Ferret: Reviewing Tabular Dataset for Manipulations

Supplementary Material

Devin Lange, Shaurya Sahai, Jeff M. Phillips, Alexander Lex

Case Study: DS-Driving

Retraction: https://doi.org/10.1073/pnas.2115397118
Blog: http://datacolada.org/98

This psychology study claims that signing an honesty pledge at the top of a document leads to more honest reporting than at the bottom. This dataset is from an experiment that asked participants to report the odometer mileage of their car both before and after some period of time.

The dataset is sorted into rows. Each row corresponds to an insurance policy number. Each policy can have between 1 and 4 cars on it. As a result, there are four sets of before/after odometer columns.

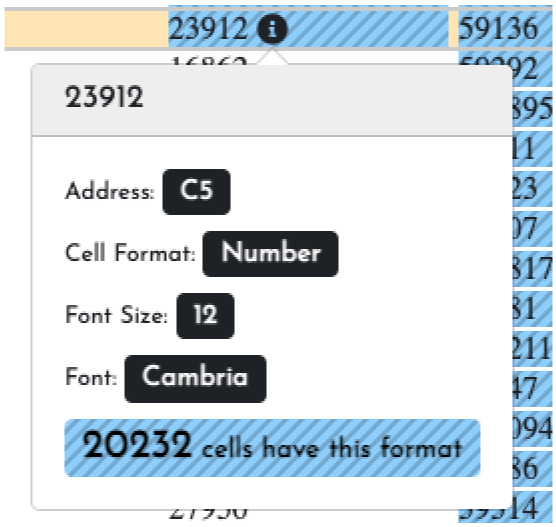
Half of the **rows** appear to be generated by adding a small amount of noise to the original values. In addition, “after” **columns** appear to be generated by adding a random number between 0 and 50,000 to the “before” number.

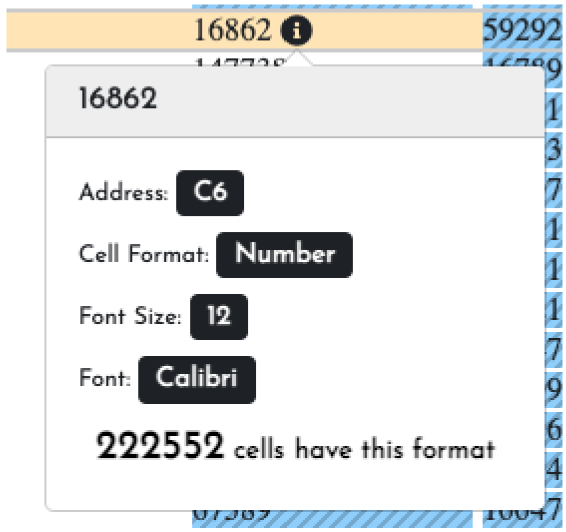
Formatting artifacts indicate fabricated rows

On initial load, it is evident that there is some different styling in the odometer reading for the first two cars.

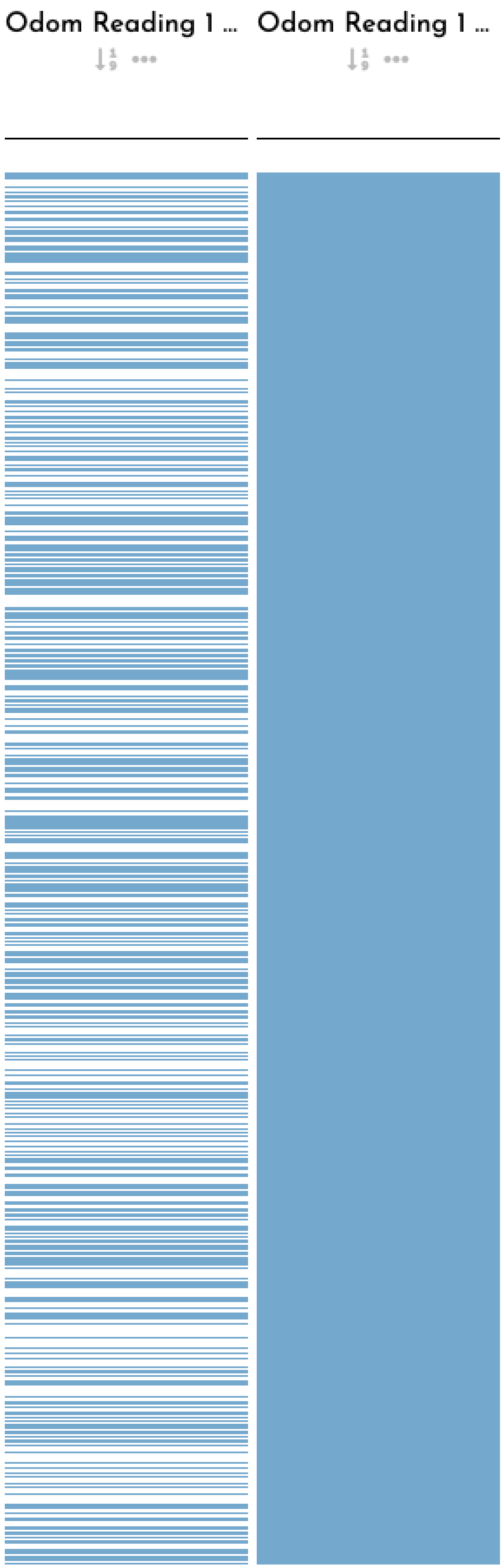
OMR Version	Policy # (masked)	Odom Reading 1 ...	Odom Reading 1 ...	Odom Reading 2...	Odom Reading 2...
Sign Top	1	896	39198		
Sign Bottom	2	21396	63511	32659	47605
Sign Bottom	3	21340	37460	44998	59002
Sign Bottom	4	23912	59136		
Sign Bottom	5	16862	59292		
Sign Top	6	147738	167895	125820	164688
Sign Bottom	7	18780	49811	45402	54824
Sign Top	8	41930	80323	181416	229852
Sign Top	9	28993	63707	13291	28165
Sign Bottom	10	78382	127817		
Sign Top	11	58500	81081		
Sign Bottom	12	99417	149211	48770	95179
Sign Bottom	13	93231	98047		
Sign Bottom	14	83443	105094		
Sign Bottom	15	22008	26486		
Sign Bottom	16	27950	59514	95883	126309
Sign Bottom	17	67589	100475	27617	74443
Sign Bottom	18	32753	76724		
Sign Top	19	33044	70775		
Sign Bottom	20	104857	109961	19548	47796
Sign Bottom	21	121699	137849		
Sign Top	22	16094	45489	159167	200316
Sign Top	23	78182	122739	21730	37863
Sign Bottom	24	21735	58693	39666	77258
Sign Top	25	47473	68971	4502	47293
Sign Bottom	26	121416	123661	53987	86852
Sign Bottom	27	4616	30597		
Sign Bottom	28	13604	31750		
Sign Bottom	29	125000	139801		
Sign Top	30	40463	70095		
Sign Top	31	34823	78262		
Sign Top	32	120296	137162	138617	169196
Sign Top	33	79173	125933		
Sign Bottom	34	110372	142100		
Sign Bottom	35	3586	41419		
Sign Top	36	26017	60347	12996	17179
Sign Top	37	149000	177873		
Sign Top	38	15939	17025		

On closer inspection, the difference is due to a difference in font between Cambria (Blue), and Calibri (White, no highlight).

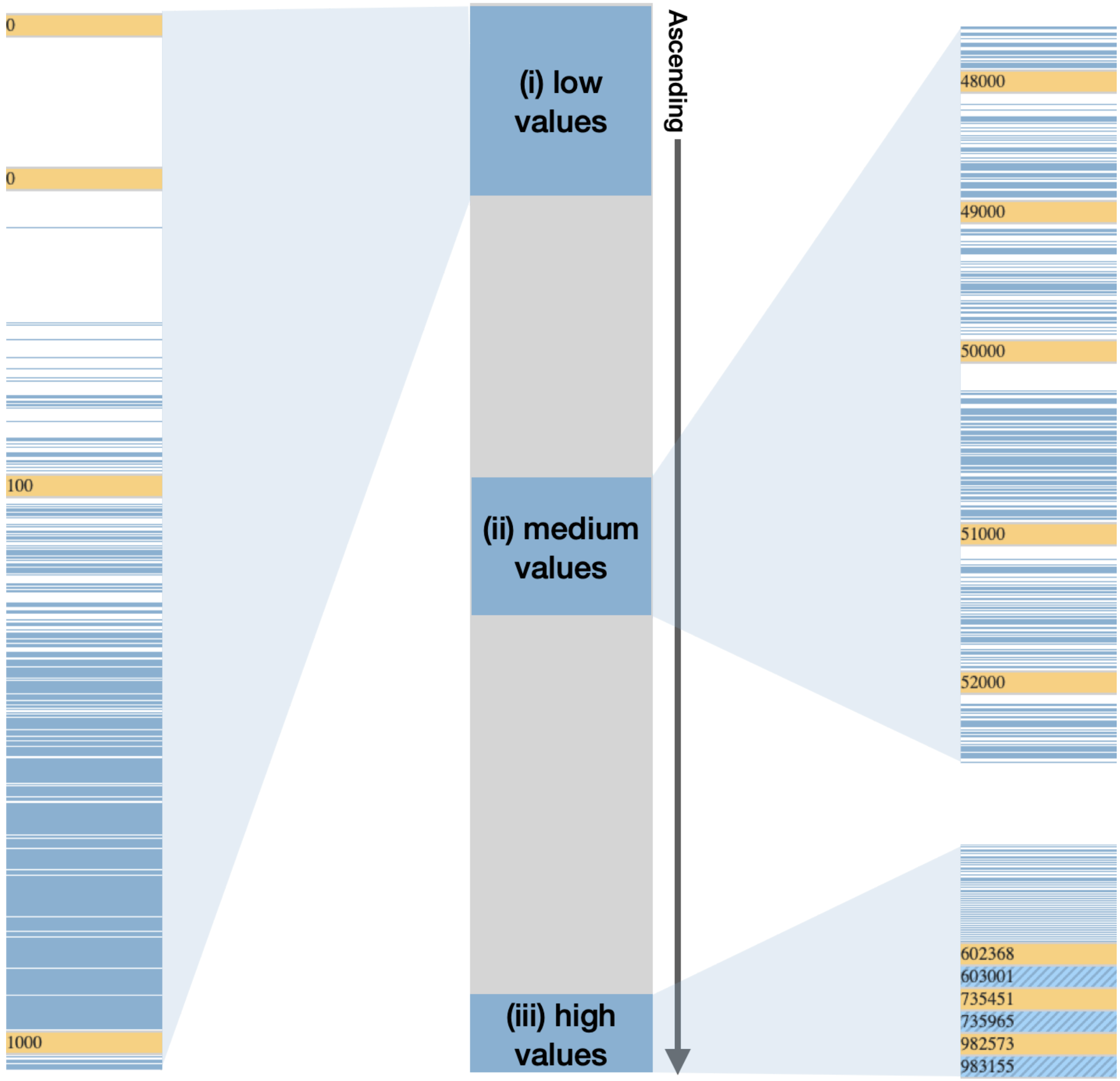




After switching to the Structural Overview, the pattern appears to continue through the entire 13,488 rows of the table:


Odom Reading 1 ...
Odom Reading 1 ...

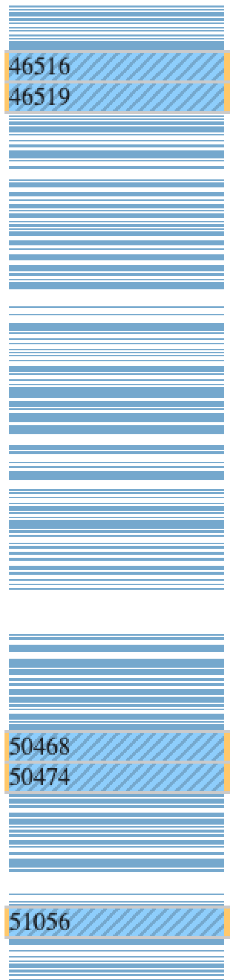
However, sorting by the odometer 1 reading results in several interesting patterns:



The values under 100 are almost entirely **Calibri font**. The values between 100 and 1000 are predominantly in Cambria.

Throughout the column, there are regular chunks of **Calibri** only font. These appear around round numbers. Cambria does not have the same large repeated regions around these large numbers. Larger runs of Cambria appear to be spurious (see below), and do not contain repeated numbers.

For the high values of the column you see rows altering back and forth between **Calibri** and Cambria.



Expanding the rows provides more detail. On closer inspection, it appears that every **Calibri** value has a Cambria equivalent that is within 1000 values.

