Top tips from a Data Specialist

James Thomas
github.com/jatonline

Jean Golding Institute
3rd June 2021

bristol.ac.uk/golding

 10.5281/zenodo.4899476

Thinking in spreadsheets.xlsx

~~Top tips from a Data Specialist~~

James Thomas

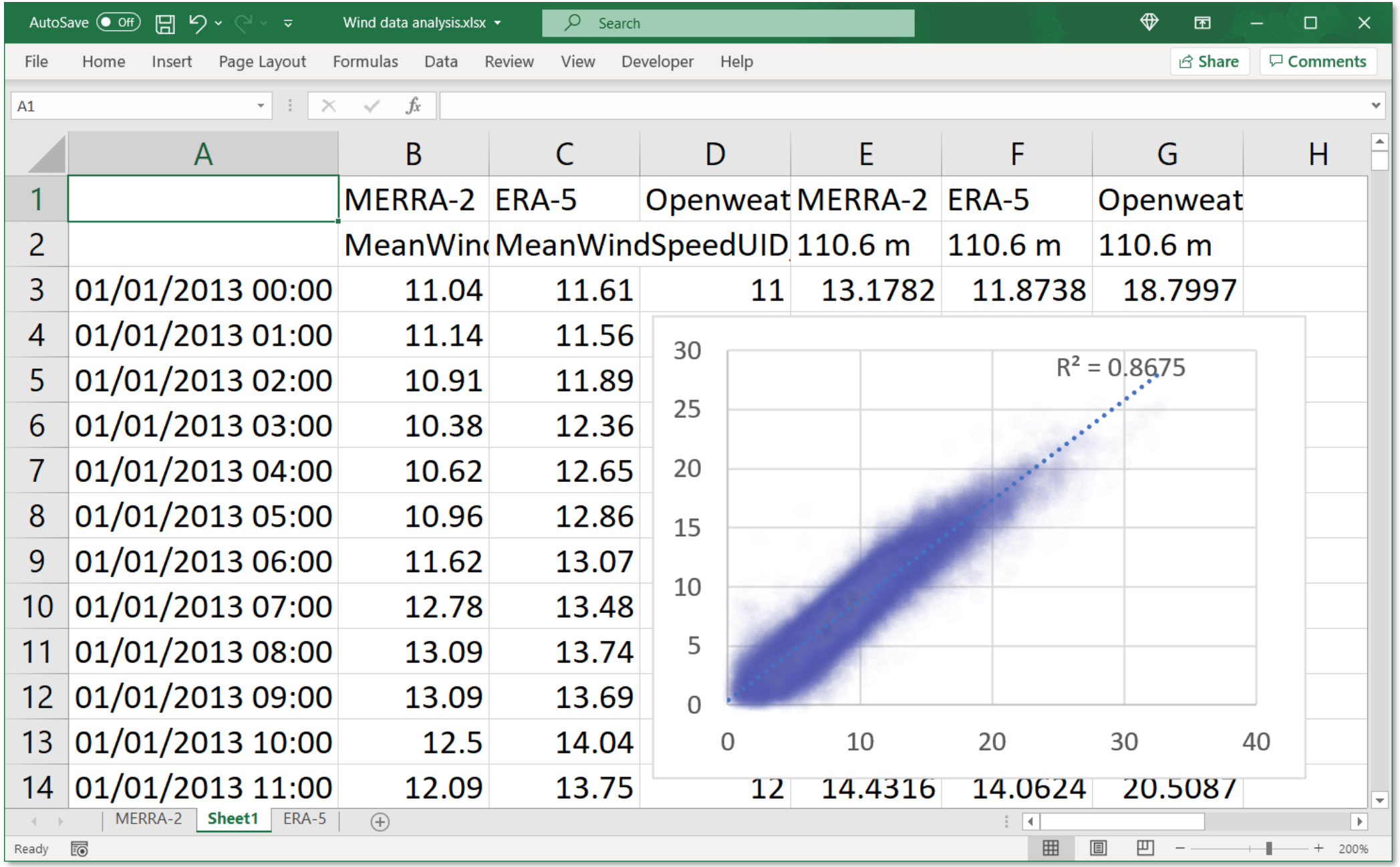
github.com/jatonline

Jean Golding Institute

3rd June 2021

bristol.ac.uk/golding

 10.5281/zenodo.4899476



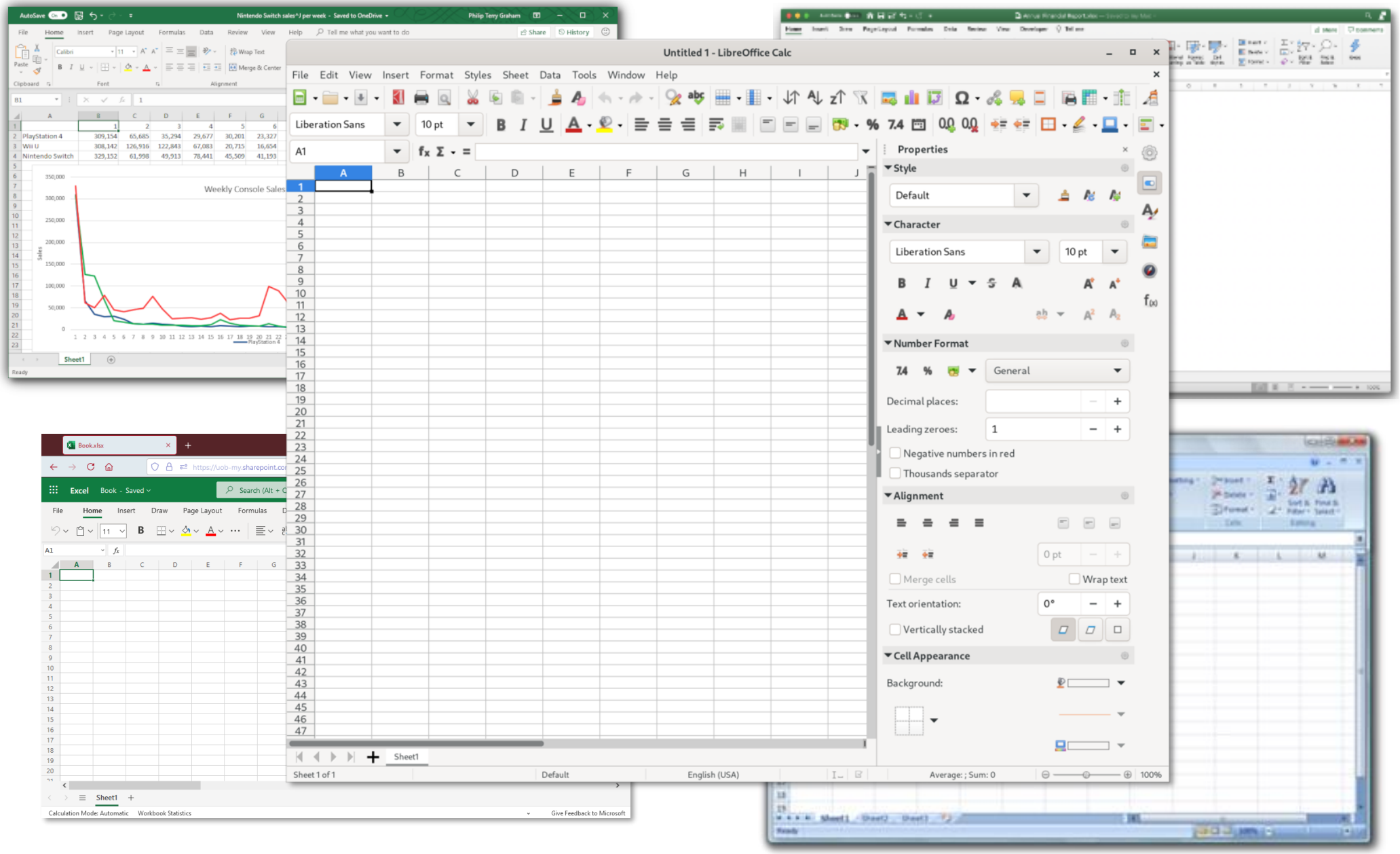
	A	B	C	D	E	F	G	H
1		MERRA-2	ERA-5	Openweat	MERRA-2	ERA-5	Openweat	
2		MeanWin	MeanWind	SpeedUID	110.6 m	110.6 m	110.6 m	
3	01/01/2013 00:00	11.04	11.61	11	13.1782	11.8738	18.7997	
4	01/01/2013 01:00	11.14	11.56					
5	01/01/2013 02:00	10.91	11.89					
6	01/01/2013 03:00	10.38	12.36					
7	01/01/2013 04:00	10.62	12.65					
8	01/01/2013 05:00	10.96	12.86					
9	01/01/2013 06:00	11.62	13.07					
10	01/01/2013 07:00	12.78	13.48					
11	01/01/2013 08:00	13.09	13.74					
12	01/01/2013 09:00	13.09	13.69					
13	01/01/2013 10:00	12.5	14.04					
14	01/01/2013 11:00	12.09	13.75	12	14.4316	14.0624	20.5087	

