DSFA

Spring 2021

Lecture 5

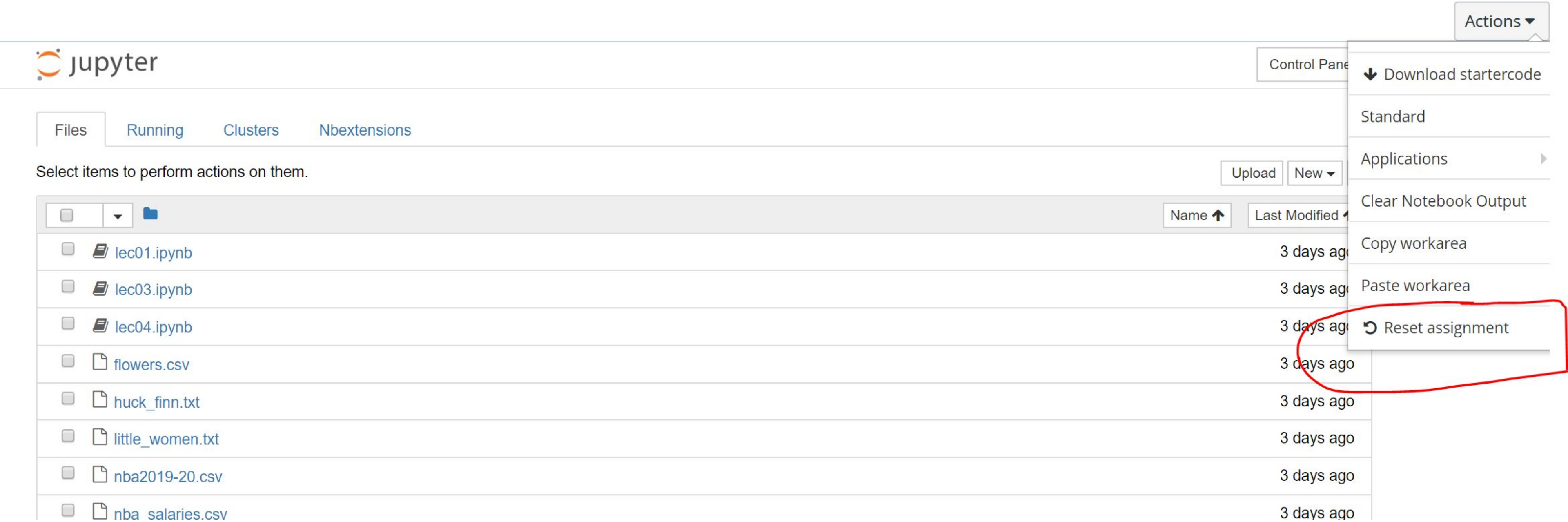
Columns and Rows

Announcements

- Course website: cornell-dsfa.org.
- What if you just added?
- HW 1 due today by 5:59PM, HW 2 out today, due next Friday at 5:59PM (with bonus point for turning in on Thursday).
- Please:
  - Office hours and Ed Discussions is your first point of contact for questions about the course
  - Your section TA is your first point of contact for questions about your personal logistics

Announcements

- If you want to follow along for the lecture notebook, go to Canvas->Assignments->Lecture Demos.
  - If lec05.ipynb not there, try clicking “Actions” (upper right corner), “Reset assignment”. Note: this will erase (I think) whatever you had in previous lecture demo notebooks.
  - Note: You need to run the first cell in the demo!



Select items to perform actions on them.

Name	Last Modified
lec01.ipynb	3 days ag
lec03.ipynb	3 days ag
lec04.ipynb	3 days ag
flowers.csv	3 days ago
huck_finn.txt	3 days ago
little_women.txt	3 days ago
nba2019-20.csv	3 days ago
nba_salaries.csv	3 days ago

Arrays and Ranges

Arrays

An array contains a sequence of values

- All elements of an array should have the same type
- Arithmetic is applied to each element individually
- When two arrays are added, they must have the same size; corresponding elements are added in the result
- A column of a table is an array

(Demo)

Ranges

A range is an array of consecutive numbers

- **`np.arange(end)`**:
  An array of increasing integers from 0 up to **`end`**
- **`np.arange(start, end)`**:
  An array of increasing integers from **`start`** up to **`end`**
- **`np.arange(start, end, step)`**:
  A range with **`step`** between consecutive values

The range always includes **`start`** but excludes **`end`**

Columns

Ways to create a table

- `Table.read_table(filename)` - reads a table from a spreadsheet
- `Table()` - an empty table
- and...