Odom Reading 1 ...	Odom Reading 2...	Odom Reading 3...	Odom Reading 4...
299987			
300000		6000	
300525		6987	
304000			
304448			
309142	900	66000	
309564	1487	66917	
313668	5270	85217	
314174	6219	86073	
318102			
318458			
327774	55914		
328396	56341		
328549			
328981			
340232			
340638			
343000			
343935			
348285			
348702			
358236	111823	40000	
358544	112660	40845	
359641			
359700			
364774	112123	48472	
365387	112247	49086	
394482			
395272			
402847			
403733			
409515	31134	95000	
409663	31578	95013	
416537	48813	118579	
417041	48826	119477	
443920			
444290			
463090			
463284			
602368	152327	130210	152600
603001	153284	130947	153254
735451	99735	163390	
735965	100512	163756	
982573			
983155			

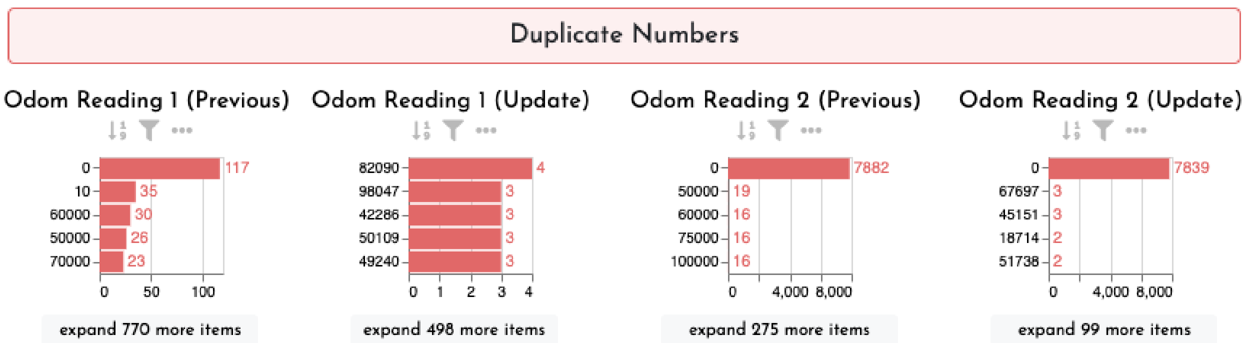
This is especially strange if you sort by the number of cars in the policy so you can easily see all of the policies with four cars. For each row with four cars in **Calibri,** there is an equivalent in Cambria where the odometer reading is within 1000 miles for each car.

Odom Reading 1 ...	Odom Reading 2...	Odom Reading 3...	Odom Reading 4...
735965	100512	163756	
0	120000	125000	146000
13	130240	37910	80791
845	131045	38591	80980
935	120126	125099	146367
1053	134778	175000	132000
1195	135427	175847	132596
8907	104849	35094	91640
9058	105406	35642	92607
10111	145650	176230	147569
10991	145902	176424	148268
11652	71000	13938	17911
12633	71384	13946	18711
14437	13640	17879	33864
14846	13821	18864	33985
17330	106000	43218	104591
18235	106253	43457	104916
18904	13024	103791	96954
19827	13425	103939	97538
34114	64000	34000	98885
34840	64523	34667	99180
41279	73641	45283	112415
41588	73757	46236	112457
47600	6500	15000	39000
47951	6901	15105	39364
49675	17709	27357	64428
50350	18421	27714	64784
51016	244058	120336	172906
51046	244307	120958	173372
57000	123663	16000	90000
57640	123666	16469	90026
58826	244390	122407	176373
59132	127063	26508	105626
59535	245203	123282	176947
59977	127872	27090	106443
89027	30	169777	143537
89625	1006	170410	143617
90367	14781	170958	147750
91079	15425	171105	147822
128392	124477	87000	14255
128516	124659	87127	14862
128628	132997	88688	145681
129585	133119	89193	146241
602368	152327	130210	152600
603001	153284	130947	153254

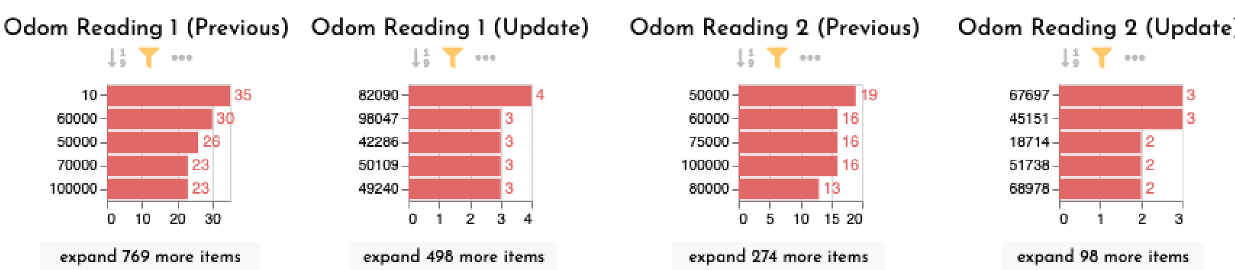
These attributes indicate that rows were copied to a temporary worksheet, then increased by a random noise function between zero and 1000, and added back into the original worksheet.

Numerical artifacts and deviations from domain expectations indicate fabricated columns

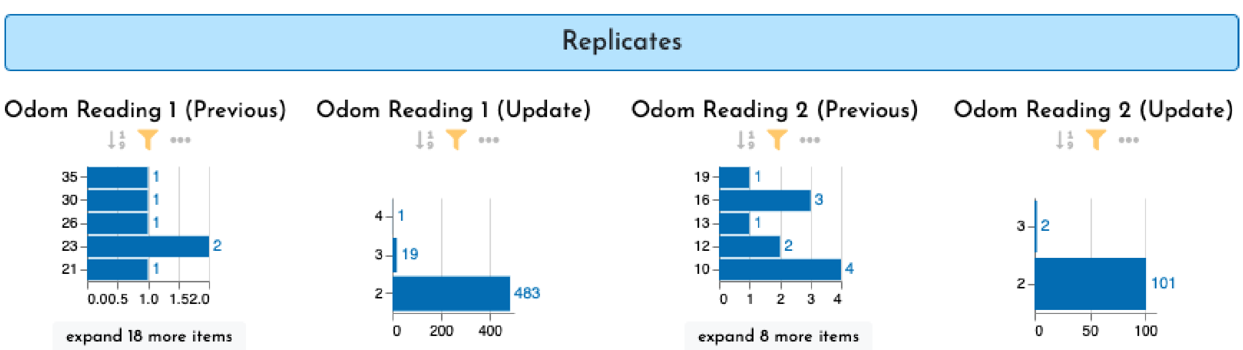
Moving beyond the formatting to the numerical artifacts. There appears to be a difference in how many duplicates are in a column in the **Previous** column compared to the **Update** column. The previous column contains many duplicates of round numbers such as 50000.



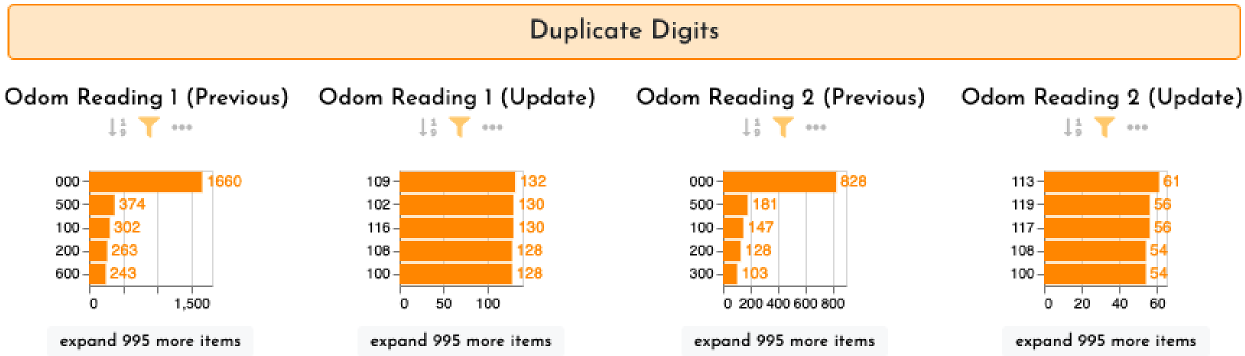
Since some blank fields are appearing as zero here, we can choose to ignore zero from our analysis to make the relevant data more clear.



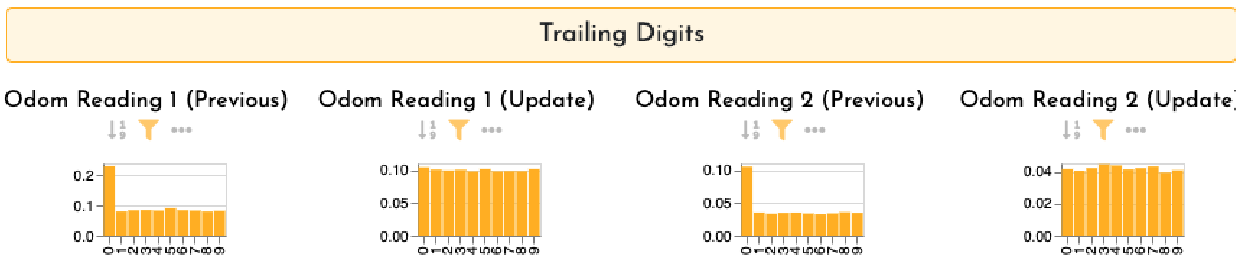
In addition, to which numbers are duplicated a lot, we can see how many times a number has been duplicated in the replicates chart.



In addition, to duplicate numbers, looking at duplicate sequences of digits again reveals that the **Previous** column includes the digits “000” much more frequently than the **Update** column.

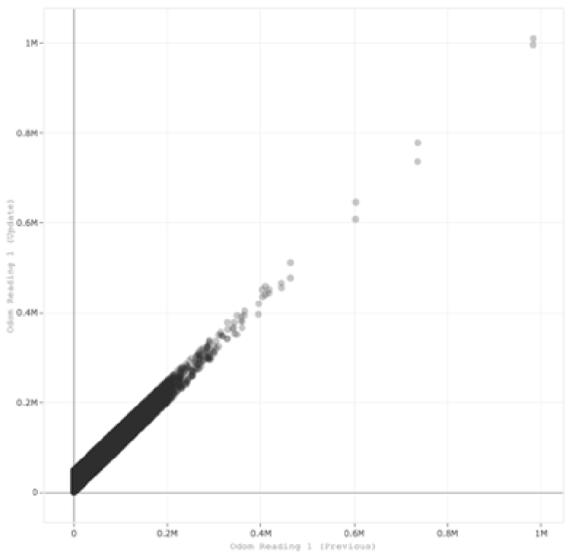


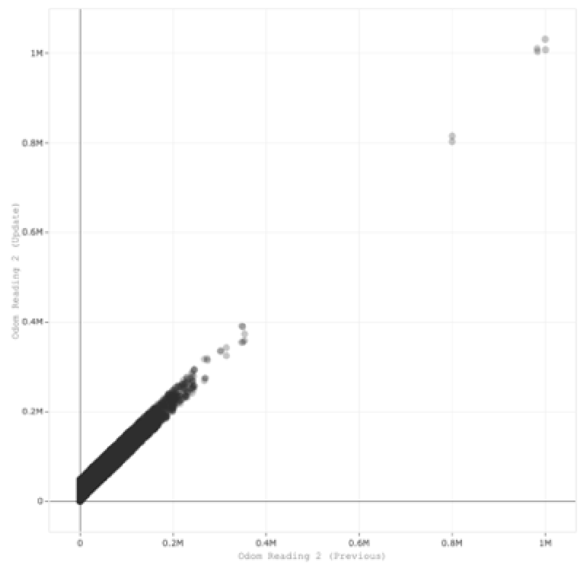
Finally, if we look at just the final trailing digit of each number, you can again see the much higher frequency of zero in the **Previous** column compared to the **Update** column.