Spreadsheets

- Store data
  - and logic for processing (kinda)
  - and results
  - built-in visualisations
  - …and all of these update automatically

- *Mostly* forwards compatible

- “Easier” to understand than more complex data structures



Screenshots of Microsoft Excel are used with permission from Microsoft

Reproducible environments

- The version of Python or R you're using makes a difference
- The versions of the packages you're using make a difference
- What if you share your code with *someone else** ?

* your future self is also *someone else*

- Python venv, conda environments, renv & more

```
numpy==1.20.3
pandas==1.2.4
xarray==0.18.2
iris==2.4.0
...
```

More info:
https://packaging.python.org/guides/installing-using-pip-and-virtual-environments/
https://docs.conda.io/projects/conda/en/latest/user-guide/concepts/environments.html
https://rstudio.github.io/renv/articles/renv.html

gates.xls [Compatibility Mode] - Microsoft Excel Viewer

	B	C	D	E	F	G	H	I	J	K	L	M
1	mass	time	limit 1	limit 2								
2	113	17,777	17,736	17,816	17,699	17,854	0	0	0,080			
3	131	19,122	19,096	19,176	19,050	19,194	0	0	0,080	M = ((T - 0,237)/1,65)^2		
4	149	20,378	20,328	20,430	20,310	20,445	0	0	0,102			
5	167	21,560	21,510	21,585	21,496	21,623	0	0	0,075	T=	20	
6	185	22,679	22,645	22,710	22,619	22,740	0	0	0,065	M=	**143,46**	
7	203	23,746	23,715	23,768	23,688	23,804	0	0	0,053			
8	221	24,766	24,722	24,792	24,710	24,821	0	0	0,070	M=	150	
9	239	25,745	25,696	25,784	25,692	25,799	0	0	0,088	T=	**20,445**	
10	257	26,689	26,672	26,728	26,637	26,740	0	0	0,056			
11	275	27,599	27,560	27,650	27,549	27,649	0	1	0,090			
12	293	28,480	28,450	28,530	28,432	28,529	0	1	0,080			
13	311	29,335	29,304	29,384	29,288	29,382	0	1	0,080			
14	329	30,165	30,136	30,216	30,120	30,211	0	1	0,080			
15	347	30,973	30,944	31,030	30,929	31,017	0	1	0,086			
16	365	31,760	31,725	31,824	31,717	31,803	0	1	0,099			
17	383	32,528	32,496	32,584	32,486	32,570	0	1	0,088			
18	401	33,278	33,248	33,328	33,237	33,319	0	1	0,080			
19	419	34,012	33,984	34,060	33,971	34,052	0	1	0,076			
20	437	34,729	34,700	34,792	34,690	34,769	0	1	0,092			
21	455	35,433	35,400	35,488	35,394	35,471	0	1	0,088			
22	473	36,122	36,080	36,168	36,084	36,160	1	1	0,088			
23	491	36,799	36,776	36,848	36,761	36,836	0	1	0,072			
24	509	37,463	37,430	37,488	37,426	37,499	0	0	0,058			
25	527	38,115	38,096	38,168	38,079	38,151	0	1	0,072			
26	545	38,757	38,728	38,800	38,721	38,792	0	1	0,072			
27	563	39,388	39,344	39,424	39,353	39,422	1	1	0,080			
28	581	40,009	39,984	40,040	39,974	40,043	0	0	0,056			
29	599	40,620	40,592	40,672	40,586	40,654	0	1	0,080			
30	617	41,222	41,176	41,264	41,189	41,255	1	1	0,088			

Hoja1 gates

Ready 100%

One-click options

Viewing

jupyter
nbviewer

GitHub

Demoing

binder

More info:
https://nbviewer.jupyter.org/
https://mybinder.org/

What about other software?

