data cleaning principles

Karl Broman

Biostatistics & Medical Informatics, UW–Madison

kbroman.org
github.com/kbroman
@kbroman@fosstodon.org
bit.ly/datacleaning2023

These are slides for a CIBM seminar at UW-Madison on 2023-03-21. They're slightly expanded from a talk I gave for the csv,conf,v6 (`https://csvconf.com/`), 2021-05-04.

Data analysts spend a lot of time organizing and cleaning data, but few of us have been trained to do so. Why is that?

Some say that data cleaning is difficult to generalize. But I think there are some general principles. Moreover, I think we have an important shared experience in data cleaning that we can commiserate about, and through which we can learn from each other.

Slides: `https://bit.ly/datacleaning2023`
with notes: `https://kbroman.org/Talk_DataCleaning2023/data_cleaning_notes.pdf`
source: `https://github.com/kbroman/Talk_DataCleaning2023`

Tidy data are all alike,
but every messy dataset
is messy in its own way.

– Hadley Wickham

r4ds.had.co.nz/tidy-data.html

Hadley's talking more about data organization than data cleanliness. And his point is that if you make data tidy, it simplifies all the downstream analyses.

But is every messy dataset uniquely messy?

For sure, many of my collaborators have shown impressive creativity in their approach to organizing and managing data. But we do see many of the same sorts of problems over and over.

If I clean up [Medicare] data ...
does any of the knowledge I gain ...
apply to the processing of RNA-seq data?

– Roger Peng

doi:10/jz69

In his discussion of David Donoho's paper about data science, Roger Peng wrote about how data cleaning is frustratingly difficult to generalize.

But my answer to his question is absolutely!

A person with experience cleaning one dataset has important experience to draw upon when moving to another dataset, even if it's of a totally different nature.

Data Mishaps Night

Join us for the first inaugural Data Mishaps Night! We will feature a lineup of data mistake stories with a focus on the human aspect of data work and lessons learned the hard way.

Caitlin Hudon & Laura Ellis

dataMishapsNight.com

4

In February, 2021, Caitlin Hudon and Laura Ellis organized a Friday evening conference where 16 people gave short presentations on data mishaps.

Many of the stories concerned mistakes in data cleaning, and these seemed to bring out a strong sense of shared experience. We have suffered and struggled through very similar data problems.

Data cleaning

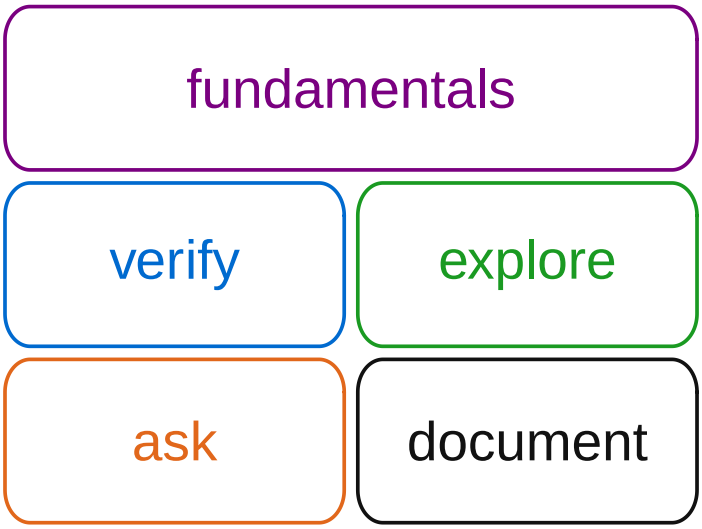
- tedious
- embarrassing
- needs context
- doesn't feel like progress
- requires creativity
- requires coding prowess
- source of many problems

Really, I think we don't usually teach data cleaning because it's something we prefer to keep private.

We're shy about it.

And data cleaning code is our ugliest code.

Data cleaning principles



I'm proposing a set of basic principles for data cleaning, and splitting them into five groups. There are some fundamental principles, followed by four basic ideas: verify things that you expect, explore to find further oddities, ask questions, and document what you've done.

fundamentals

1. Don't clean data when you're tired or hungry.

(paraphrasing Ghazal Gulati)

At her talk at the Data Mishaps night, Ghazal Gulati emphasized this point, of not cleaning data when you're tired or hungry.

