Frictionless Data Intro

Reproducible RDM workflows
TU Delft April 2022

Lilly Winfree (lilly.winfree@okfn.org)

Hi! I'm Lilly

💻 Product Manager, Frictionless Data @ Open Knowledge Foundation

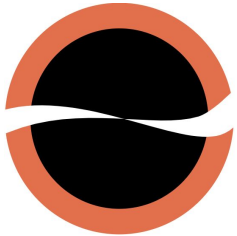
🧠 Neuroscience PhD

📊 Love open data + open science

🤠 Based in Austin, TX, USA

Frictionless Data for Reproducible Research

Removing the “friction” in research data to move from data to insight faster



FRICTIONLESS DATA

Open source & community focused:

https://github.com/frictionlessdata

What are some “frictions” in data?

Data cleaning

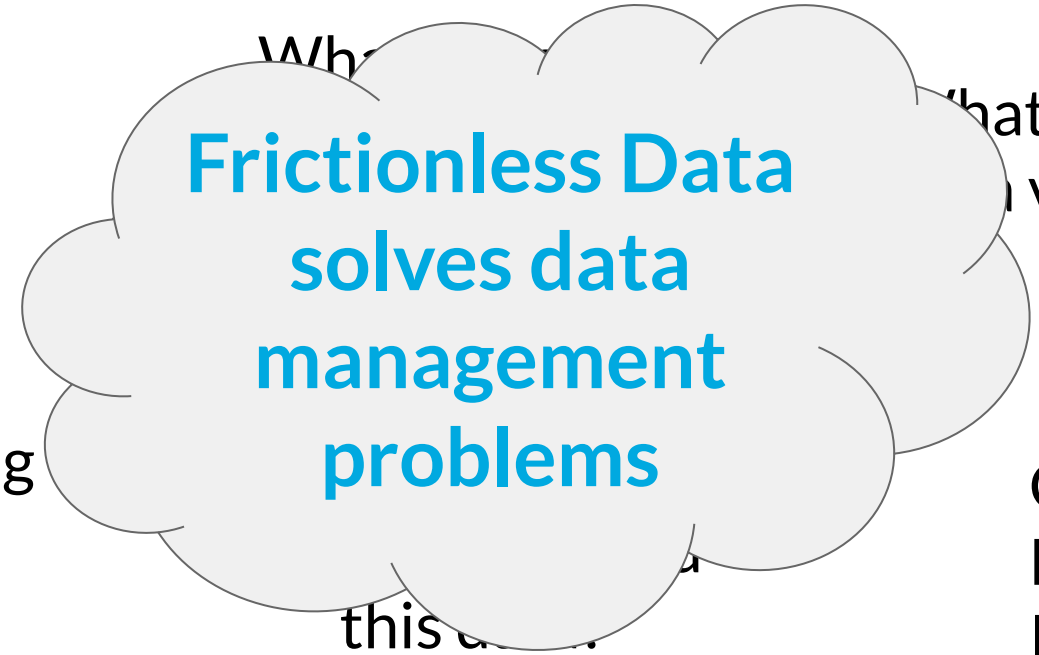
Wh

hat does this value mean?

Checking data quality

this d

Can I use this Excel data in Python?



Frictionless Data is open code, tools + standards

https://github.com/frictionlessdata

https://frictionlessdata.io/software

https://frictionlessdata.io/standards

Reproducible Data Workflows lead to better science

- What is a Reproducible Data Workflow?
- Why does it matter?
- We'll see how you can use Frictionless to create a Reproducible Data Workflow!

Frictionless Python Framework has 4 main functions

Describe Data

Extract Data

Validate Data

Transform Data

FRICTIONLESS FRAMEWORK

Example of a data dictionary

- `Data collected`: the date the data were collected in YYYY-MM-DD format
- `Species`: a code for the species of the animal detected See below for a table of what the codes stand for
- `Sex`: the sex of the animal detected `M` for male and `F` for female
- `Weight`: the weight of the animal detected measured in grams

Missing data is coded as `NA`.

species code	scientific name	common name
PF	Perognathus flavus	Silky pocket mouse
OT	Onychomys torridus	Southern grasshopper mouse
NA	Neotoma albigula	White-throated woodrat

Is this Machine Readable?