- Docker containers

```
# Start with a Python 3.9 environment
FROM python:3.9

# Install important package needed for analysis
RUN apt-get -y update
RUN apt-get -y install very-important-package

# Install other requirements
RUN pip install -r requirements.txt
```

- Virtual machines

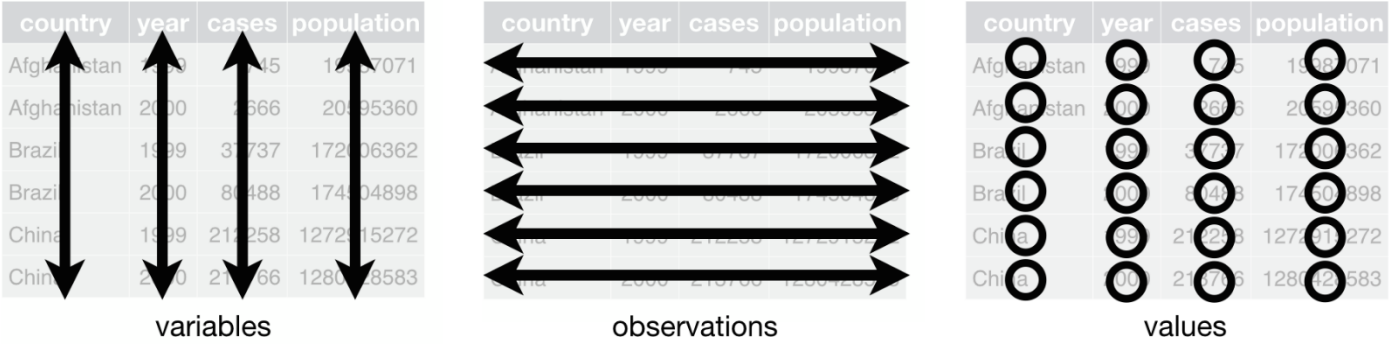
Mean value in m/s from 10 s readings

Interpolated data

Time	Wind Speed	Wind Direction	Temperature
	[m/s]	[Degrees]	[Deg C]
01/01/2001 00:00	12.56	188.7	1.53
01/01/2001 01:00	12.44	200.0	2.19
01/01/2001 02:00	12.05	202.0	2.78
01/01/2001 03:00	**102.26**	204.0	3.36
01/01/2001 04:00	**102.02**	206.0	3.73
01/01/2001 05:00	11.28	208.0	3.84
01/01/2001 06:00	10.46	106.5	3.69
01/01/2001 07:00	10.01	201.8	#N/A
01/01/2001 08:00	10.04	197.1	#N/A
01/01/2001 09:00	9.79	191.2	#N/A
01/01/2001 10:00	9.57	185.1	#N/A
01/01/2001 11:00	9.35	182.2	3.76

Tidy data and metadata

- Use ***tidy*** data



- Use a data dictionary
- Don’t alter data → save intermediate data that can be regenerated
- Don’t hard-code workarounds in code → use data validity indicators

More info:
https://the-turing-way.netlify.app/reproducible-research/rdm/rdm-spreadsheets.html
https://help.osf.io/hc/en-us/articles/360019739054-How-to-Make-a-Data-Dictionary