Data cleaning requires considerable concentration, and you need to allow sufficient time to do the work. If you're in a hurry, you'll miss things.

fundamentals

2. Don't trust anyone (even yourself)

"my motto is 'trust no one'
...except maybe @kwbroman?"

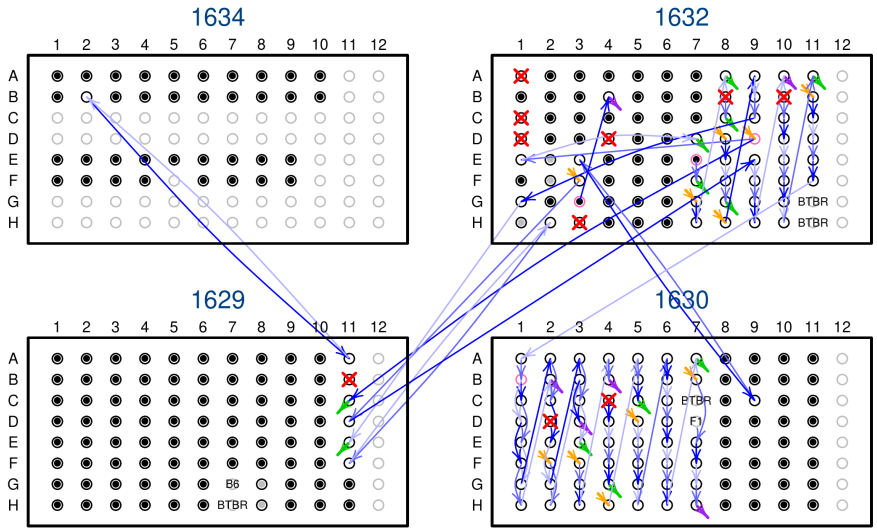
– Jenny Bryan

Next: don't trust anyone. Even if the initial data cleaning was done by someone you respect, you should double-check things that they may have missed. And data cleaning is an ongoing process.

Jenny Bryan's tweet is among the nicest things anyone has said about me.

fundamentals

3. Think about what might have gone wrong and how it might be revealed



doi:10/gpfzs8

Personally, I think this is the most important principle for data cleaning. It has been central in guiding my approach.

The figure here is an illustration of the most startlingly result I've had in data cleaning: a genetics project where almost 20% of the DNA samples had been mislabelled. The samples were arranged in wells in 8×12 plates; four of the six plates are shown here. The dots indicate the correct DNA was placed in the well, but the arrows point from where a sample should have been to where it actually was placed.

I ultimately came to this finding by thinking about what might have gone wrong in the project, checking for particularly problems, and then following the trail of evidence to this mess.

fundamentals

4. Use care in merging

	A	B	C	D	E	F	G
1	id	glucose.0	glucose.5	glucose.15	glucose.30	insulin.0	insulin.5
2	DO−221	145.742786	206.452638	216.640608	299.55501	0.74455	2.0264
3	DO−222						
4	DO−223						
5	DO−224						
6	DO−225						
7	DO−226						
8	DO−227						
9	DO−228						
10	DO−229						
11	DO−230						

	A	B	C	D	E	F	G
1	id	glucose.0	insulin.0	glucose.5	insulin.5	glucose.15	insulin.15
2	DO−321	66.839405	0.04	246.685995	0.04	305.26214	0.04
3	DO−322	98.12509	0.51185	246.25574	1.4062	301.8201	2.828
4	DO−323	94.68305	1.7812	448.1068	1.0248	521.61894	1.02725
5	DO−324	121.051535	0.0882	407.355505	0.63475	470.541525	0.8195
6	DO−325	122.95695	0.19155	298.193665	0.6467	323.148455	0.40515
7	DO−326	201.447755	0.7454	386.51887	0.6081	654.99799	1.07225
8	DO−327	130.025425	0.0509	477.302675	0.166	610.49733	0.4842
9	DO−328	143.60919	0.23435	438.88705	0.70505	406.249135	0.2498
10	DO−329	125.29262	0.04	543.74634	1.7366	520.205245	0.8498
11	DO−330	135.61874	0.91275	393.03416	3.73095	454.62209	1.7325

Many problems arise due to mistakes when merging data from multiple files. A common problem is a change in the data arrangement, such as in the order of columns.

