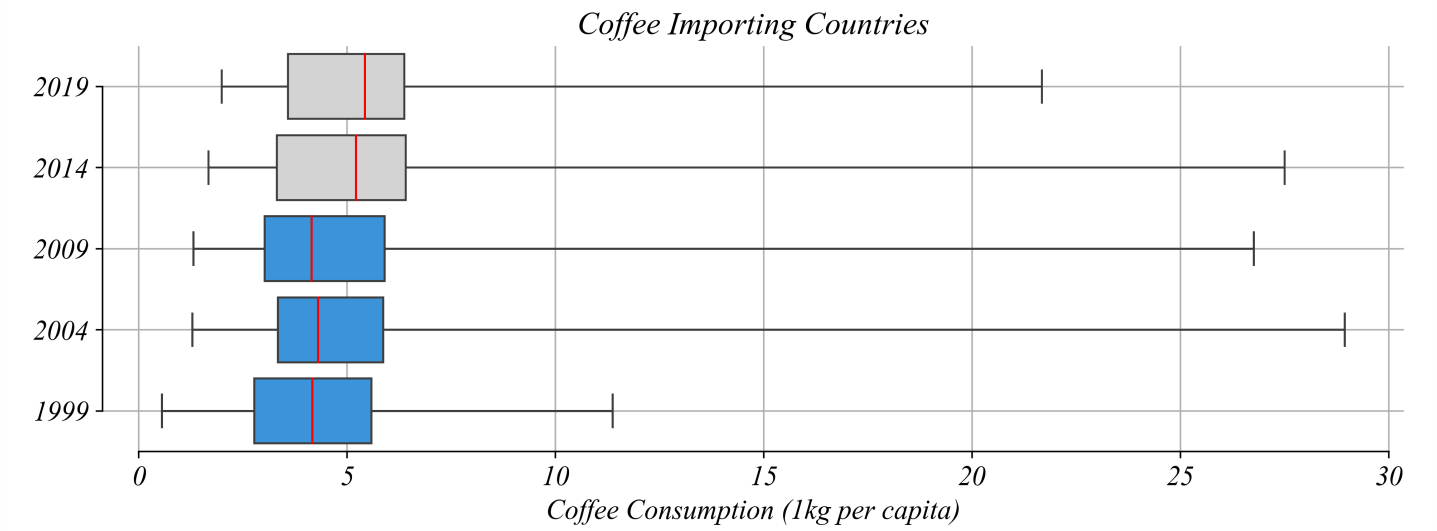


In each boxplot, the minimum consumption is slightly larger than in the previous boxplot. The pattern of maximum consumption isn't as clear.



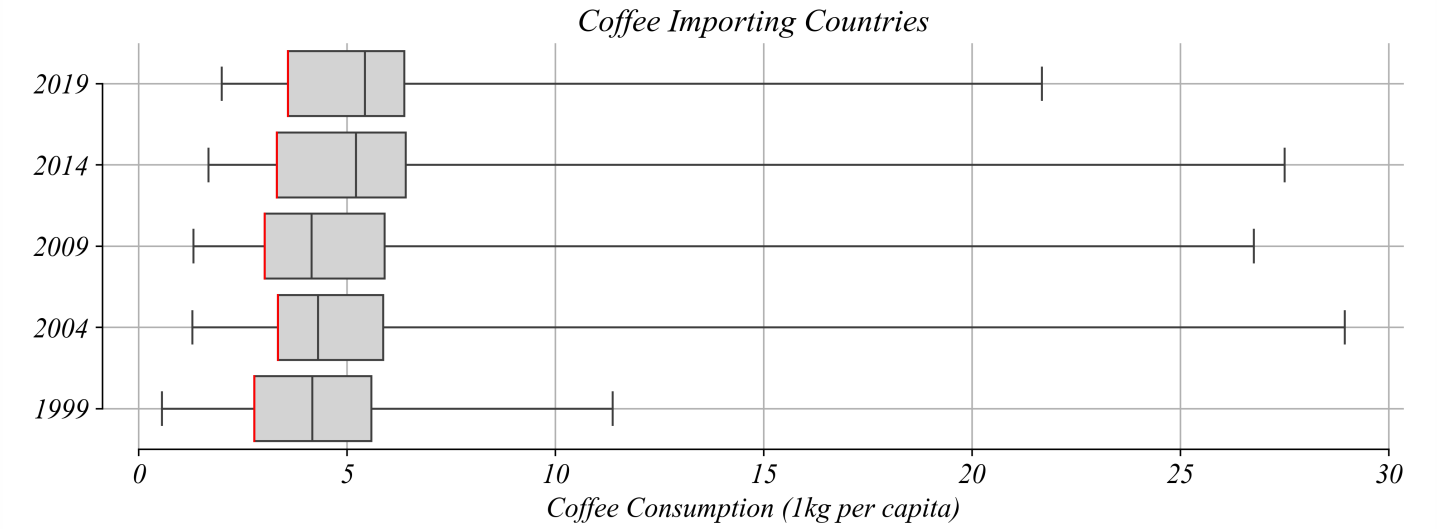
For example, it increased between 1999 and 2004 but decreased between 2004 and 2009. The typical consumption hovered around 5 kg per person, so let's explore this value in more detail.