Data.xlsx

Data_Phase2.xlsx

Data_Combined.xlsx

Other_data.xlsx

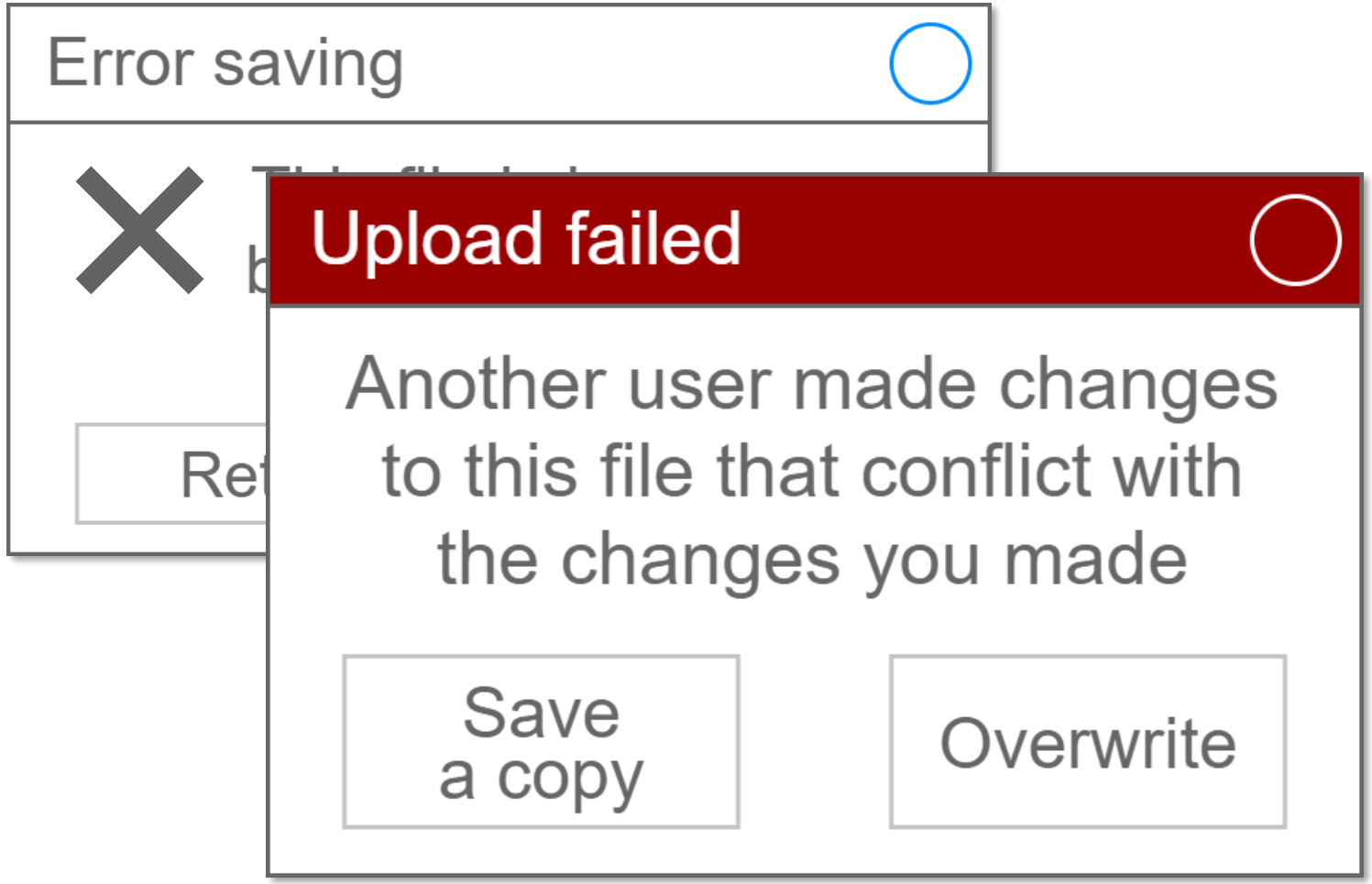
Data_Combined_Combined.xlsx

Data_Combined_Combined_Revised_v2_Final.xlsx

Keeping track

- **Version control** for code
  - Don’t need multiple copies
  - Free history

- **Pipelines** for data
  - make, Makefiles
  - (many others)


Error saving
Upload failed
Another user made changes to this file that conflict with the changes you made
Save a copy
Overwrite

Working with others

- Use collaborative tools – many are free!
  - Issues
  - Branches – pull requests
  - Automated actions (more on this later)
  - Project management

GitHub

GitLab

More info:
University of Bristol Advanced Computing Research Centre training
Introducing Version Control with Git course, also available as a YouTube video
Git for Collaboration course, also available as a YouTube video