Focus on the labels (which are more likely correct), rather than the position of variables in a file (which are more likely to change).

fundamentals

5. Dates & categories suck

The fifth fundamental principle is that dates and categories suck. You'll expend an inordinate amount of time dealing with these: typos in category labels, different date formats, people who died a decade before they were born or lived to be 150.

Just be glad if you're not dealing with time zones.

But you may be asking yourself, "How is this a principle."

Principle:

a fundamental truth that guides our thinking

Yeah, are these principles? I was thinking the same thing. Was I drifting away from principles and more to just stuff to know or do?

This seems a pretty good definition, and is sufficiently broad to cover what I'm proposing, for the most part.

fundamentals

5. Dates & categories suck

So yeah, this counts as a principle.

Much of your pain will come from the dates and categorical data; you should be ready for that.

verify

6. Check that distinct things are distinct

	A	B	C	D	E	F	G
1	WiscID	ID	NEOID	Fem_CA	Fem_Imax	Fem_Imin	Fem_J
2	F2.C1W.F.1248	1248	NEO183	0.7524	0.1427	0.1006	0.2433
3	F2.C1W.M.1250	1250	NEO184	0.7669	0.1556	0.09652	0.2521
4	F2.C1W.F.1251	1251	NEO185	0.7613	0.1549	0.09659	0.2515
5	F2.C1W.F.1254	1254	NEO186	0.7475	0.1503	0.08603	0.2363
6	F2.C1W.M.1257	1257	NEO187	0.8197	0.1849	0.1056	0.2905
7	F2.___.F.715	715	NEO764	0.6017	0.09662	0.05969	0.1563
8	F2.___.F.751	751	NEO765	0.7273	0.1304	0.08735	0.2178
9	F2.___.F.1251	1251	NEO766	0.6675	0.1157	0.07814	0.1938
10	F2.___.M.1340	1340	NEO768	0.6656	0.1387	0.08122	0.2199
11	F2.C1W.M.739	739	NEO779	0.9336	0.2828	0.1628	0.4456

The next major section of principles concern efforts to verify all of the things that should be true for your data.

First up: are the variables that are supposed to have distinct values really showing distinct values? Here, there are a pair of individual identifiers that are duplicated.

verify

7. Check that matching things match

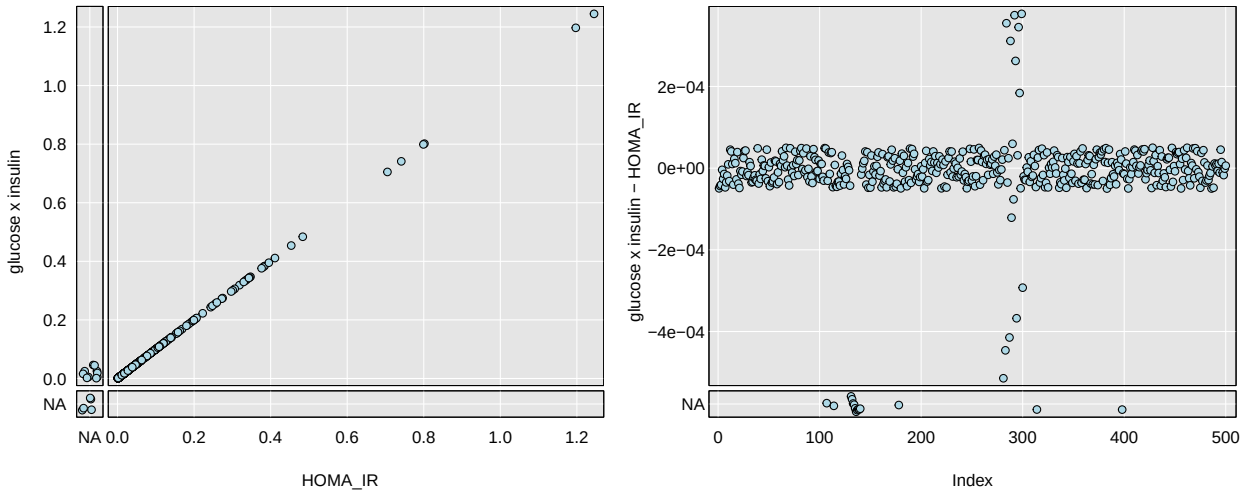
	A	B	C	D
1	id	sex	n_gen	age_days
2	F20.25	M	20	75
3	F21.30	M	21	75
4	F21.68	M	21	71
5	F22.52	M	22	73
6	F21.71	F	22	63
7	F22.116	F	22	57
8	F21.F20.9.M5	M	20	82
9	F21.F20.18.M5	M	20	77
10	F20.26	M	20	75
11	F21.62	M	21	72