Arrays → Tables

- `Table().with_column(label, data)` - creates a table with a single column; `data` is an array
- `Table().with_columns(label1, data1, ...)` - creates a table, with an array of data for each column

Table Methods

- Creating and extending tables:
  - **`Table().with_columns`** and **`Table.read_table`**
- Finding the size: **`num_rows`** and **`num_columns`**
- Referring to columns: labels, relabeling, and indices
  - **`labels`** and **`relabeled`**; column indices start at 0
- Accessing data in a column
  - **`column`** takes a label or index and returns an array
- Using array methods to work with data in columns
  - **`item`**, **`sum`**, **`min`**, **`max`**, and so on
- Creating new tables containing some of the original columns:
  - **`select`**, **`drop`**

Rows

Take Rows, Select Columns

The `select` method returns a table with only some columns

The `take` method returns a table with only some rows

- Rows are numbered, starting at 0
- Taking a single number returns a one-row table
- Taking a list of numbers returns a table as well

(Demo)

The where method

- `t.where(label, condition)` - constructs a new table with just the rows that match the condition

(Demo)

Manipulating Rows

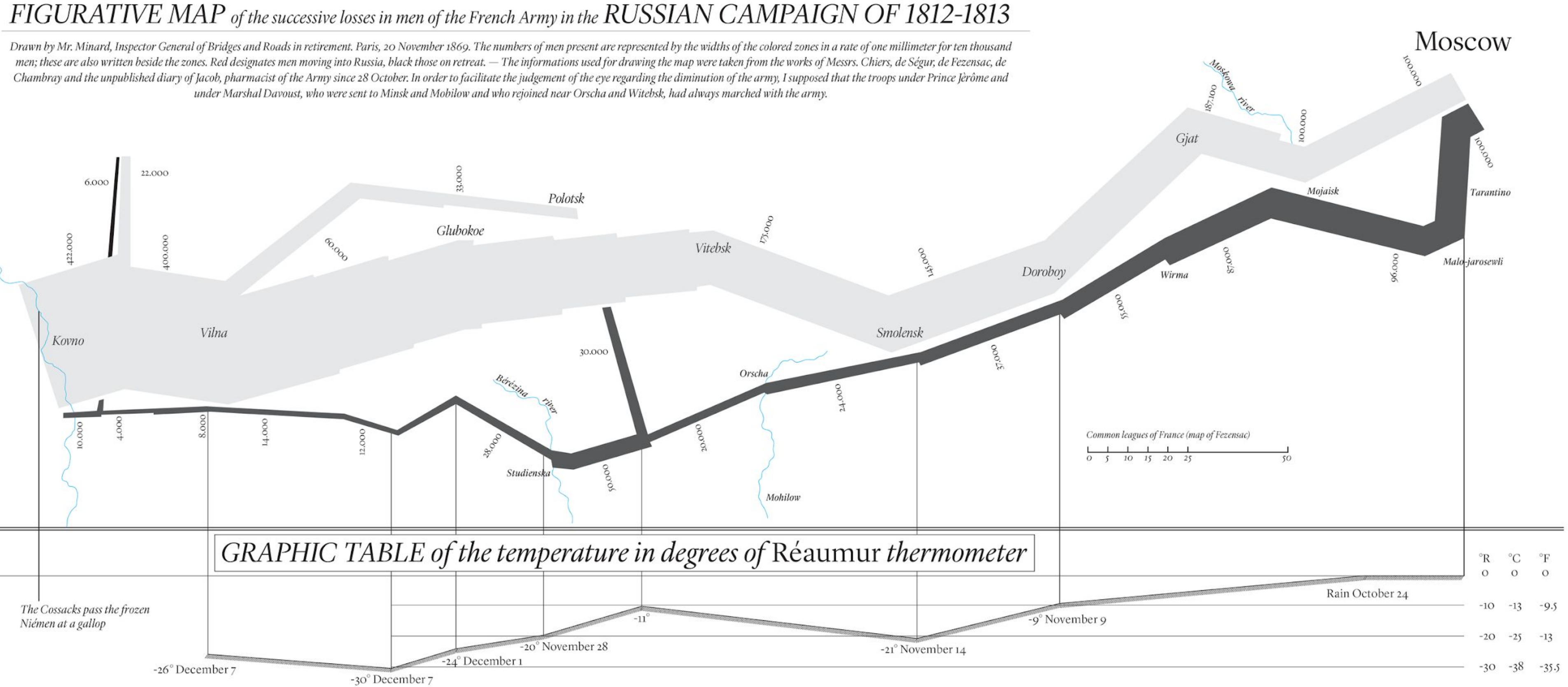
- `t.sort(column)` sorts the rows in increasing order
- `t.take(row_numbers)` keeps the numbered rows
  - Each `row` has an index, starting at 0
- `t.where(`*`column`*`, are.`*`condition`*`)` keeps all rows for which a column's value satisfies a condition
- `t.where(`*`column`*`, value)` keeps all rows containing a certain value in a column