	A	B	C
1	**Time**	**Temperature**	**Pressure**
2		[Deg C]	[barg]
3	01/01/2001 00:00	12.56	0.188
4	01/01/2001 01:00	12.44	0.200
5	01/01/2001 02:00	12.05	0.202
6	01/01/2001 03:00	12.26	0.204
7	01/01/2001 04:00	12.02	0.206
8	01/01/2001 05:00	11.28	0.208
9	01/01/2001 06:00	10.46	0.206
10	01/01/2001 07:00	10.01	0.201

```
=MAX($B3*(1–$C3)^$H$14, $H$15)
```

```
=MAX(Temperature
        *(1–Pressure)^Power_Law,
      Minimum_Value)
```

Newlines in a formula?!

```
=Calculate_special_metric(
      Temperature,
      Pressure)
```

Helping the reader* *(you)

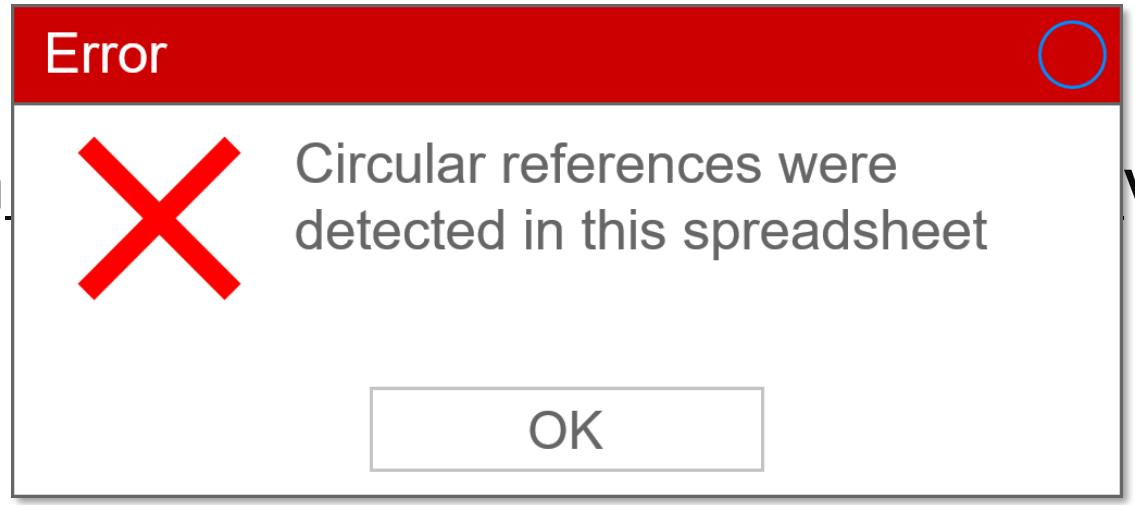
- Informative variable names

```
t = 28.3
temp = 28.3
temperature = 28.3
temperature_Kelvin = 28.3
```

- Encapsulating work inside functions
- Sticking to “business logic” in your main code

```
(
    load_data()
    .pipe(mask_invalid_data)
    .pipe(interpolate_missing_data)
    .pipe(regrid_data)
    .pipe(calculate_important_metric)
    .plot()
)
```

Data_ v2_Final.xlsx



Verifying outputs

- Testing frameworks
  - `assert` statement
  - Python unittest

- Automated testing

GitHub Actions

The most important thing is your system (your process)

- Documentation (not just comments)
- Docstrings in functions

SPHINX

MkDocs

- Worked examples and tutorials

jupyter

rmarkdown

Further resources

- The Turing Way – making collaborative, reusable and transparent research “too easy not to do”
  https://the-turing-way.netlify.app

- University of Bristol Advanced Computing Research Centre training
  https://www.bristol.ac.uk/acrc/acrc-training/
  - **Python**: Beginning Python; Intermediate Python; Data Analysis with Python
  - **R**: Beginning R; Intermediate R; Data Analysis with R
  - **Git/GitHub**: Introducing Version Control with Git; Git for Collaboration
  - …and more!

- Free talks, courses and workshops at JGI Data Week Online 14–18 June 2021
  https://www.bristol.ac.uk/golding/get-involved/data-week-online-2021/

- ECMWF Workshop: Building reproducible workflows for earth sciences
  https://www.ecmwf.int/en/learning/workshops/building-reproducible-workflows