	A	B	C	D
1	id	sex	age_at_dosing	n_gen
2	F22.69	F	67	22
3	F22.106	F	69	22
4	F22.70	F	67	22
5	F22.107	F	69	22
6	F21.71	F	65	21
7	F22.116	F	62	22
8	F22.73	F	65	22
9	F22.117	F	62	22
10	F21.108	F	62	21
11	F22.118	F	59	22

Next, are the things that are supposed to match actually matching? This often concerns data that are repeated in multiple data files. For example, you might have body weight included multiple times; are individuals' weights the same in all files? Here, there's a column for "number of generations" that shows an inconsistency between two files.

verify

8. Check calculations



Any time there is a calculation, you should verify the values. This is useful both for ensuring that you understand the calculation, and you also have the chance to reduce round-off error (though that seldom matters). Occasionally you may see mistakes.

Here, I'm plotting HOMA-IR which is the product of blood glucose and blood insulin; the values in the data on the x-axis and the values I calculated on the y-axis. It's often better to look at the differences directly; in the right panel I plot the differences on the y-axis. It appears that there is a batch of individuals that whose values were rounded more coarsely.

Note that rather than omitting missing values, I've pulled them out and plotted them in the margins. Showing the missing values can be really important for identifying problems. Here there are some HOMA-IR values that are missing but maybe shouldn't be.

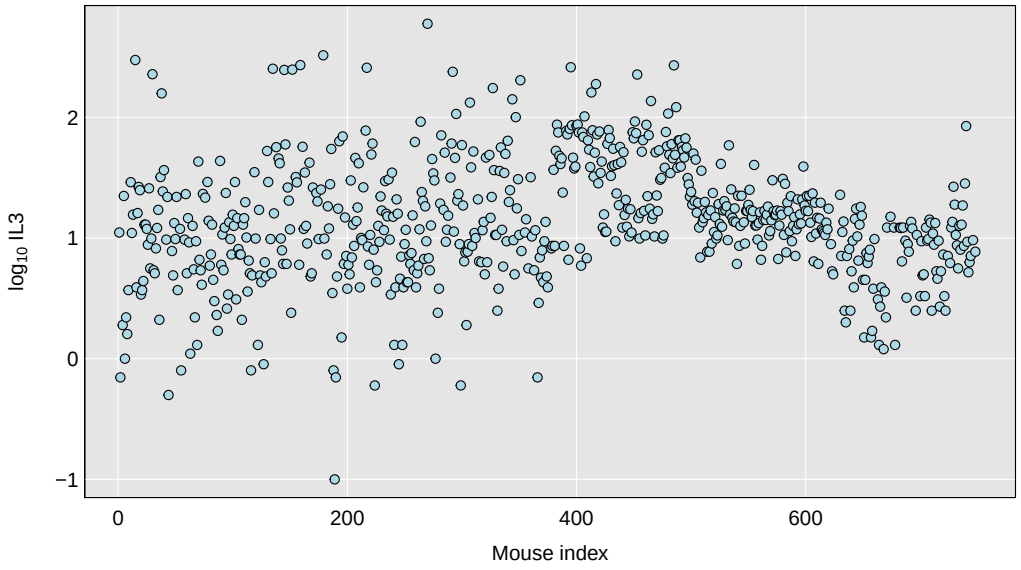
verify

9. Look for other instances of a problem

As with software testing, any time you find a problem, be sure to look for other instances of that problem.

explore

10. Make lots of plots



A particularly important aspect of data cleaning concerns just exploring the data to identify potential problems. And a particularly important aspect of that exploration is to just make lots of plots.