Slide text from https://eirini-zormpa.github.io/frictionless-data-workshop/data-organisation

Keep your data + metadata together in a datapackage

Data
e.g.,
experiment.csv

Metadata
+ schema
(this is like the data dictionary)

Data Package

https://frictionlessdata.io/standards/

Create machine readable metadata + schemas with Describe

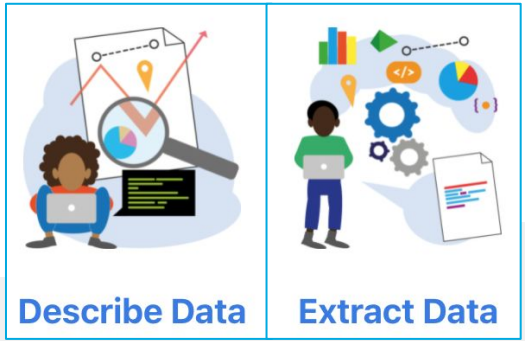
This is a JSON file

We can also use YAML

```
{
  "path": "combined.csv",
  "name": "combined",
  "profile": "tabular-data-resource",
  "scheme": "file",
  "format": "csv",
  "hashing": "md5",
  "encoding": "utf-8",
  "schema": {
    "fields": [
      {
        "type": "integer",
        "name": "record_id"
      },
      {
        "type": "integer",
        "name": "month"
      },
```

Metadata about the datapackage (aka ***Resource***)

Schema describing the data fields



Open Knowledge Foundation

Data importing and exporting problems

We've known since **2016** that many genetics papers (**1 in 5**) have mistakes because Microsoft Excel interprets gene names as dates.

MICROSOFT | REPORT | SCIENCE

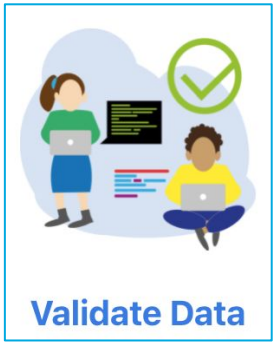
Scientists rename human genes to stop Microsoft Excel from misreading them as dates

Sometimes it's easier to rewrite genetics than update Excel

By James Vincent | Aug 6, 2020, 8:44am EDT

Also, for any **language scientists** out there working with a language that's not English, always use **UTF-8 text encoding**.

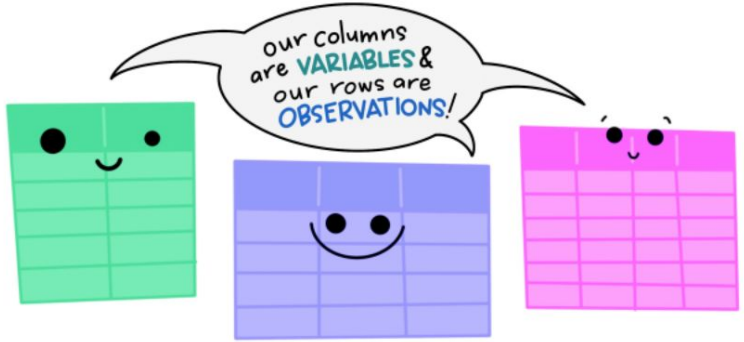
How could data validation help with this issue?



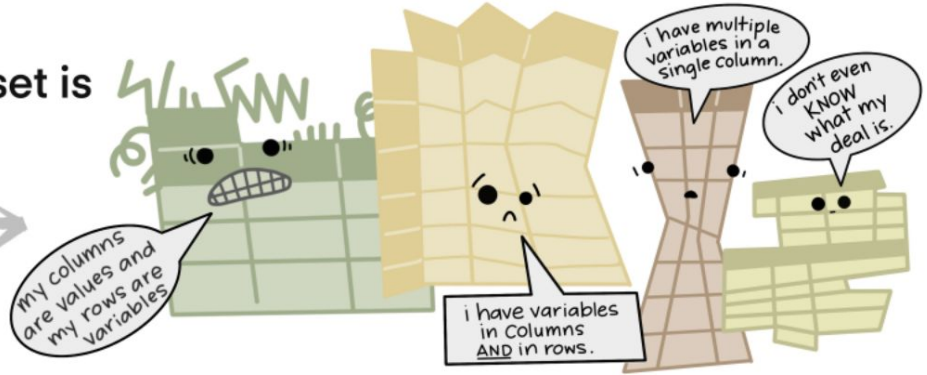
Slide text from https://eirini-zormpa.github.io/frictionless-data-workshop/data-organisation

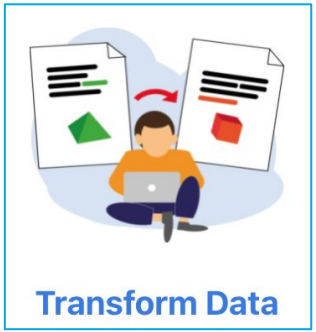
Transform your data by cleaning it

The standard structure of tidy data means that "tidy datasets are all alike..."



"...but every messy dataset is messy in its own way."
–HADLEY WICKHAM





Open Knowledge Foundation

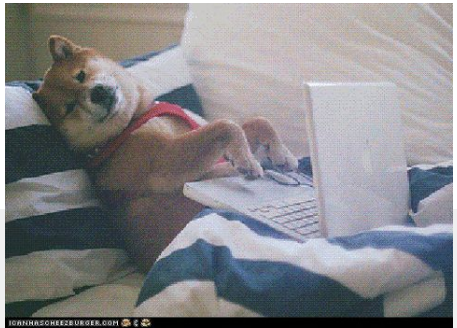
Illustrations from the Openscapes blog Tidy Data for reproducibility, efficiency, and collaboration by Julia Lowndes and Allison Horst.