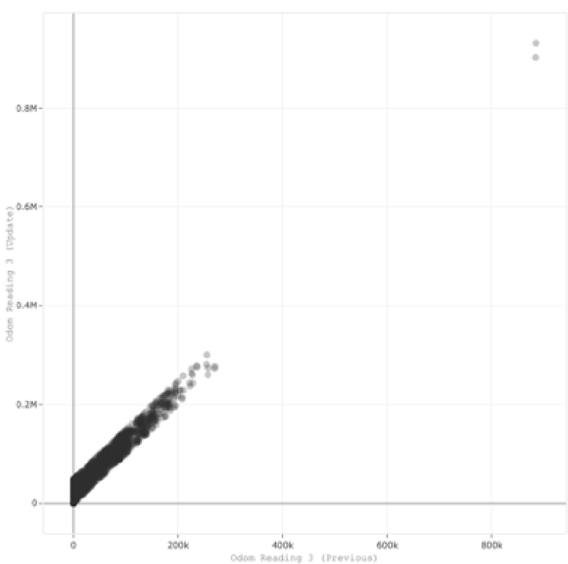


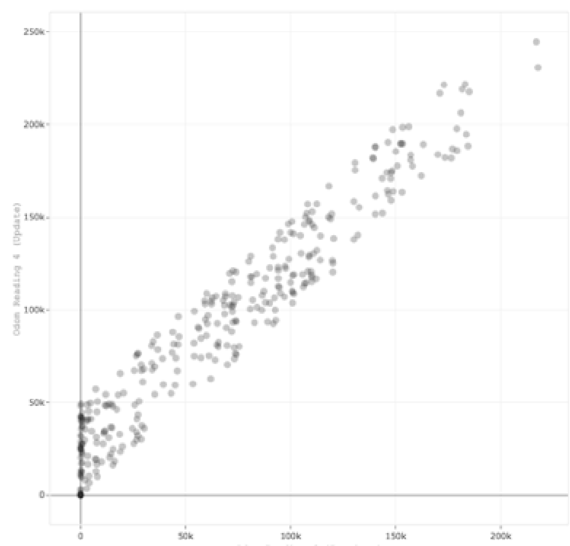
With all of these charts, it is clear that there is a rounding effect present in the **Previous** column but not in the **Update** column. This may lead you to question the relationship between the Previous and Update column.

With the general visualization analysis tool in Ferret, it is easy to plot scatter plots of these two columns for the four different pairs of columns.









These plots reveal a strange correlation between the Previous and Update columns. That is, the miles driven falls below 50,000 miles.

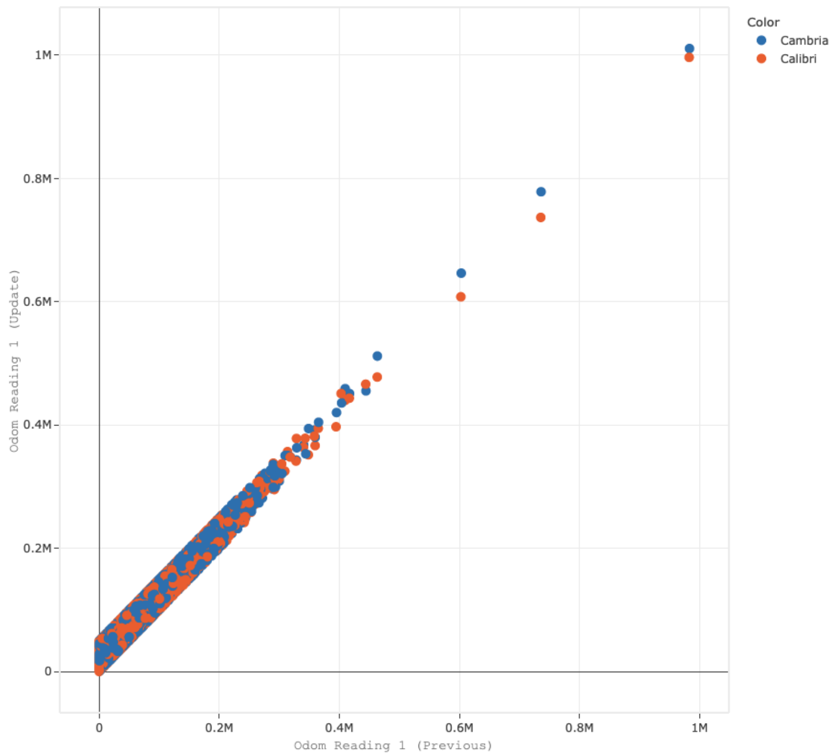
It is believed that the Update column was generated by adding a random number between zero and 50,000 miles.

Interestingly, this plot also hints at the duplicated rows. Examining the tails closely will reveal that points are always grouped in pairs. There is always at least one point within 1000 miles in the x-axis for any specific x value.

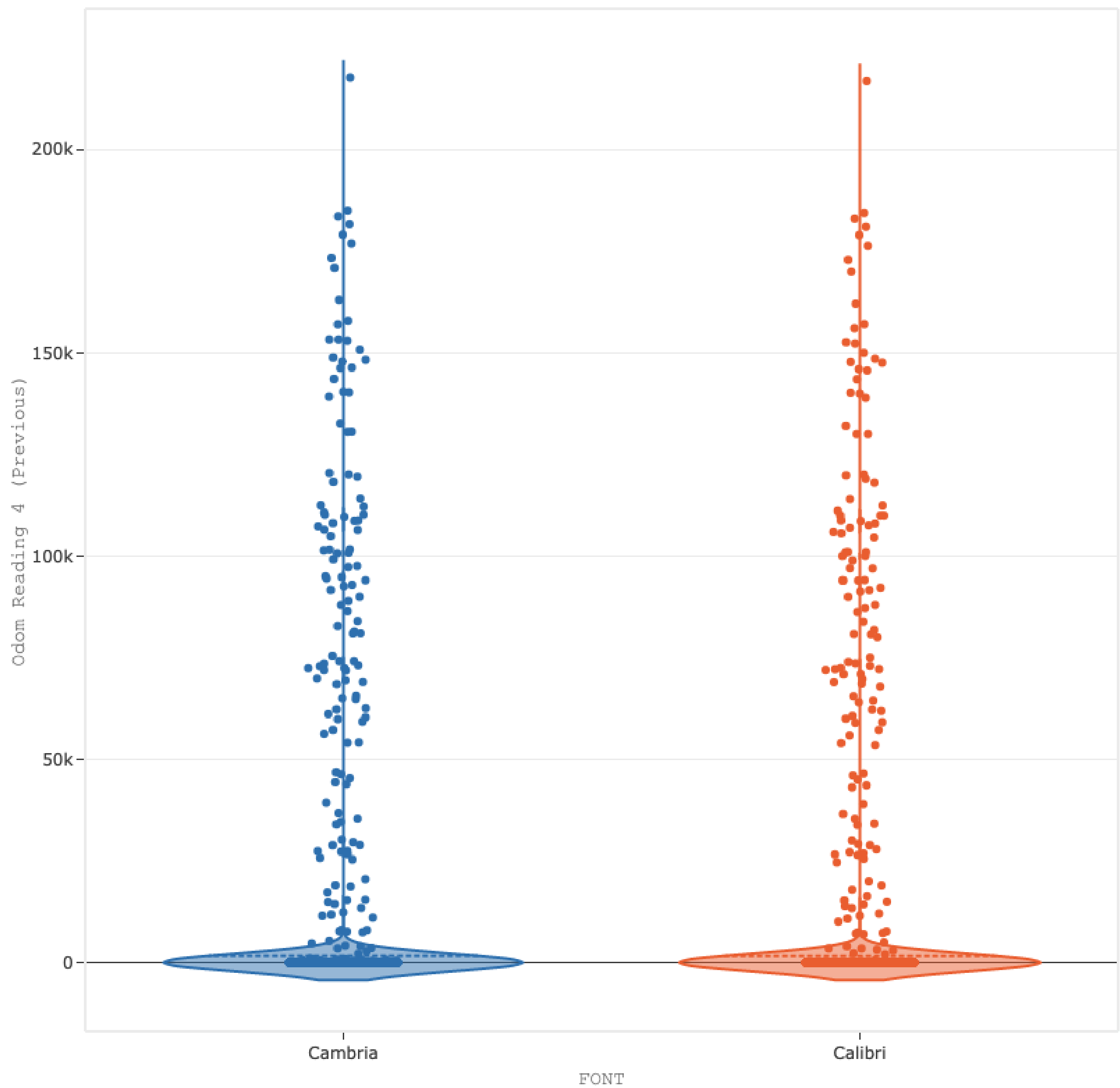
Bonus, including font as a column

All analysis prior to this, was done with the original excel data sheet within Ferret. Adding a categorical column to track the font and loading it back into Ferret provides a few more options.

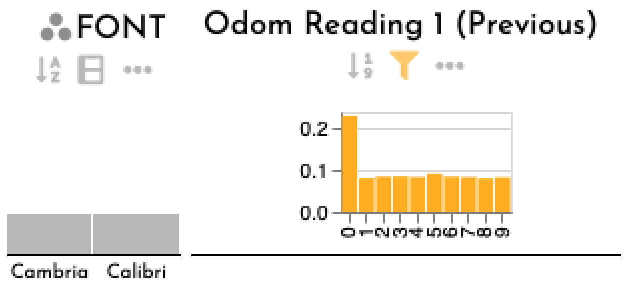
First, the observations of pairs at the extreme are strengthened by the fact that the pairs always include one **Calibri** formatted cell and one Cambria formatted cell.

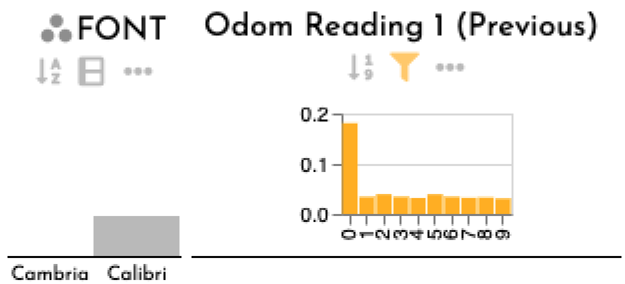


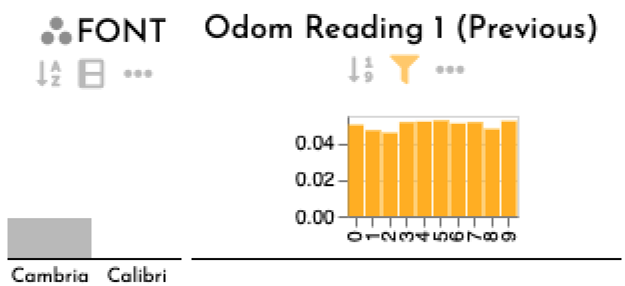
Furthermore, if we plot a violin plot of the previous column faceted by the font, you see they are extremely similar, again strengthening the hypothesis that these data are nearly copies.



Lastly, we can also observe the lack of rounding effects in the Cambria plot using the trailing digit frequency visualization combined with dynamically filtering out one font compared to another.







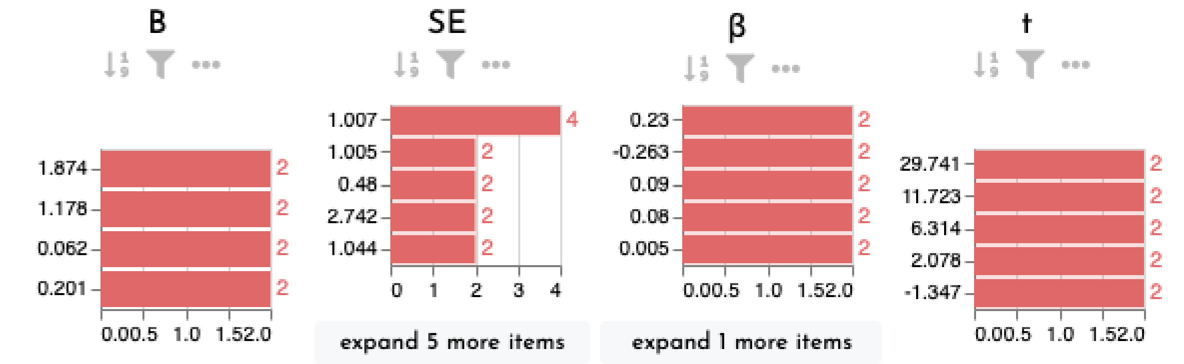
Case Study: DS-Gaming

Retraction: https://doi.org/10.1038/s41598-020-66798-w
Blog: http://steamtraen.blogspot.com/2020/04/some-issues-in-recent-gaming-research.html

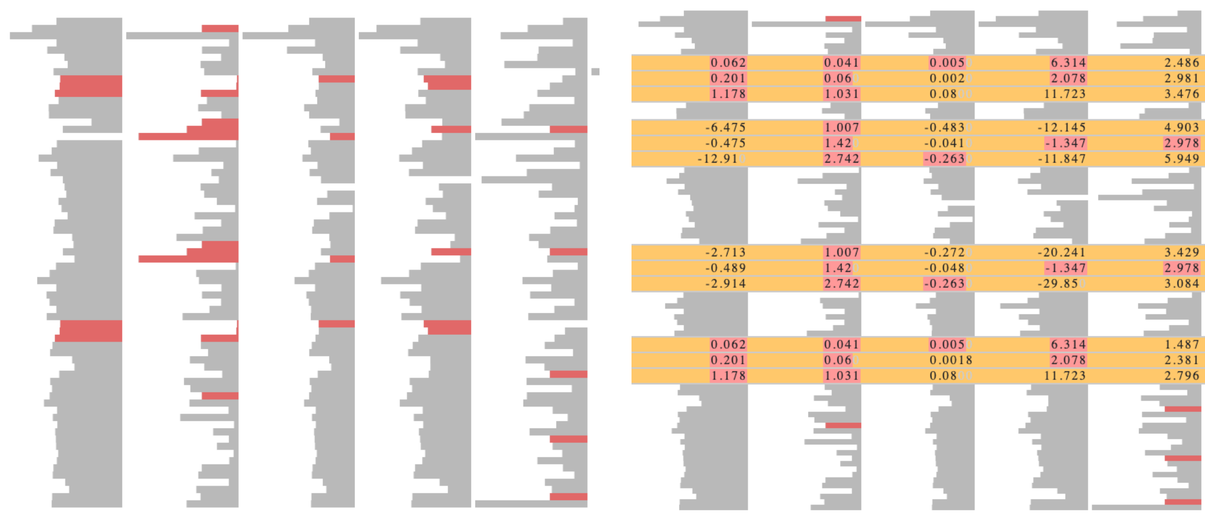
This study looked for a relationship between video gaming habits and sleep habits. A survey was sent over email asking about video gaming habits, demographic information, and sleeping habits. The paper contains a table with summary statistics based on survey responses. Nick Brown, the author of the blog associated with this dataset converted the table into an excel file which we have utilized.