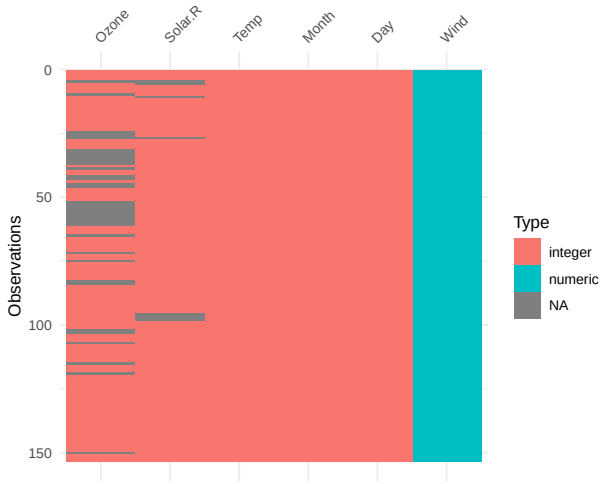
Plot variables over time or by subject ID, which may indicate things like batch effects.

Make scatterplots of variables against each other, looking particularly for outliers. Outliers could be real biological variation, but they could also be data entry problems, like a pair of numbers being swapped, or the weights entered in grams rather than milligrams.

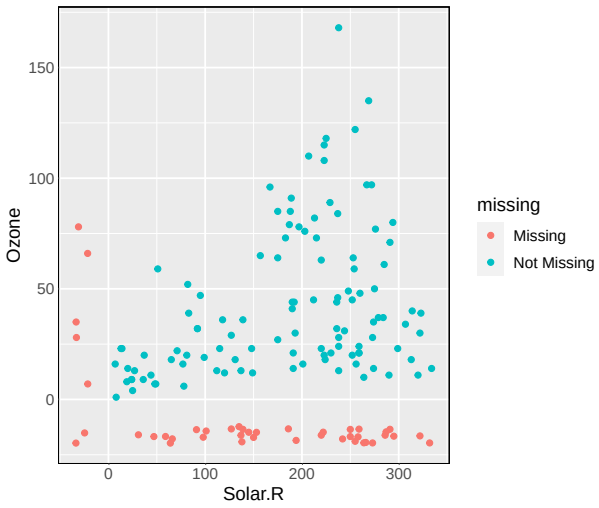
explore

11. Look at missing value patterns

{visdat}



{naniar}



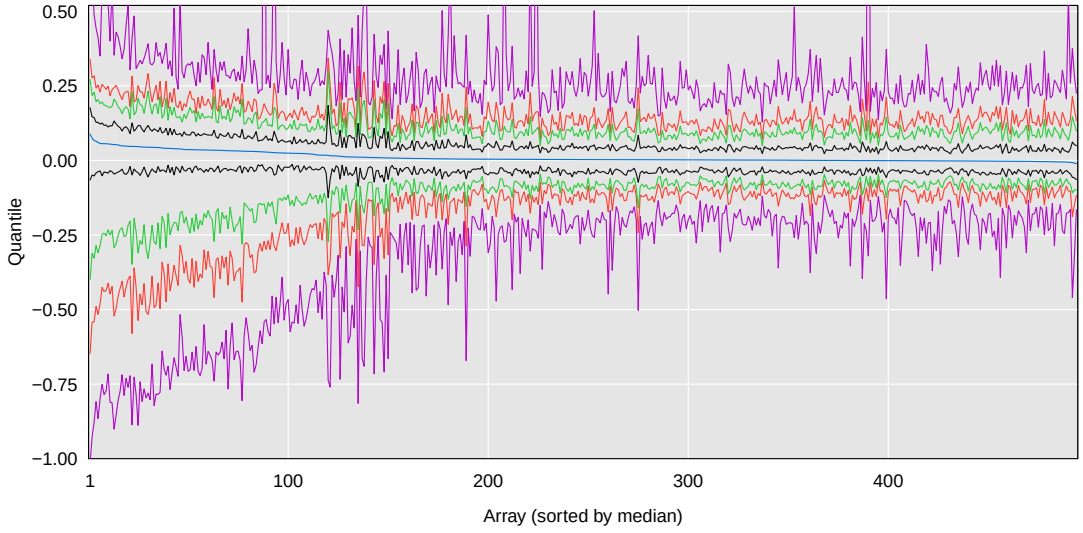
The pattern of missing data can be particularly informative about problems. Two particularly useful tools are the R packages visdat and naniar.