Which years show at least half of the countries consuming less than 5 kg of coffee per capita?



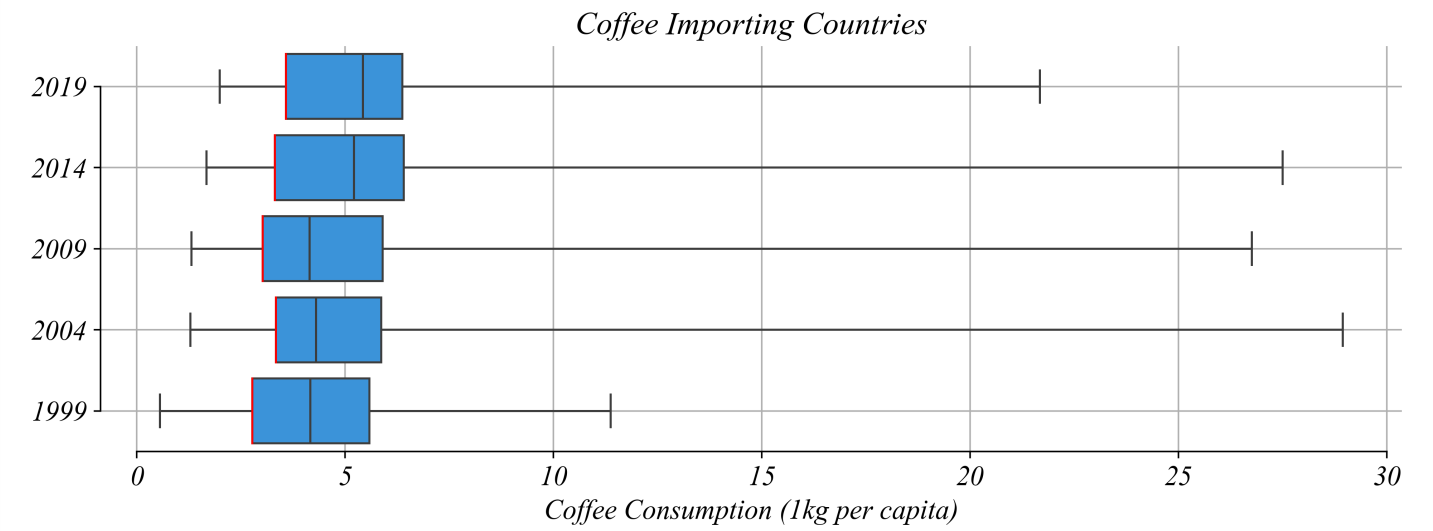
In each boxplot, half of the countries consume more than the median, and half less than the median. In 1999, 2004, and 2009, the median was smaller than 5 kg, so at least half of the countries consumed less than 5 kg per capita.