Minard's Map

Charles Joseph Minard, 1781-1870

- French civil engineer who created one of the greatest graphs of all time
- Visualized Napoleon's 1812 invasion of Russia, including
    - the number of soldiers
    - the direction of the march
    - the latitude and longitude of each city
    - the temperature on the return journey
    - Dates in November and December

Visualization of 1812 March



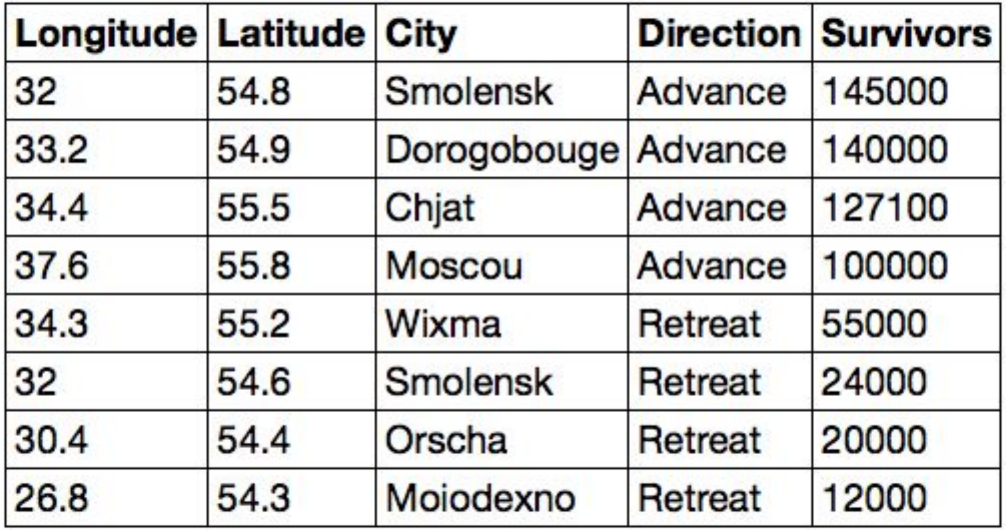
FIGURATIVE MAP *of the successive losses in men of the French Army in the* RUSSIAN CAMPAIGN OF 1812-1813

Drawn by Mr. Minard, Inspector General of Bridges and Roads in retirement. Paris, 20 November 1869. The numbers of men present are represented by the widths of the colored zones in a rate of one millimeter for ten thousand men; these are also written beside the zones. Red designates men moving into Russia, black those on retreat. — The informations used for drawing the map were taken from the works of Messrs. Chiers, de Ségur, de Fezensac, de Chambray and the unpublished diary of Jacob, pharmacist of the Army since 28 October. In order to facilitate the judgement of the eye regarding the diminution of the army, I supposed that the troops under Prince Jèrôme and under Marshal Davoust, who were sent to Minsk and Mobilow and who rejoined near Orscha and Witebsk, had always marched with the army.

GRAPHIC TABLE *of the temperature in degrees of* Réaumur *thermometer*

The Cossacks pass the frozen Niémen at a gallop

Different types of data



Longitude	Latitude	City	Direction	Survivors
32	54.8	Smolensk	Advance	145000
33.2	54.9	Dorogobouge	Advance	140000
34.4	55.5	Chjat	Advance	127100
37.6	55.8	Moscou	Advance	100000
34.3	55.2	Wixma	Retreat	55000
32	54.6	Smolensk	Retreat	24000
30.4	54.4	Orscha	Retreat	20000
26.8	54.3	Moiodexno	Retreat	12000

float:
decimal number

string:
text

int:
integer

Lists

Lists are Generic Sequences

A list is a sequence of values (just like an array), but the values can all have different types

```
[2+3, 'four', Table().with_column('K', [3, 4])]
```

If you create a table column from a list, it will be converted to an array automatically

(Demo)