In the left panel, visdat (https://docs.ropensci.org/visdat/) provides a heatmap indicating which data points are missing, and also the variable types.

In the right panel, naniar (https://naniar.njtierney.com/) provides a scatterplot that include the cases that are missing one or both variables.

explore

12. With massive data, make more plots not fewer



With large-scale datasets, it can be hard to make the sort of exploratory plots that you'd typically make. With oodles of data, you'd think you'd be looking at oodles of plots, but there's a tendency to give up and not look at any.

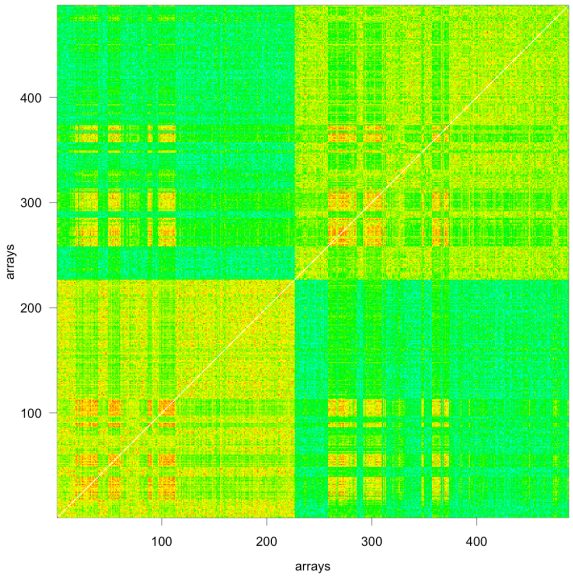
It's hard to look at 500 histograms, but it can be done. Superimpose a bunch of density estimates, maybe highlighting some portion of them. You can also pull out a couple of summary statistics, such as the median and inter-quartile range.

Or here I'm looking at the equivalent of 500 boxplots. I sorted a set of gene expression microarrays by their median, and then plotted the median in blue, the 25th and 75th percentile in black, the 10th and 90th in green, 5th and 95th in red, and 1st and 99th in purple.

With these data, it became apparent that there were 120 badly behaved arrays, with median shifted to the right and with a long left tail.

explore

13. Follow up all artifacts



kbroman.org/blog/2012/04/25/microarrays-suck

Wow the clash of those colors is particularly bad.

This is a heat map of the correlation matrix for a set of gene expression microarrays. The plaid pattern was a shock to me, and was caused by a set of bad arrays that we hadn't noticed previously.

My point here is simply to follow up all artifacts.

If you see something weird, follow through and try to figure out the underlying cause. If could be an error, or a set of bad assays, or it could be the most interesting finding in the study.

ask

14. Ask questions

When were the data gathered?

How, and by whom?

Was the data gathered in batches?

How were the data files created?

Was any calibration or normalization done?

The next section concerns asking questions and for additional information.

Above all, don't be shy about asking questions.

The answers can give important clues about potential problems or things to look for.

ask

15. Ask for the primary data

	A	B	C	D	E	F	G	H
1	MouseNum	Wean Date	Assay Date	Weight	Glucose	Insulin	Trigly	HOMA
2	Mouse3001	6/22/2005	8/18/2005	47.3	617	11.7	175.1	321.9
3	Mouse3002	6/22/2005	8/18/2005	51	256.5	50.6	97.6	576.6
4	Mouse3003	6/22/2005	8/18/2005	50.6	274.9	52.5	160.5	641.3
5	Mouse3004	6/22/2005	8/18/2005	46	615.1	9	238.7	246
6	Mouse3005	6/30/2005	b	42.4	NA	NA	102.3	587.1
7	Mouse3006	6/30/2005	b	39.7	NA	NA	209.4	338.7
8	Mouse3007	6/30/2005	b	36.9	NA	NA	69.8	140.6
9	Mouse3008	6/30/2005	b	50	195.5	45.3	142.1	393.4
10	Mouse3009	6/30/2005	b	40.1	569.4	12.4	411	312.9
11	Mouse3010	6/30/2005	b	40.7	593.8	15.6	333.6	411.8

Always ask for the primary data files, even if the present data files are sufficient.