In which years are more than 25% of the countries consuming less than 5 kg of coffee per capita?

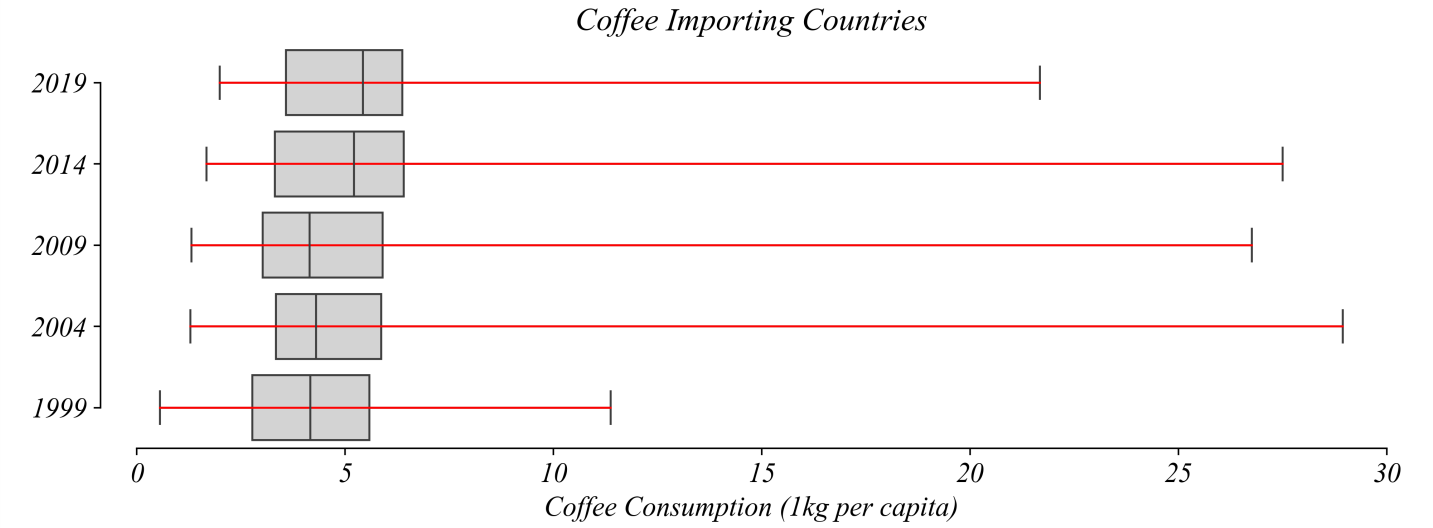


In all five years, Q1 was smaller than 5 kg, so more than 25% of the countries consumed less than 5 kg.

Which year has the greatest range of consumption values?



The minimum consumption didn't differ much between years. The maximum consumption, however, was the largest in 2004, which makes the range of values the largest that year.

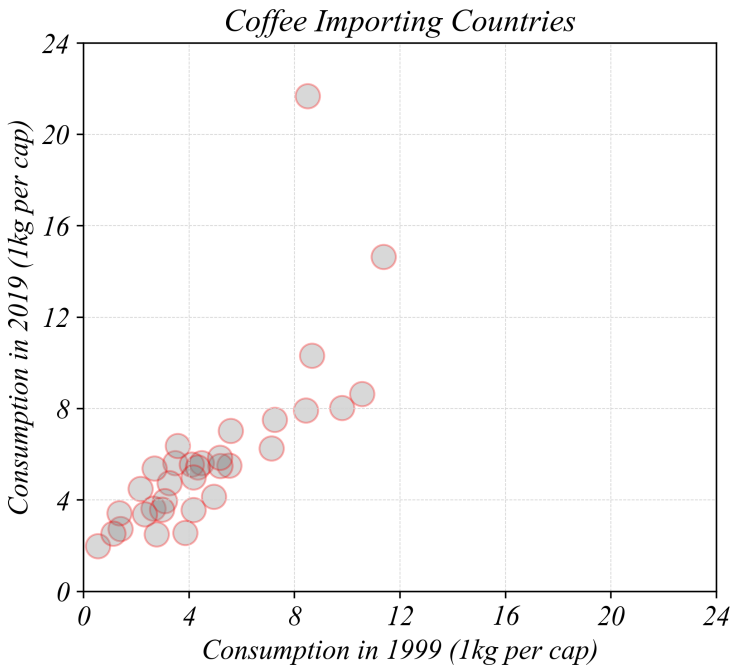


Thanks to boxplots, we saw that while coffee consumption increased between 1999 and 2019, the increase wasn't uniform over the years.