Repeated Regions

Since this table only contains 68 rows in total, the amount of duplicate numbers is a bit high, though it may not be enough to be conclusive on its own.



After highlighting some numbers, however, repeated regions become more clear.



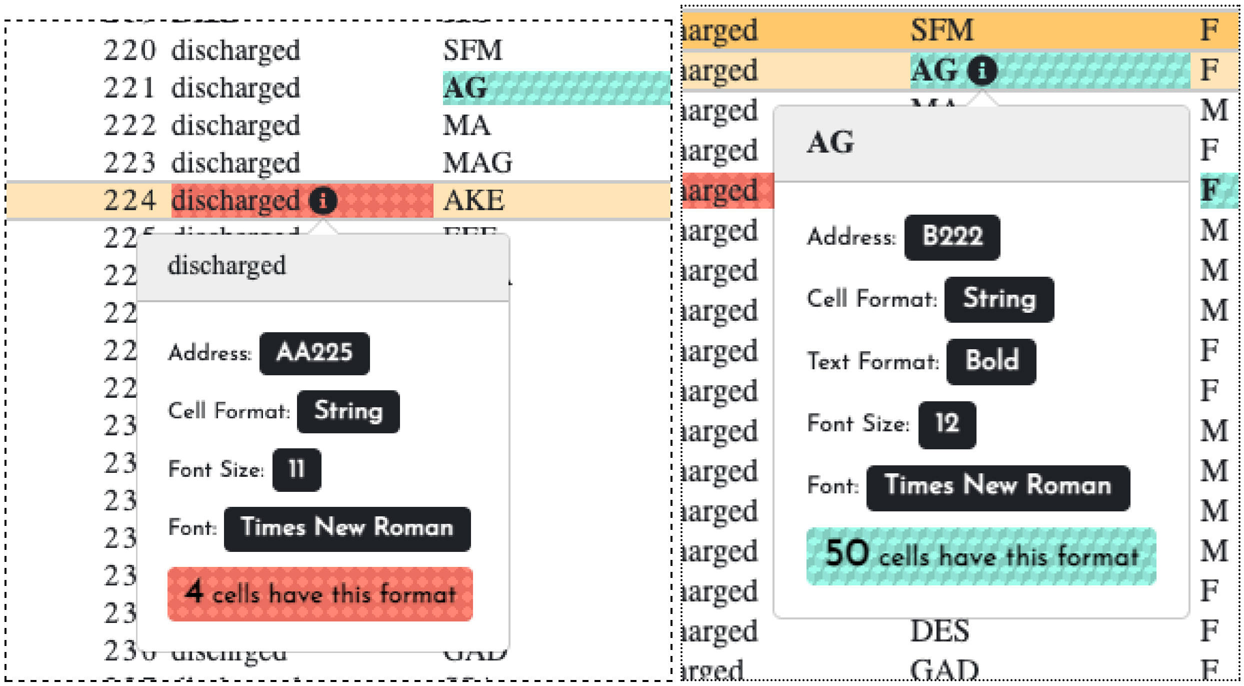
Case Study: DS-Covid

Retraction: https://grftr.news/why-was-a-major-study-on-ivermectin-for-covid-19-just-retracted/
Blog: http://steamtraen.blogspot.com/2021/07/Some-problems-with-the-data-from-a-Covid-study.html

This dataset collected data on how effective and safe ivermectin is for testing Covid-19.

Unexpected Formatting

This dataset contains many instances of unexpected formatting. Excluding the cell data format from the formatting highlighting makes it easier to identify relevant formatting discrepancies, such as outlier fonts:

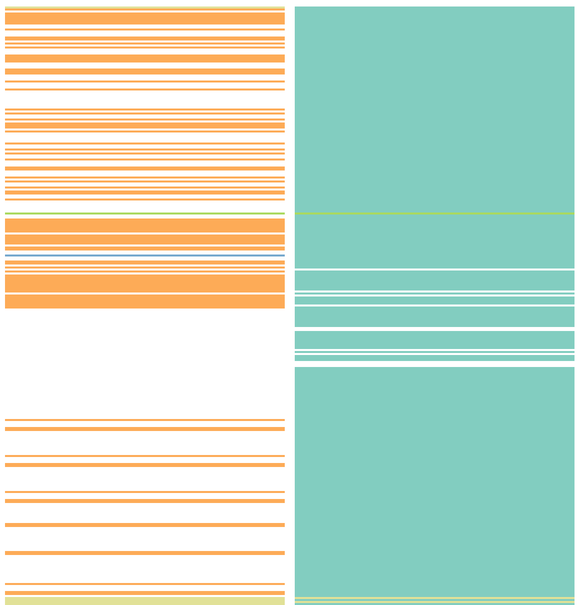


By including the data format of cells in the styling criteria it is easier to spot issues with the actual malformed data. Such errors and inconsistencies are immediately obvious in the detailed view of the table.

TLC (X 103	lymph. %	symptoms date&…	recovery date & -…
4	0.15	Wed Aug 05 2020 …	Sat Dec 05 2020 1…
4.7	0.14	Wed Aug 05 2020 …	Thu Nov 05 2020 …
5.2	0.16	Sat Sep 05 2020 1…	Sat Dec 05 2020 1…
Fri Jan 05 1900 00…	0.15	Sat Sep 05 2020 1…	14/6/2020
6.8	0.14	Sat Sep 05 2020 1…	13/6/2020
5.2	0.15	Mon Oct 05 2020 …	13/6/2020
6.7	0.2	Mon Oct 05 2020 …	14/6/2020 5 days
7.2	0.17	Mon Oct 05 2020 …	15/6/2020
8.9	0.15	Thu Nov 05 2020 …	14/6/2020
7.3	0.16	Thu Nov 05 2020 …	16/6/2020
6.4	0.17	Thu Nov 05 2020 …	16/6/2020
6.4	0.18	Thu Nov 05 2020 …	15/6/2020
5.8	0.2	Thu Nov 05 2020 …	15/6/2020
6.2	0.17	Sat Dec 05 2020 1…	17/6/2020 6 days
7.4	0.15	Sat Dec 05 2020 1…	17/6/2020
9.o	0.18	Sat Dec 05 2020 1…	17/6/2020
8.4	0.16	Sat Dec 05 2020 1…	16/6/2020
6.4	0.2	13/6/2020	16/6/2020
7.2	0.19	13/6/2020	17/6/2020
5.9	0.16	13/6/2020	16/6/2020
7.4	0.15	13/6/2020	18/6/2020
8.3	0.18	14/6/2020	17/6/2020 4 days
7.5	0.2	14/6/2020	17/6/2020
8.4	0.2	14/6/2020	19/6/2020
5.9	0.17	13/6/2020	18/6/2020
8.4	0.18	14/6/2020	19/6/2020
7.9	0.18	15/6/2020	20/6/2020
8.4	0.16	15/6/2020	20/6/2020
7.2	0.15	15/6/2020	19/6/2020
7.4	0.14	15/6/2020	19/6/2020
8.2	0.19	15/6/2020	18/6/2020
8.4	0.2	15/6/2020	18/6/2020
7.9	0.16	16/6/2020	19/6/2020
8.4	0.15	16/6/2020	19/6/2020
4.9	0.17	16/6/2020	20/6/2020
5.9	0.18	16/6/2020	20/6/2020
7.3	16.oo%	17/6/2020	20/6/2020
6.8	0.2	17/6/2020	20/6/2020
7.2	0.18	17/6/2020	22/6/2020

The overview mode can also be helpful in reviewing how many errors of this kind exist.
For instance, here, the left column is recording date values. Orange rows are correctly formatted as a date in Excel, whereas white and yellow rows are strings. In this figure, roughly half of the rows are recorded correctly.

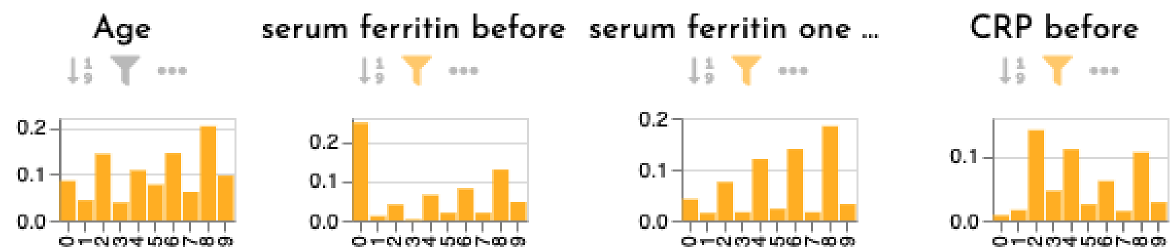
The right column is recording numerical values. The teal rows are correctly formatted as numbers, and the white ones are invalid numbers formatted as strings, such as “9.o%”



These types of errors are likely the result of entering data into a spreadsheet manually.

Unexpected Trailing Digit

There are four columns within this dataset that show a strange preference for even numbers over odd ones. This can be in the following Trailing Digit visualizations. The blog for this post dataset does mention asymmetry in odd/even values for the **Age** column, **however, it does not mention it for the other three columns identified by Ferret.**



DS-Spider: Dataset Description

Blog: https://laskowskilab.faculty.ucdavis.edu/2020/01/29/retractions/\

The three spider studies share some common authors and were retracted in the same wave. The three datasets are related to each other but have different structures and attributes.

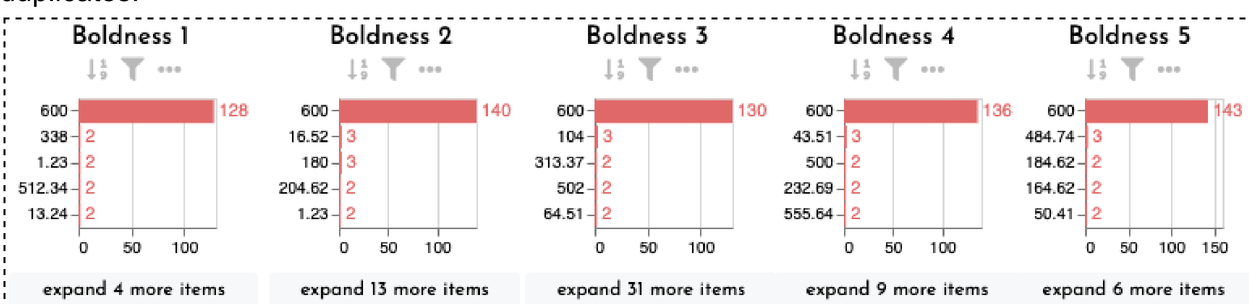
All of the datasets include a "boldness" of spiders. This "boldness" was measured by recording how long it will take spiders to reemerge from their enclosure after a simulated predator attack.