My favorite example is for a case where a derived column was available but the original columns were not. And actually I had a case like that where just the rounding of the columns was patchy and odd, and I wanted to get access to the original version.

It turned out that the primary data file was a large excel file with one sheet per subject, and the file I was given was produced by complicated and time-consuming copy-paste by hand. Once I obtained the original file, I saved someone a ton of unnecessary and error-prone work.

ask

16. Ask for metadata

"What the heck is '`FAD_NAD SI 8.3_3.3G`'?"

If you don't understand what the variables mean, then you're just analyzing a random pile of numbers rather than really connecting with the data.

This is my favorite example of this; it's hard to connect with a variable name like this.

So always ask for meta data.

ask

16. Ask for metadata

	A	B	C	D
1	name	plot_name	group	description
2	mouse	Mouse	demographic	Animal identifier
3	sex	Sex	demographic	Male (M) or Female (F)
4	sac_date	Date of sac	demographic	Date mouse was sacrificed
5	partial_inflation	Partial inflation	clinical	Indicates if mouse showed partial pancreatic inflation
6	coat_color	Coat color	demographic	Coat color, by visual inspection
7	crumblers	Crumblers	clinical	Indicates if mouse stored food in their bedding
8	diet_days	Days on diet	clinical	Number of days on high−fat diet

And meta data are data, so it's best to have them formulated as a proper data file. Have a data dictionary in a spreadsheet that you can compute on, rather than as an unformatted glossary in a Word document.

ask

17. Ask why data are missing

Assay failed?

Below detection limit?

Viewed as outliers?

Subjects dropped out?

Missing data are often informative or require special consideration, so be sure to always ask about them.

When considering how to handle missing data, it is critical to first understand why they data are missing. (Sometimes a key question is whether there is information in the missingness.)

And sometimes they are not intentionally missing. Subjects missing from one file but present in others could indicate that a file was truncated. Be sure to ask!

document

18. Create checklists & pipelines

- ☐ Percent missing genotypes
- ☐ Sample duplicates
- ☐ Sex and X/Y genotypes
- ☐ Heterozygosity
- ☐ Genotype frequencies
- ☐ Crossover counts
- ☐ Genotyping error rates

The final section of principles concerns documentation.

Create checklists and pipelines for yourself and others, so that when you return to similar data, you will remember many of the things to check, and you can build on what you've learned.

document

19. Document not just what but why

Gough project diagnostics

Karl Broman, 3 March 2014

Combine genotypes and phenotypes

I've combined the initial genotypes (using the re-clustered genotypes for plates 14-16) with the well-behaved portion of the re-run genotypes. I'm focusing on 36813 markers that are informative (though, as we'll see, there are still a lot of badly behaved and basically non-informative markers that need to be removed). I've combined data on replicate samples, to give one set of genotype calls for each sample.

There are 1497 genotyped mice and 1464 phenotyped mice. All of the mice in the phenotype data have genotypes, but there are 33 genotyped mice with no phenotypes, including 3 Gough mice and 30 F2 progeny.

For this sort of work, we need things to be more than reproducible; you will need to capture not just what you did, but also why. For example, if you decide to omit some subset of samples, will you remember 2 years from now just why you chose to omit them?

A script that automates the process is great, but even better is a reproducible report, like a Jupyter notebook or RMarkdown document, which automates and documents and also captures your reasoning.