Scatterplot Changes

When aggregating data like this we can see what's going on overall. But we also might want to get a better view of individual changes. We've seen that coffee consumption has gone up overall, but does that mean all countries have increased their coffee consumption during these years? We don't have the right view to answer that question yet.

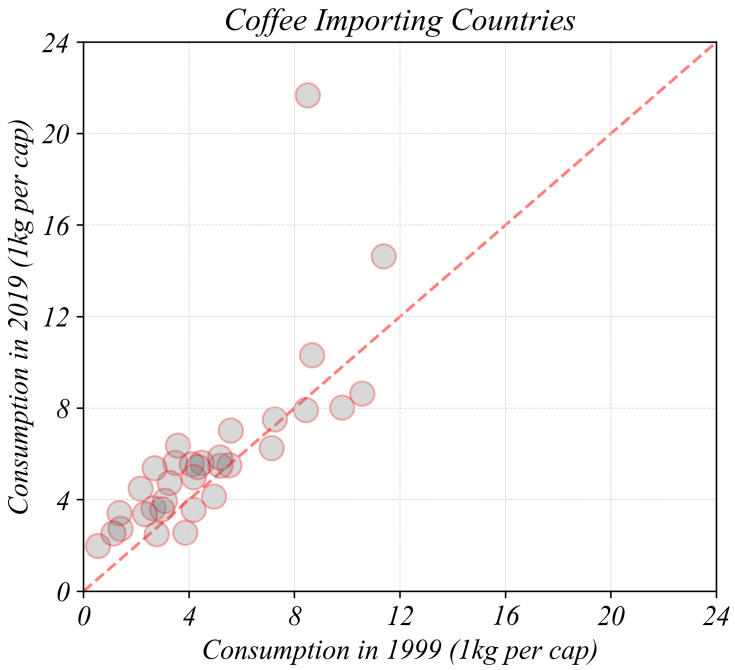
We're going to go back to our trusty scatter plot. We have multiple years to examine, which gives us the ability to explore the relationship between coffee consumption in each country between any two years. Let's focus on 1999 and 2019.



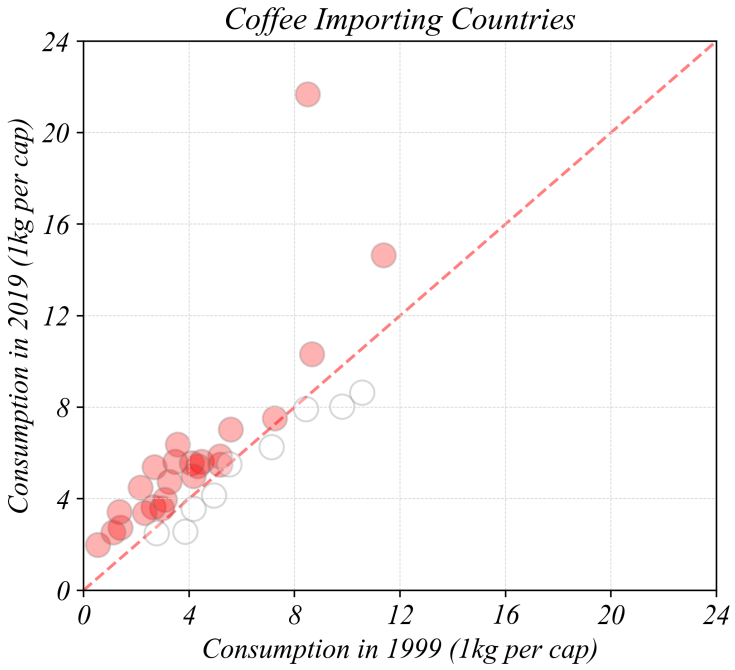
Each point represents a country. The horizontal axis shows that country's coffee consumption in 1999, and the vertical axis shows consumption in 2019.