Case Study: DS-Spider-E

Retraction: https://doi.org/10.1098/rspb.2020.0077

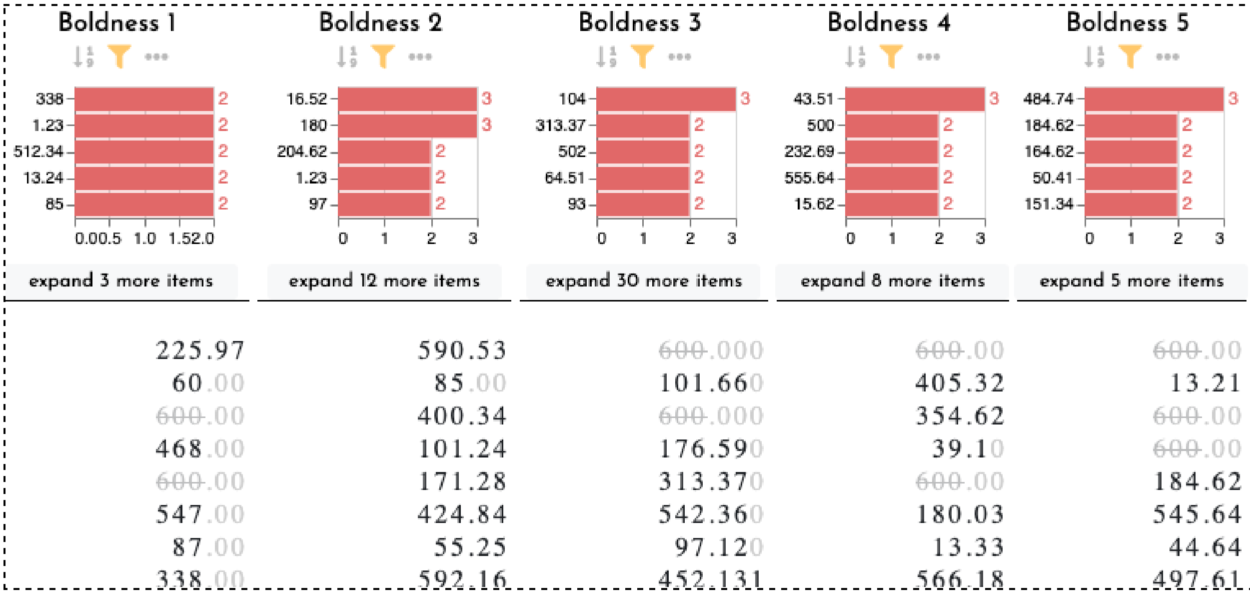
Duplicate Numbers Artifact

The five **Boldness** columns in this dataset, all show a very large number of 600s. Since these are time measurements, 600 seconds corresponds to 10 minutes, the maximum amount of time they waited for a spider to reemerge. In other words, there is a reasonable explanation for these duplicates.

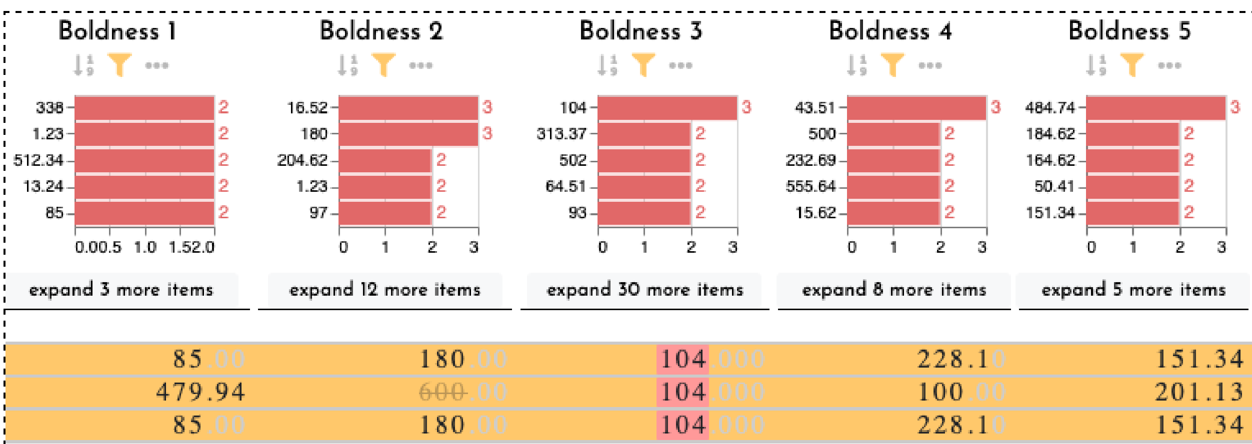


Their presence distorts the bar chart scale, making it difficult to see other duplicate values.

. Ignoring 600 globally removes it from the analysis and strikes out the 600s in the table view. This is different from removing any row that contains a 600 as that would remove a large relevant data to examine.

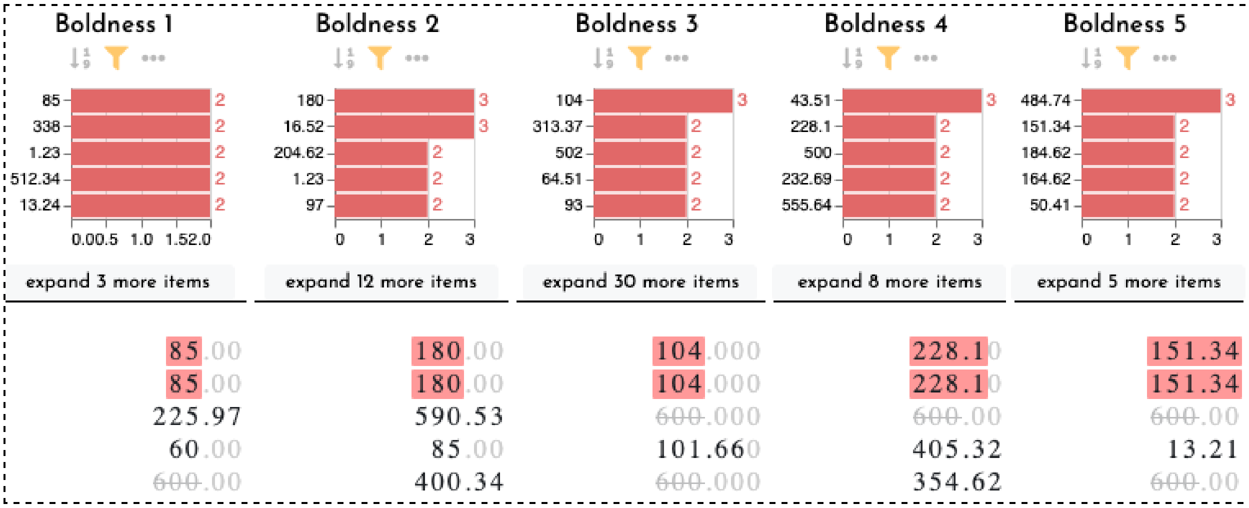


The amount of remaining duplicates is still large for this dataset of 350 rows, especially with the two degrees of precision listed. Highlighting 104 in the **Boldness 3** column makes it easy to examine the neighborhood of cells.



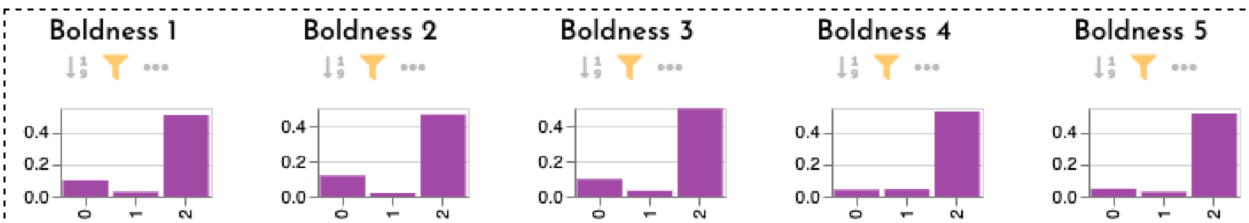
Repeated Regions Artifacts

Highlighting more values makes it more obvious that at least one duplicated row exists in this dataset.

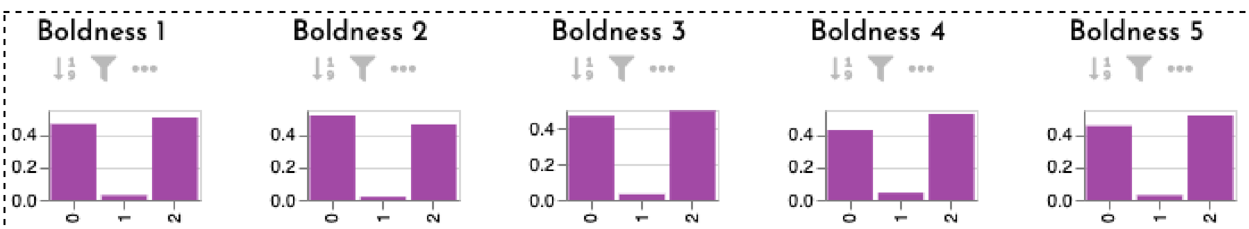


Unexpected Varied Precision Artifacts

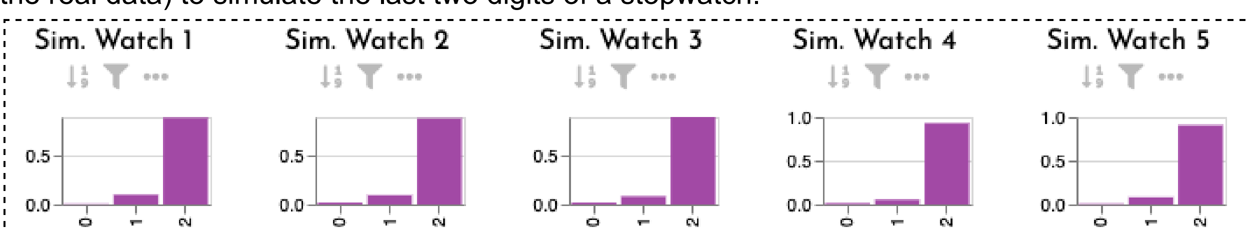
The precision analysis reveals an interesting pattern. There are more values with 0 digits of precision than there are with 1 digit of precision. Since these are time measurements, you would expect most values to have two digits of precision (e.g. 3.12 seconds), less with one digit of precision (e.g. 3.1 seconds), and very few to have zero digits of precision (e.g. 3 seconds).



This anomaly is not due to the large numbers of 600s. The chart above is ignoring 600s. The chart below is what it would look like with 600s included.



The expected results are easy to simulate. The charts below are created from an excel spreadsheet that used **round(rand(),2)** in five columns by 350 rows (the same dimensions as the real data) to simulate the last two digits of a stopwatch.



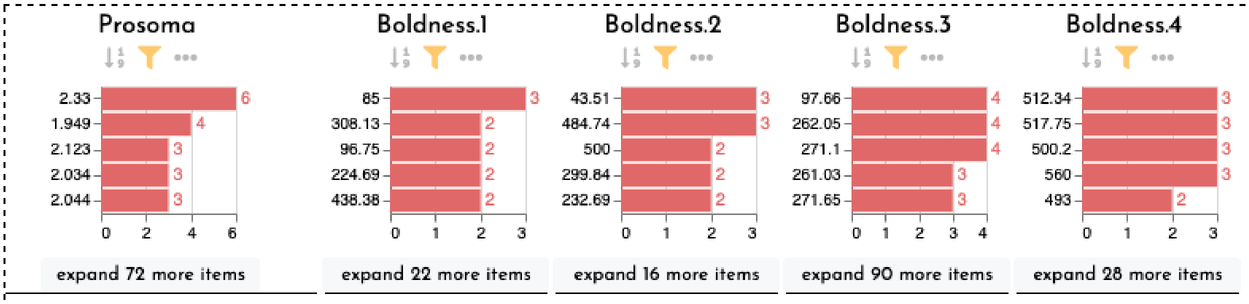
This unexpected distribution of precision is not mentioned in any blog posts.