@lilscientista

Let’s start coding!

In the terminal (Mac) or command prompt (Windows):

- Make sure you are in the same folder (directory) as the data you downloaded
- Type `jupyter notebook`
- This will open a Jupyter notebook in your internet browser
- If this does not work, raise your hand & we will help you
- As a backup, here is the code notebook: https://github.com/frictionlessdata/frictionless-py/blob/main/docs/tutorials/notebooks/frictionless-RDM-workflows.ipynb
- Helpers say hi!



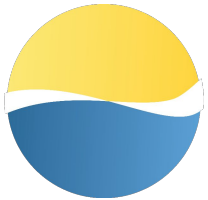
Welcome to Day 2!

Describe Data

Extract Data

Validate Data

Transform Data



FRICTIONLESS FRAMEWORK

Keep your data + metadata together in a datapackage

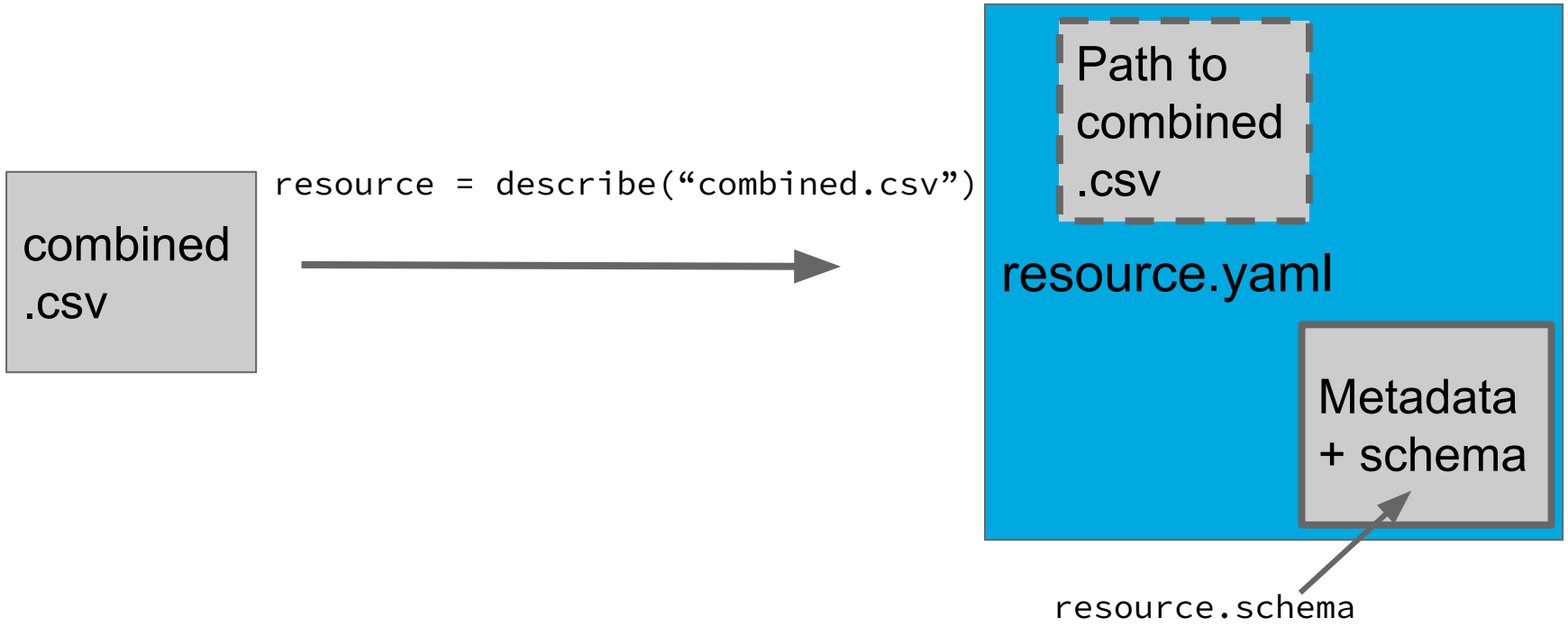
Data
e.g.,
experiment.csv

Metadata
+ schema
(this is like the data dictionary)

Data Package

https://frictionlessdata.io/standards/

Resource descriptor contains data + metadata/schema



Let’s start coding!

In the terminal (Mac) or command prompt (Windows):

- Make sure you are in the same folder (directory) as the data you downloaded
- Type `jupyter notebook`
- This will open a Jupyter notebook in your internet browser
- If this does not work, raise your hand & we will help you
- As a backup, here is the code notebook: https://github.com/frictionlessdata/frictionless-py/blob/main/docs/tutorials/notebooks/frictionless-RDM-workflows.ipynb
- Helpers say hi!