The 45-Degree Line

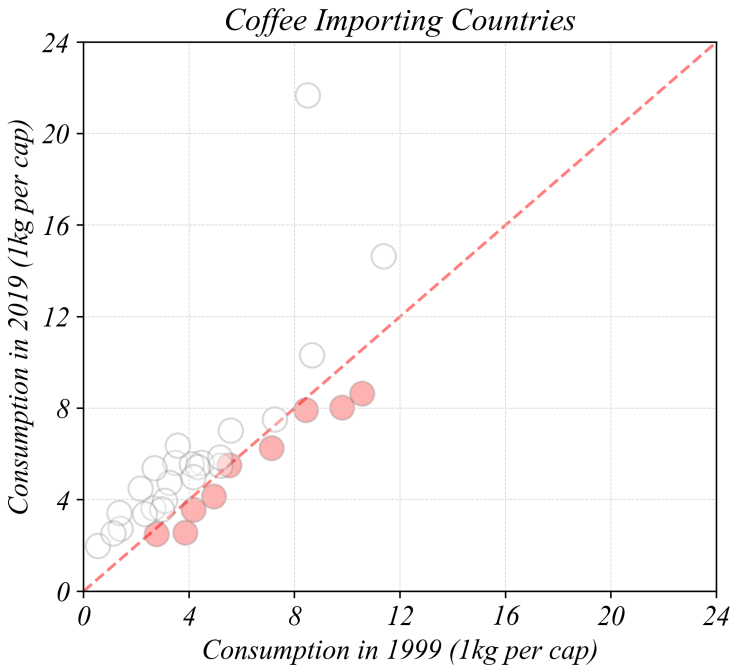
If a country drank the same amount in both years, where would it appear? We can add a 45-degree line. Any point on this line represents a country with identical consumption in both years.



- Countries **above** the line increased their consumption
- Countries **below** the line decreased their consumption



We can count points above and below the line to see how many countries increased vs decreased. We can use color to make this even clearer.



This view gives us something the boxplots couldn't: we can track individual countries across time. The boxplots showed that the overall distribution shifted upward, but they couldn't tell us whether *every* country increased or just some. The scatter plot reveals that while most countries increased consumption, a few actually decreased.

Filtering: Counting Changes

We can see visually that most points are above the 45-degree line. But how do we count exactly how many countries increased or decreased?

First, we create a column that calculates the change for each country:

```
# Create a change column
percap['change'] = percap['2019'] - percap['1999']
```

Now we can filter to count how many increased (change > 0) and how many decreased (change < 0):