Case Study: DS-Spider-P

Retraction: https://doi.org/10.1098/rsbl.2020.0062

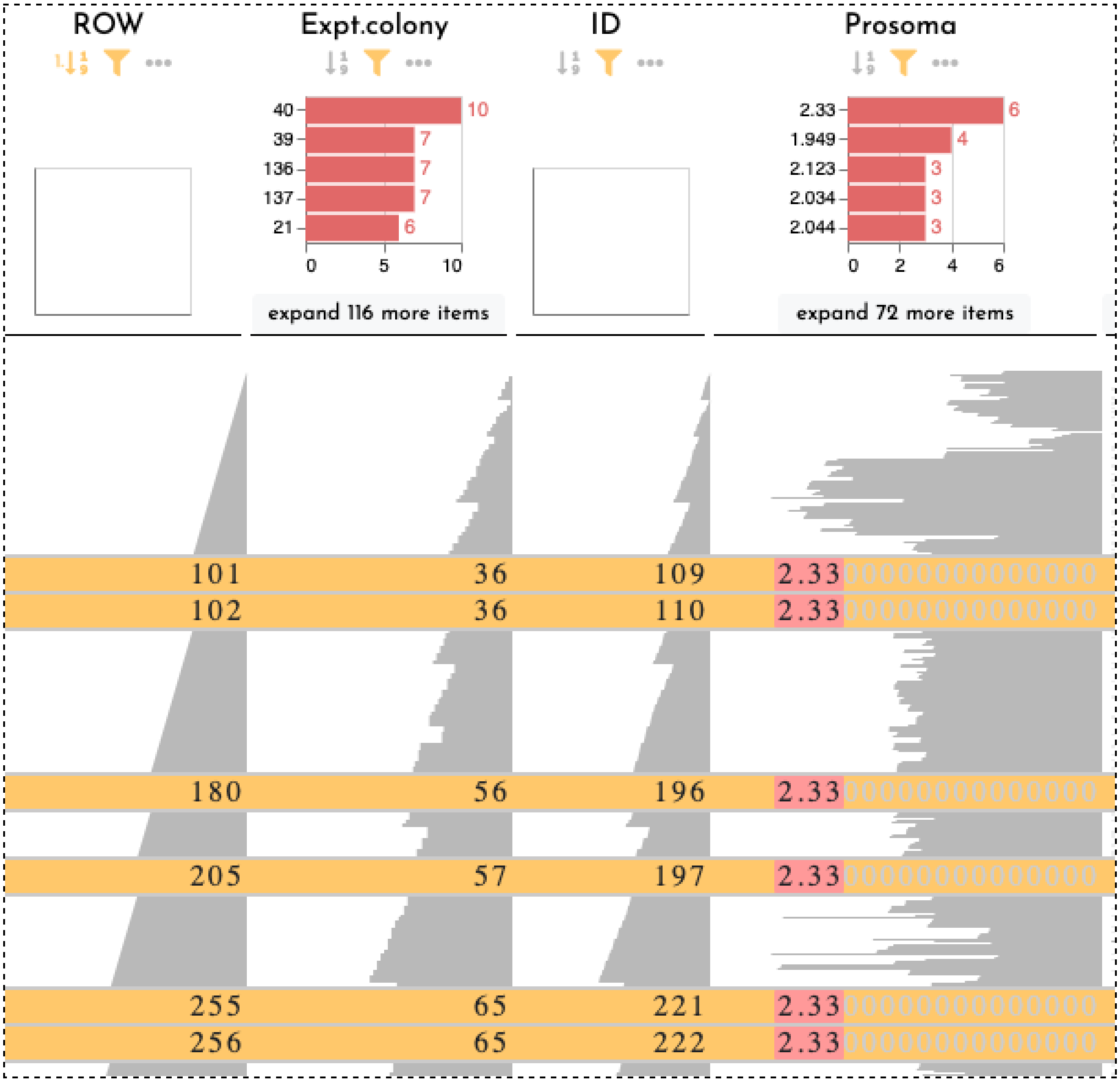
Duplicate Numbers Artifact

Similar to **DS-Spider-E** there is a duplicate numbers artifact in this dataset of 479 rows. Again 600 is ignored from the analysis.



Repeated Regions Artifact

Highlighting the most duplicated value (2.33) in the **Prosoma** columns shows that these values are from actually repeated regions.

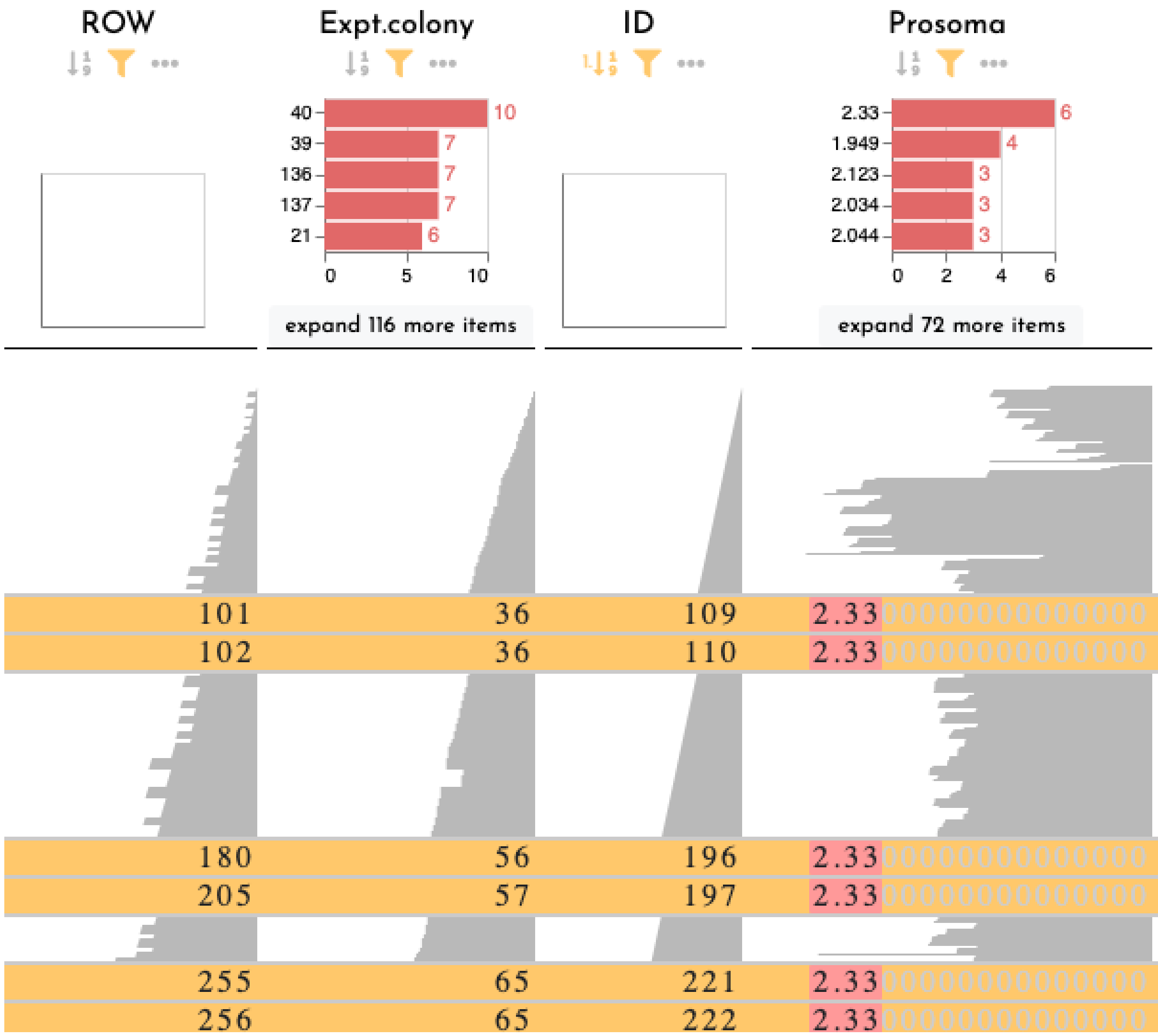


Ordering Artifacts.

The image above also illustrates an ordering artifact in this dataset. The **ROW** column is the original row value in the dataset (automatically inserted by Ferret). The **Expt.colony** and **ID** columns share a strange relationship with **ROW.**

Sorting by the ID column reveals that 2.33 appears even more like a repeated region with the data sorted by ID (which it is likely to have been at some point). In addition, this sort shows that

ID and **Expt.colony** also share a strange relationship with each other. **This ordering artifact is not mentioned in any blog posts.**



Case Study: DS-Spider-I

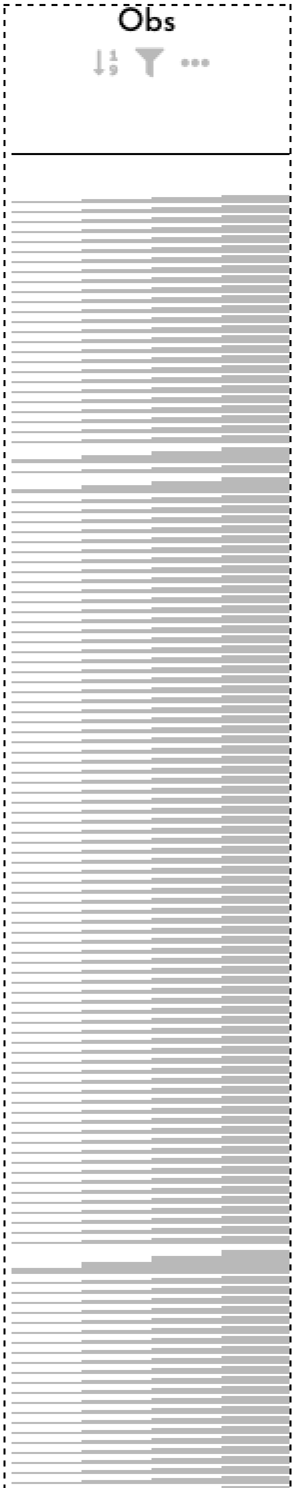
Retraction: https://doi.org/10.1086/708066

This dataset is still related to the same boldness measurement of spiders. However, it is formatted differently than the previous two — it is in long format. Before each row contained all of the boldness scores for a single spider. In this long format, each row corresponds to one observation. So there are 5 rows of observation for each spider.

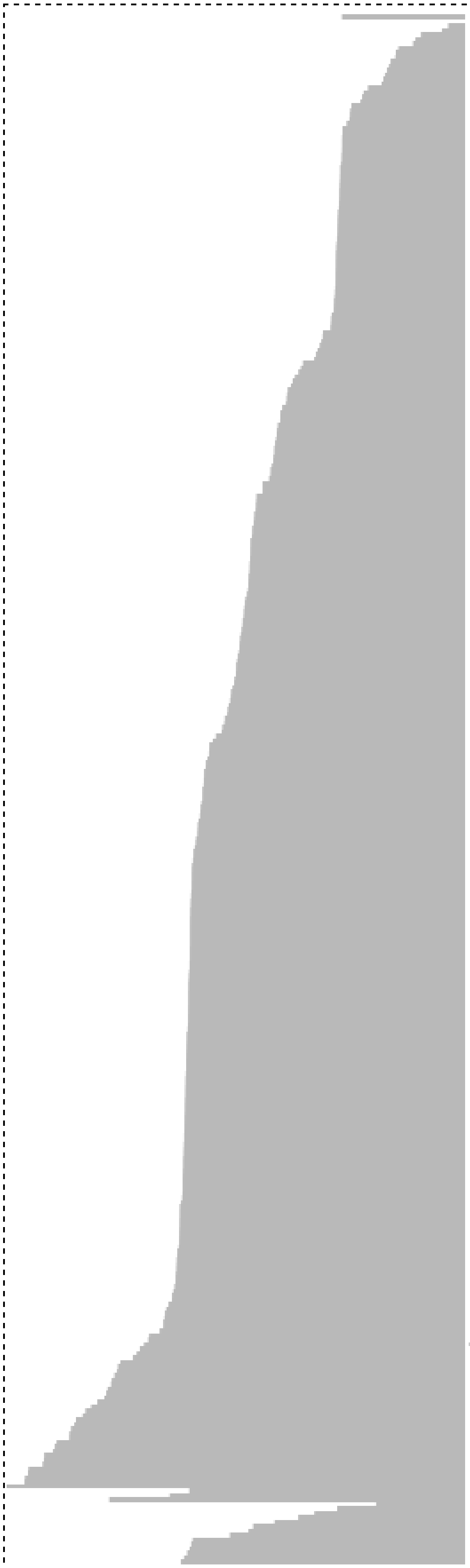
Ordering Artifacts

There are a few unusual ordering patterns in this dataset.

The **obs** column generally follows a repeated structure, of 1,2,3,4,5. Alternating through the 5 observations for each individual spider. However, this pattern is broken several times.



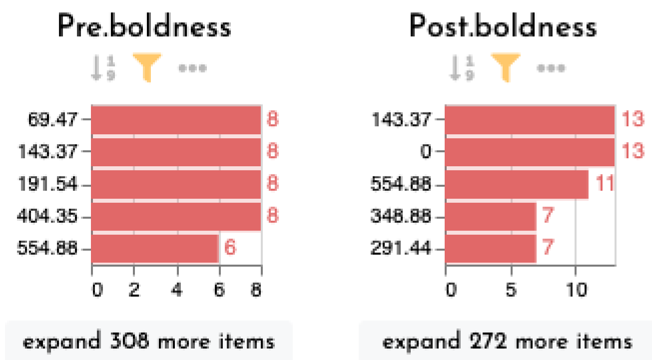
There is also unexplained ordering in the **Percent.mass.change** column.



This column is monotonically increasing for a majority of the dataset with the exception of the very first value, and the last few values. It is not clear how this ordering could occur. **Neither of these ordering artifacts are mentioned in any blog post.**

Duplicate Numbers Artifacts

Similar to the other spider datasets, the boldness columns contain many duplicates.
Since this dataset is in long format, there are only two columns. **Pre.boldness** and **Post.boldness**. With 1745 rows.



Repeated Regions

We know that this dataset contains repeated regions thanks to the blog post written about it by one of the co-authors of the retracted paper. That blog mentions the duplicate numbers found in the data in this form, then describes how when the wide formatted version of this data contained repeated regions. Since Ferret does not support the ability to convert between long and wide data formats we were not able to identify this known artifact in this dataset.

