ES 2: Critical thinking with Python

aka, Introduction to Computing for Engineers

Steven Bell

13 March 2025

Tufts UNIVERSITY

Two ways to use Python

`python` in terminal

Scripting

Write code in a .py file

Run with `python hello.py`

Results printed in terminal

Good for writing programs

Interactive REPL

Type commands one at a time

Commands run on "enter"

Results printed immediately

Good for experimenting

Interactive help()

breakpoint()

exit() or CTRL-D

Pandas works with tabular data

Download the CSV file from the website, and open it in a spreadsheet program (Excel, Numbers, LibreOffice Calc)

Or open the Google Sheets link

Read a CSV into Python with Pandas

```
import pandas as pd
df = pd.read_csv("CSV_FILE.csv")
```

Whatever variable name you want to use for the whole DataFrame

replace with actual name or system path to CSV file

Load the MBTA data

Exploring data

```
df.columns
```

```
len(df)
```

How many rows and columns are in the MBTA dataset?

Indexing

Square brackets mean indexing:
"pick out a specific piece of this larger data structure"

```
my_list[0]
```

```
my_dictionary["name"]
```

Pandas extends this for dataframes:

`df["some_column_name"]`	Get a column by name
`df.iloc[0]`	Get row 0

Iterating through a table

Usually it's better to use Pandas operators to filter and calculate directly, but you can do:

```
for idx in range(len(df)):
  row = df.iloc[idx]
  # Do whatever you want with this row
```

Computing on columns

We can do math with a whole series

```
df["ONE_COLUMN"] + df["ANOTHER_COLUMN"]
```

And save that in another variable

```
both = df["ONE_COLUMN"] + df["ANOTHER_COLUMN"]
```

Or even make a new column in the dataframe

```
df["total"] = df["ONE_COLUMN"] + df["ANOTHER_COLUMN"]
```

Try it!

Find the total of the "trip purpose" columns.

Find the total of the "fare" columns.

Analysis

The following work on a data Series
(either standalone or a DataFrame column)

```
.sum(), .mean(), .std()
.min(), .max()
.idxmax(), idxmin()
```

Filtering

We can compare a whole series with a data value

```
df["COLUMN"] > 100
```

```
df["Mode"] == "Bus"
```

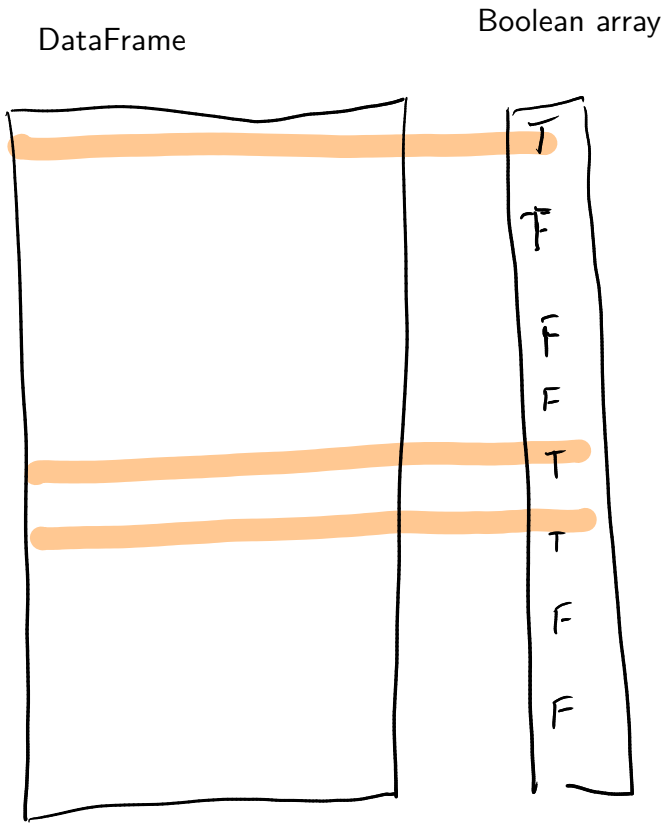
The result is a boolean (true/false) array

Using a boolean array to pick rows

We can compare a whole series with a data value

```
df[BOOLEAN_ARRAY]
```

```
bus = df[df["Mode"] == "Bus"]
```



Which bus line has the highest
`"Trip purpose: Home-based work"`?

with idxmax

iloc vs loc

`iloc` is an integer index into the subset

`loc` is an index into the whole dataset

Is that a lot?

Every time we pre-rinse our dishes, we can waste up to 20 gallons of water. Together we can save up to 150 billion gallons of water. That's billions. With a B. With Finish, you can Skip the Rinse.

https://www.finishdishwashing.com/skip-the-rinse