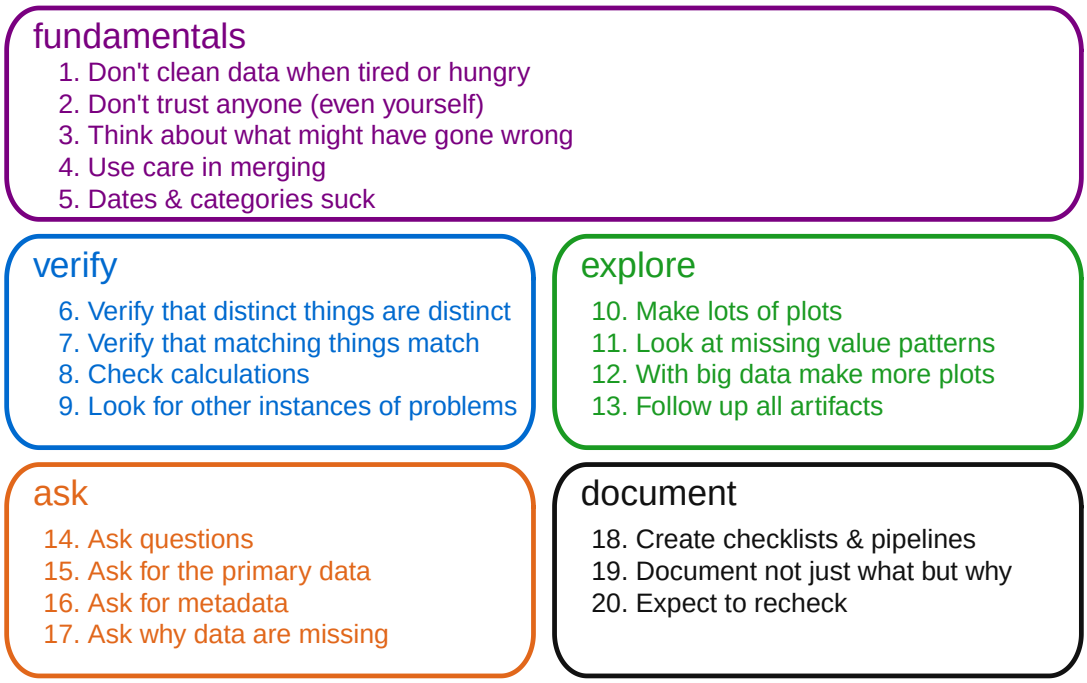
Another advantage of cleaning data within reproducible reports is that of communicating the process and results to collaborators. This is the top of one such document.

document

20. Expect to recheck

Finally, data cleaning is not a single step in the analysis chain; rather, it is an ongoing process that you will need to continually revisit as you delve deeper into the data. Keep an eye out for hints of problems, and arrange your work with the expectation that you'll need to re-run everything at some point, and maybe a number of times.

Data cleaning principles



In summary, data cleaning is not an activity that needs to be constructed from scratch in each instance for each dataset. There are a number of principles that can guide our approach to cleaning data.

There may actually be more commonalities in our data cleaning experiences and methods than in the following stages of our data analysis work.

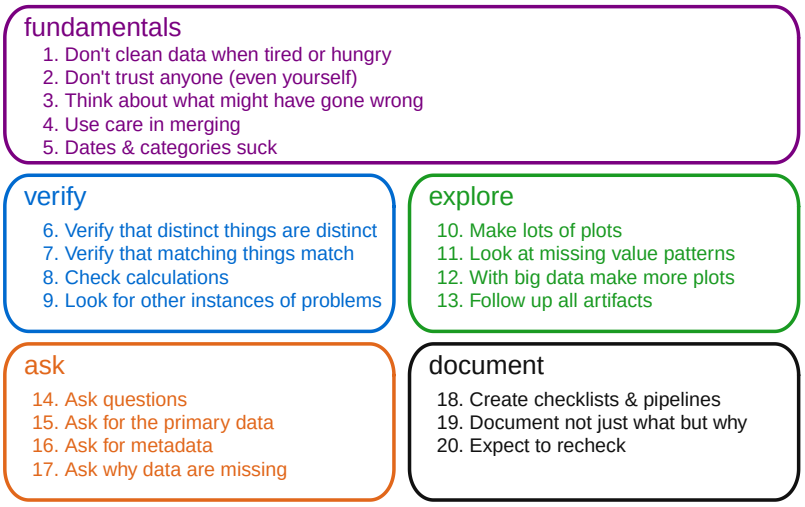
I will let the data speak for itself
when it cleans itself.

– Allison Reichel

I love this.

The proportion of our time spent cleaning data is likely to increase. It's never going to clean itself.

Slides: bit.ly/datacleaning2023
kbroman.org
github.com/kbroman
@kbroman@fosstodon.org



Here's where you can find me and these slides.