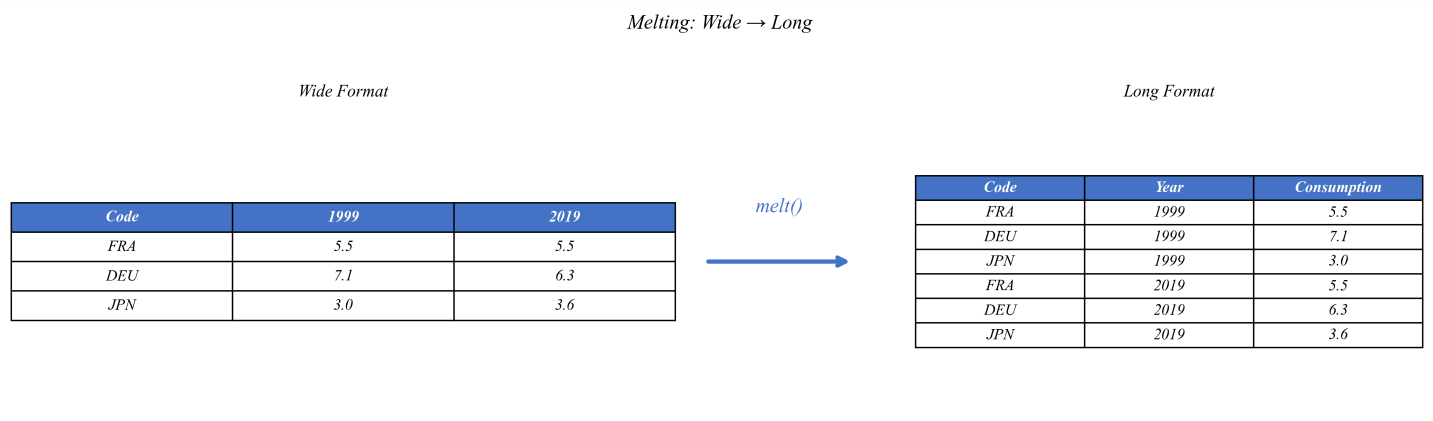
```
# Count countries that increased
increased = percap[percap['change'] > 0]
len(increased)

# Count countries that decreased
decreased = percap[percap['change'] < 0]
len(decreased)
```

Filtering uses square brackets with a condition inside. The expression `percap['change'] > 0` returns True or False for each row, and putting it inside brackets keeps only the rows where it's True.

We'll explore filtering more systematically in Part 2.3, but this basic pattern — create a column, then filter by a condition — is useful for answering "how many" questions.

Reshaping: Wide to Long

We can convert between formats. Going from wide to long is called "melting" or "unpivoting."



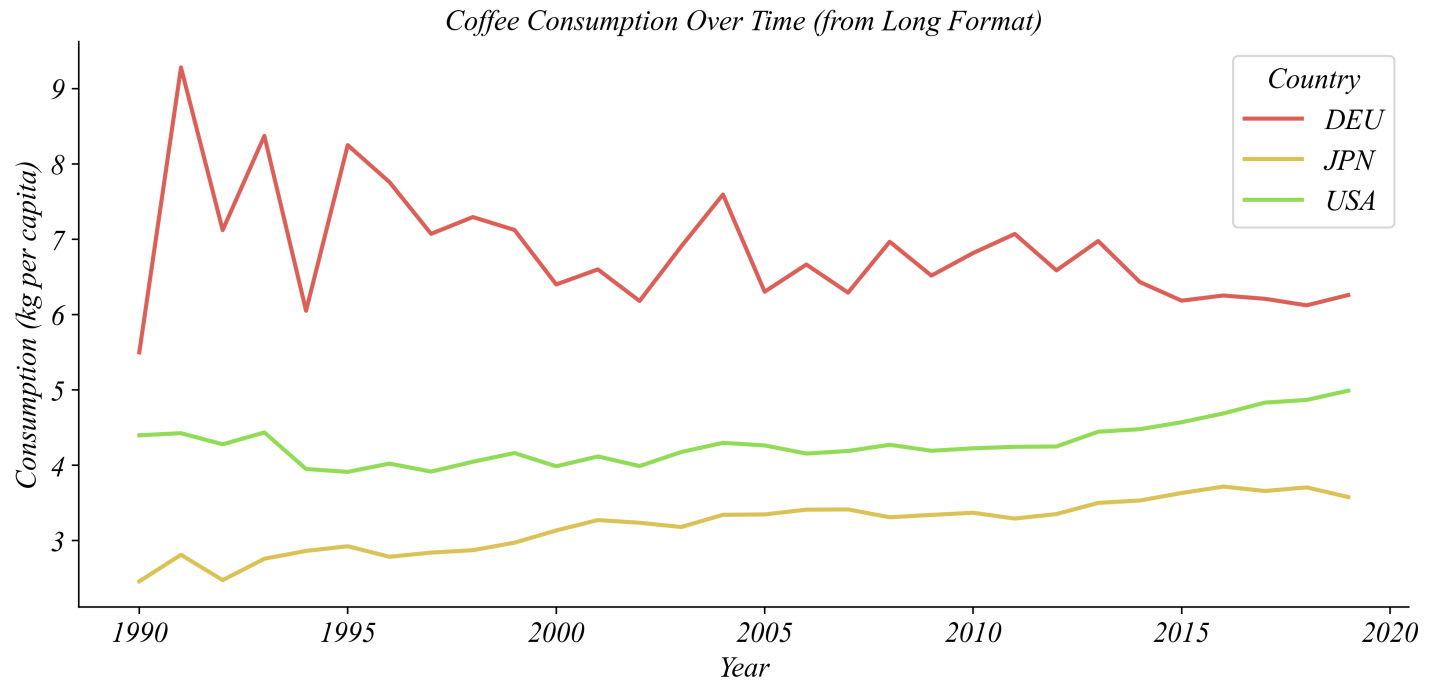
Code	1999	2019
FRA	5.5	5.5
DEU	7.1	6.3
JPN	3.0	3.6