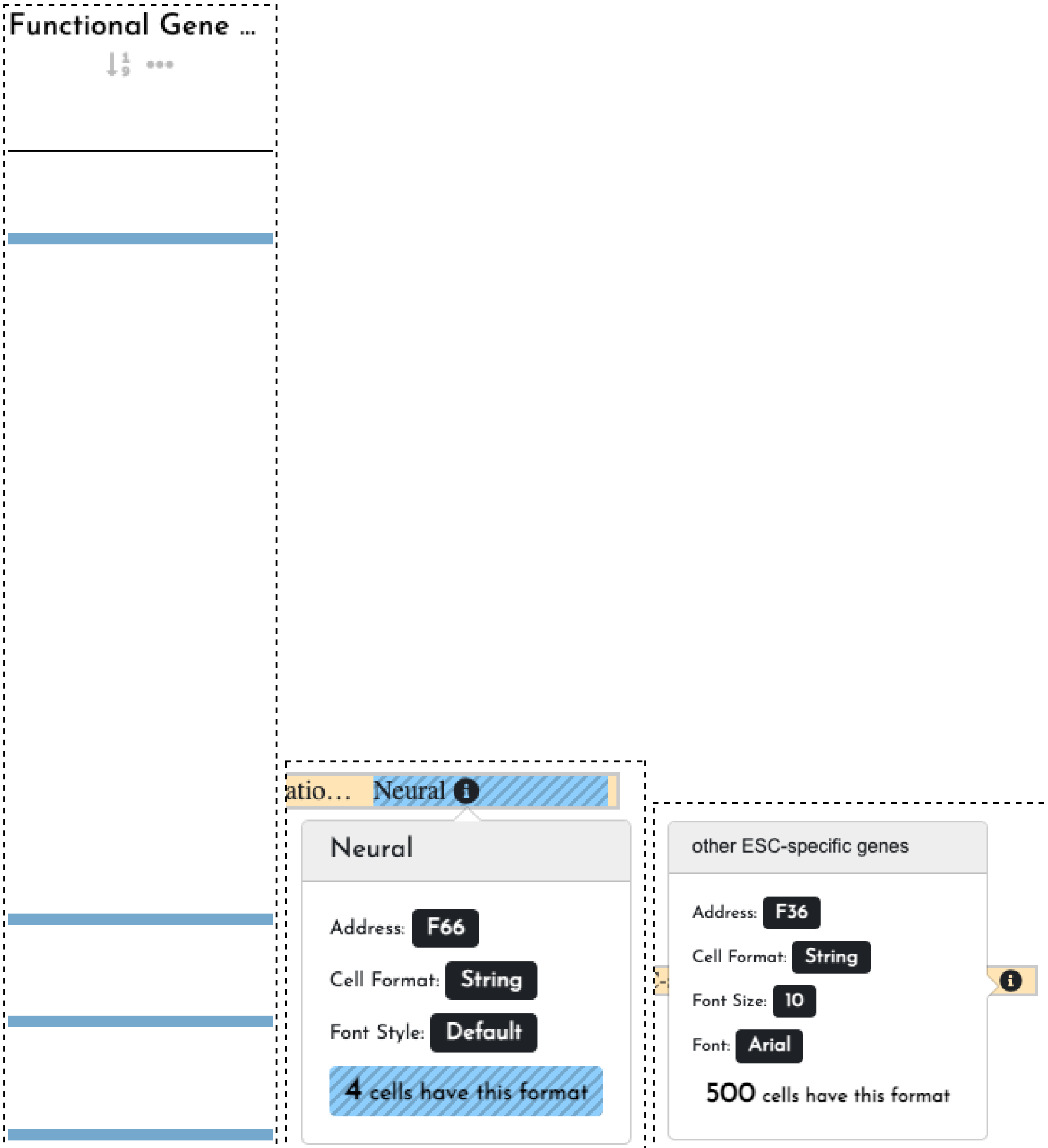
Case Study: DS-Glioma

Retraction: https://doi.org/10.1021/acs.molpharmaceut.9b00837
Blog: NA - found via retraction watch.
This dataset contains numerical measurements of **fold change** of mouse embryonic stem cells.

When loading this dataset, it is evident there are a few outlier cells with regard to formatting.

Formatting artifacts

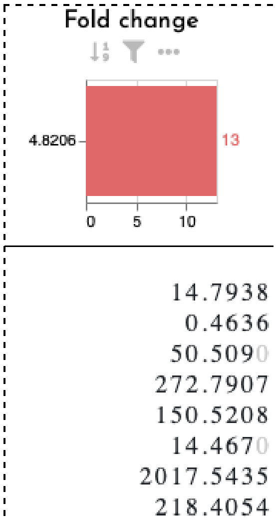
Agg...	Rank	Sele...	ROW	Accession	Symbol	Description	Fold change	Functional Gene ...	Functional Gene Grouping (II)
		☐							
	1	☐	1	NM_010425	Foxd3	Forkhead box D3	14.7938	ESC-specific genes	Transcription factors maintaining "stem-ness"
	2	☐	2	NM_010258	Gata6	GATA binding prot…	0.4636	ESC-specific genes	Transcription factors maintaining "stem-ness"
	3	☐	3	NM_010262	Gbx2	Gastrulation brain …	50.509	ESC-specific genes	Transcription factors maintaining "stem-ness"
	4	☐	4	NM_028016	Nanog	Nanog homeobox	272.7907	ESC-specific genes	Transcription factors maintaining "stem-ness"
	5	☐	5	NM_030676	Nr5a2	Nuclear receptor s…	150.5208	ESC-specific genes	Transcription factors maintaining "stem-ness"
	6	☐	6	NM_010264	Nr6a1	Nuclear receptor s…	14.467	ESC-specific genes	Transcription factors maintaining "stem-ness"
	7	☐	7	NM_013633	Pou5f1	POU domain, class…	2017.5435	ESC-specific genes	Transcription factors maintaining "stem-ness"
	8	☐	8	NM_011443	Sox2	SRY-box containin…	218.4054	ESC-specific genes	Transcription factors maintaining "stem-ness"
	9	☐	9	NM_023755	Tcfcp2l1	Transcription facto…	228.8357	ESC-specific genes	Transcription factors maintaining "stem-ness"
	10	☐	10	NM_009482	Utf1	Undifferentiated e…	1194.5975	ESC-specific genes	Transcription factors maintaining "stem-ness"
	11	☐	11	NM_009556	Zfp42	Zinc finger protein 42	1106.6193	ESC-specific genes	Transcription factors maintaining "stem-ness"
	12	☐	12	NM_147778	Commd3	COMM domain co…	0.3192	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	13	☐	13	NM_007759	Crabp2	Cellular retinoic ac…	0.3381	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	14	☐	14	NM_007904	Ednrb	Endothelin recepto…	0.1798	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	15	☐	15	NM_010202	Fgf4	Fibroblast growth f…	346.5816	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	16	☐	16	NM_010203	Fgf5	Fibroblast growth f…	0.7147	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	17	☐	17	NM_008071	Gabrb3	Gamma-aminobut…	2.8777	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	18	☐	18	NM_010253	Gal	Galanin	5.0343	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	19	☐	19	NM_010346	Grb7	Growth factor rece…	27.4283	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	20	☐	20	NM_010407	Hck	Hemopoietic cell k…	2.9869	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	21	☐	21	NM_026820	Ifitm1	Interferon induced …	27.1546	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	22	☐	22	NM_010560	Il6st	Interleukin 6 signa…	0.9224	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	23	☐	23	NM_021099	Kit	Kit oncogene	36.3721	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	24	☐	24	NM_010094	Lefty1	Left right determin…	212.8439	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	25	☐	25	NM_177099	Lefty2	Left-right determin…	287.4606	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	26	☐	26	NM_013584	Lifr	Leukemia inhibitor…	0.1562	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	27	☐	27	NM_013611	Nodal	Nodal	40.6486	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	28	☐	28	NM_008711	Nog	Noggin	0.1023	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	29	☐	29	NM_010949	Numb	Numb gene homol…	0.5352	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	30	☐	30	NM_008960	Pten	Phosphatase and te…	0.4161	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	31	☐	31	NM_009144	Sfrp2	Secreted frizzled-r…	0.1661	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	32	☐	32	NM_011562	Tdgf1	Teratocarcinoma-d…	897.2167	ESC-specific genes	Signalling molecules required for pluripotency and self-Renewal
	33	☐	33	NM_026396	Bxdc2	Brix domain contai…	1.2078	ESC-specific genes	other ESC-specific genes
	34	☐	34	NM_007657	Cd9	CD9 antigen	0.5989	ESC-specific genes	other ESC-specific genes
	35	☐	35	NM_017398	Diap2	Diaphanous homol…	1.1919	ESC-specific genes	other ESC-specific genes
	36	☐	36	NM_010068	Dnmt3b	DNA methyltransf…	15.6366	ESC-specific genes	other ESC-specific genes
	37	☐	37	NM_030694	Ifitm2	Interferon induced …	0.8363	ESC-specific genes	other ESC-specific genes
	38	☐	38	NM_183029	Igf2bp2	Insulin-like growth…	0.8695	ESC-specific genes	other ESC-specific genes
	39	☐	39	NM_145833	Lin28	Lin-28 homolog (…	100.8125	ESC-specific genes	other ESC-specific genes
	40	☐	40	NM_013723	Podxl	Podocalyxin-like	35.1346	ESC-specific genes	other ESC-specific genes
	41	☐	41	NM_011263	Rest	RE1-silencing tran…	1.5946	ESC-specific genes	other ESC-specific genes



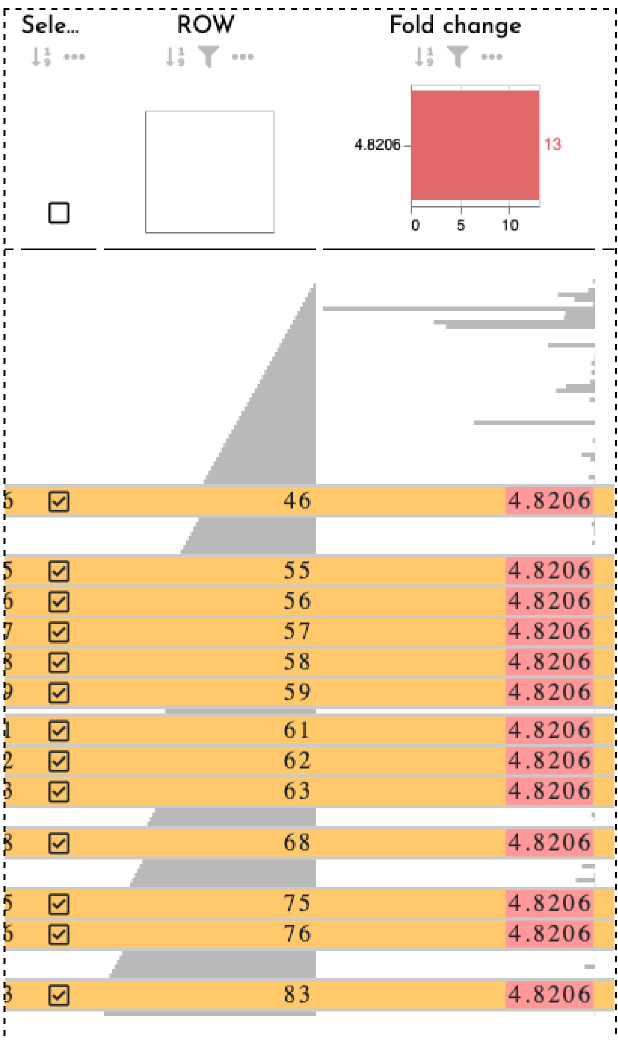
These four outlier table cells still have an unset font and font-size. In excel these 4 cells are displayed and listed as Arial/10. However, there is a still difference in how the cells are saved, indicating some difference in formatting happened to these cells compared to the others at some point. **This formatting artifact has not been mentioned in any blog post.**

Duplicate Numbers Artifact

Switching to the numerical analysis, we can see that the **Fold change** only has one value duplicated (4.8206), but it is duplicated 13 times in a dataset of under 84 rows.