Code	Year	Consumption
FRA	1999	5.5
DEU	1999	7.1
JPN	1999	3.0
FRA	2019	5.5
DEU	2019	6.3
JPN	2019	3.6

In Python pandas, use `melt()`:

```
# Wide to Long
long_df = wide_df.melt(
    id_vars=['Code'],        # Keep as identifier columns
    var_name='Year',         # Name for the former column headers
    value_name='Consumption' # Name for the values
)
```

`id_vars` specifies columns to keep as identifiers. Everything else gets "melted" into rows. Each year column becomes rows in a new "Year" column.



Reshaping: Long to Wide

Going from long to wide is called "pivoting" or "spreading."

In Python pandas, use `pivot()`:

```
# Long to Wide
wide_df = long_df.pivot(
    index='Code',            # What becomes rows
    columns='Year',          # What becomes columns
    values='Consumption'     # What fills the cells
)
```

Each unique Year value becomes its own column.

When to Use Which Format

Choose based on what you're trying to do:

Task	Better Format
Line plot over time	Long
Faceted plots by group	Long
Compare two specific years	Wide
Scatterplot (Year1 vs Year2)	Wide
Multi-boxplot across years	Wide

Key insight:

- **Long format:** Good when you want to *group by* or *color by* a variable
- **Wide format:** Good when you want to *compare* or *correlate* specific columns
- Neither is "better" — it depends on your task

Python Exercise 1.5 | Wide Format Visualizations

Multi-Boxplots with Wide Format:

With wide-format panel data, we can pass multiple columns directly to seaborn:

```
# Wide Format Multi-Boxplot
sns.boxplot(percap[['1999','2004','2009','2014','2019']], orient='h', whis=(0, 100))
```

Scatterplot Comparing Years:

In Python with wide-format data, we can directly plot two columns against each other:

```
# Wide Format Scatterplot
sns.scatterplot(percap, x='1999', y='2019')
```

To add a 45-degree line:

```
# Scatterplot with 45-degree line
sns.scatterplot(percap, x='1999', y='2019')
plt.plot([0, 15], [0, 15], color='red', linestyle='--', label='No change')
plt.legend()
```

Filtering to Count Changes:

```python
# Create a change column
percap['change'] = percap['2019'] - percap['1999']

# Count increases and decreases
increased = percap[percap['change'] > 0]
decreased = percap[percap['change'] < 0]
print(f"Increased: {len(increased)}, Decreased: {len(decreased)}")
```

Reshaping:

```python
# Wide to Long
percap_long = percap.melt(id_vars=['Code'], var_name='Year', value_name='Consumption')

# Long to Wide
percap_wide = percap_long.pivot(index='Code', columns='Year', values='Consumption')
```

Summary

Part 1.5 covered wide format panel data and its applications:

- **Wide format:** Each time period is a column
- **Multi-boxplots** compare distributions across time periods
- **Scatterplots with 45° lines** track individual changes between two time points
- **Filtering** with `df[df['col'] > 0]` counts subsets
- **melt()** converts wide → long (columns become rows)
- **pivot()** converts long → wide (rows become columns)