By highlighting the frequent value and switching to overview mode, we can clearly see the repeated region. **This repeated region artifact has not been mentioned in any blog post.**



Case Study: DS-Fly

Editor's Note: https://www.doi.org/10.1098/rspb.2021.0505
Blog: https://pubpeer.com/publications/70DDAFDEA32DD2D9181998DBF1EECB

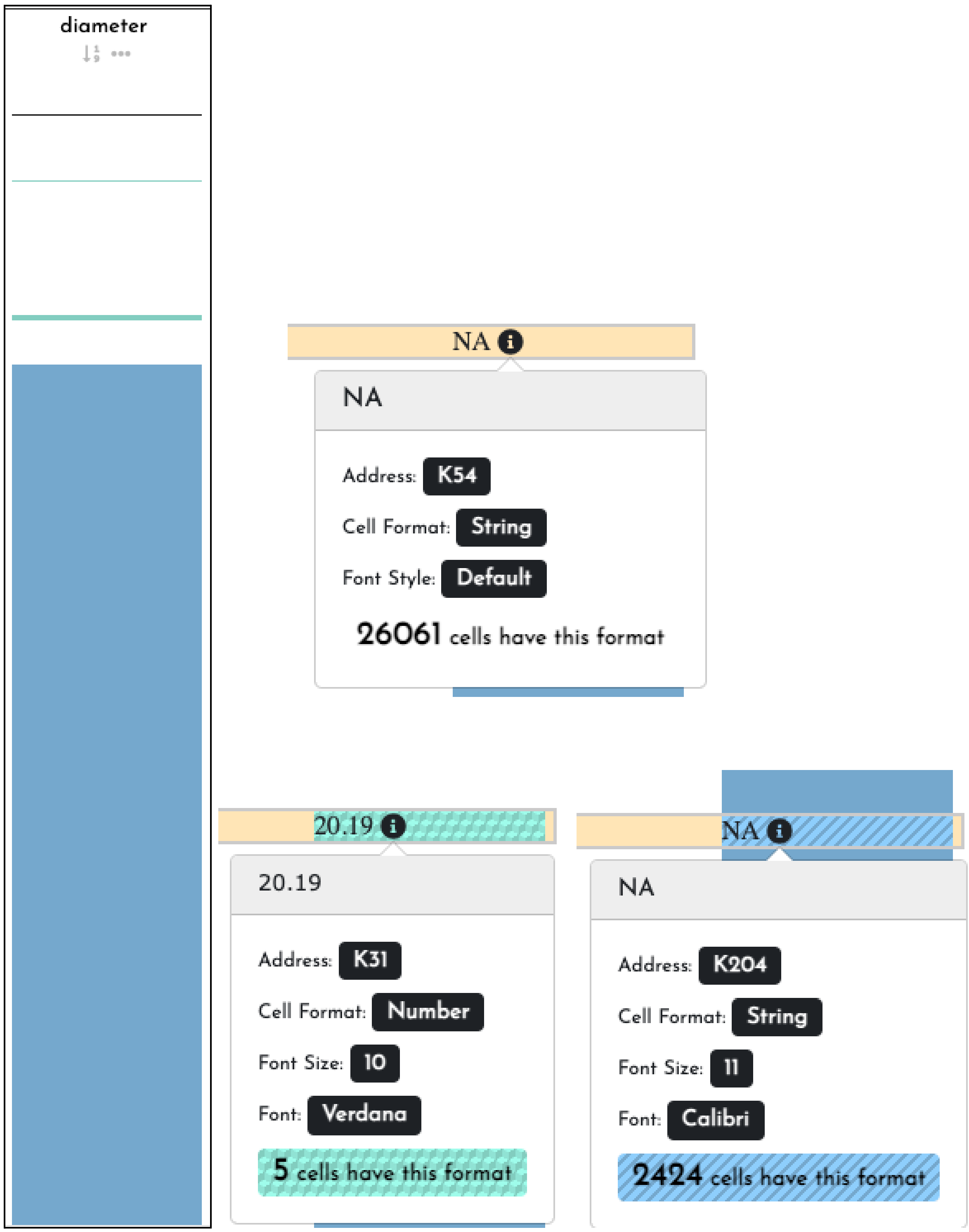
This biology experiment was studying the behavior of flies. This dataset contained two separate sheets. The first contains a column named **diameter**, which is the diameter of the flies. The second sheet includes **dispersal,** which measures the distance a fly has flown.

Formatting artifacts

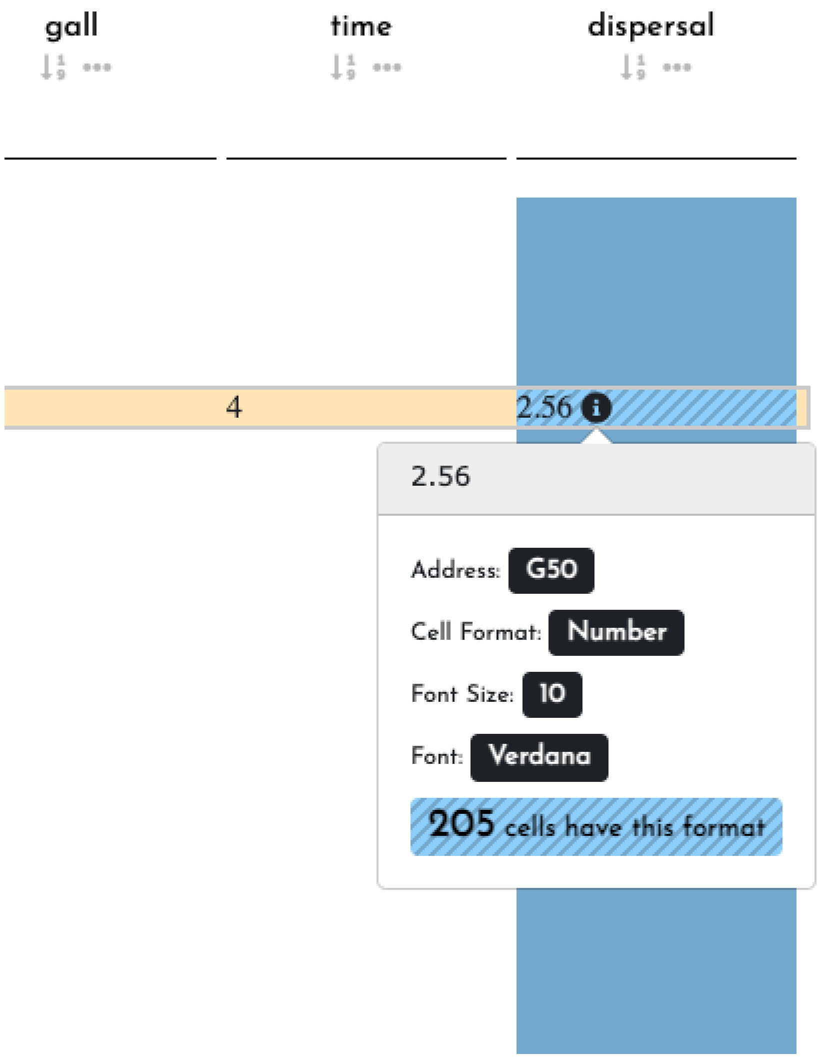
In the first analysis of Sheet 1, an outlier in formatting can be quickly spotted:

distance	stems	gall	fly	larvae	bird	mordellistena	eury.obs	eury.gig	diameter
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
78	7	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
87	9	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
82	12	0	0	NA	NA	NA	NA	NA	NA
74	22	4	1	1	0	0	0	0	20.19
74	22	4	1	1	0	0	0	0	22.22
74	22	4	1	1	0	0	0	0	19.99
74	22	4	1	1	0	0	0	0	21.42
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA
74	22	4	0	NA	NA	NA	NA	NA	NA

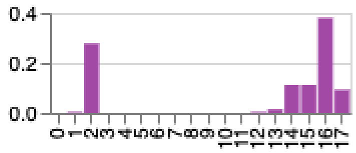
Switching to the overview mode reveals a bigger picture. This column contains three different kinds of formatting (ignoring the cell data format). **This formatting artifact has not been mentioned in any blog post.**



In the second sheet we see that dispersal is formatted differently than the rest of the sheet. On closer investigation, it has the same font as the green cells from sheet 1 (Verdana, 10-point).



Stepping through the numerical artifacts in **dispersal** column appears to have some unexpected variation in precision.

dispersal

0.94792489838110930
1.01000000000000000
1.62319303830208830
1.62000000000000000
0.13212690720821374
1.80511228341283210
1.82000000000000000
1.16594676271109000
1.23613609877601260
2.84276271996890500
2.89912519843103930
1.24394910287012600
2.01634300440641030
2.31088659789645600
2.77029445937031360
0.79322363328396730
1.95632838106376950
2.34186656380859300
0.46747358616961127
1.17609170796926630
1.88947546233840050
1.95000000000000000
0.64197374897472770
1.17132357341018430
1.22560528909542970
0.07369836242426997

Ordering artifact combined with precision artifact

In Sheet 1, the **diameter** column has this same variation in precision. There also appear to be some ordering effects with respect to if values have 2 degrees of precision, or many.

The first three regions of non-null diameters all have 2 digits of precision.

ROW	T diameter
28	–
29	–
30	20.19
31	22.22
32	19.99
33	21.42
34	–
35	–

ROW	T diameter
75	–
76	15.68
77	18.95
78	21.02
79	16.78
80	19.11
81	19.85
82	16.52
83	–
84	–

- - -	
101	–
102	16.23
103	18.25
104	14.75
105	19.22
106	17.24
107	16.98
108	–
109	–

In the next region, every number has high precision:

129	–
130	19.182055669222805
131	19.250352060765866
132	19.225638308856986
134	–

After this, there are larger regions of nun null values. These vary, but they are predominantly filled with high precision numbers, with a few low precision numbers near the bottom of the region.

213	–
214	12.233923184938636
215	17.023932759034206
216	18.282996426388973
217	20.688780448623383
218	20.71885488335425
219	20.88786368060483
220	21.626116559670525
221	23.375222757418683
225	19.5
226	16.24
227	–

319	–
320	17.665179804485
321	19.682002079175124
322	21.050236949953526
327	16.24
328	–
512	–
513	13.590875280868142
514	15.021849729490205
515	15.45706703080656
516	15.58543819176719
517	16.591257829707036
518	16.70496590594684
519	22.170371221034777
522	15.45
523	17.8
524	–
580	–
581	13.273803525640387
582	15.357050841163028
583	20.31891757217873
584	22.41311229861569
589	15.42
590	20.4
591	–

Case Study: DS-Priming

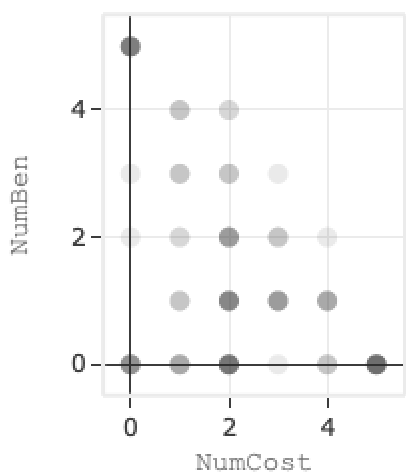
Retraction: https://link.springer.com/article/10.1007/s11002-016-9401-6
Blogs:

- https://blog.openmktg.org/2021/07/retracted-article-why-money-meanings.html
- https://www.tandfonline.com/doi/full/10.1080/01973533.2015.1124767

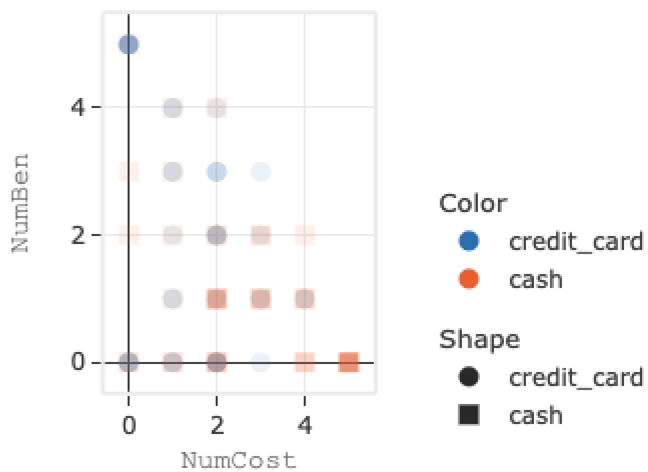
This psychology study measured different priming effects on charitable behavior. Specifically, if there was a difference in being primed with words related to cash vs. credit cards. They provided word completion tasks and counted the number of words the participant responded with that indicated a benefit of volunteering (**NumBen**) and the number of words that correspond to a cost of volunteering (**NumCost**). The original dataset included an experimental condition with four values. For convenience we have added a new column to the beginning of the dataset (**CashOrCredit_Ferret**)

Deviations from domain expectations in response distribution

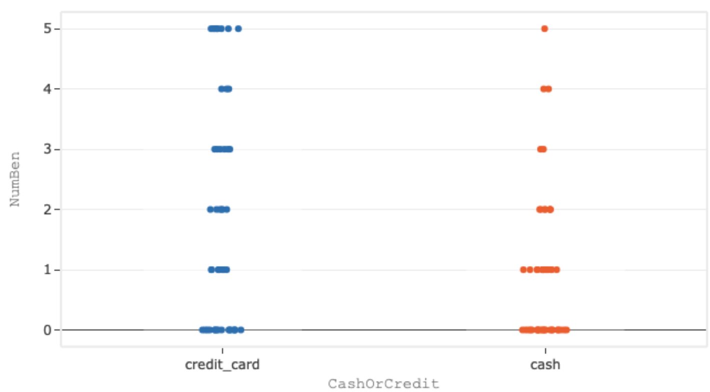
The primary numerical data from this dataset is the number of cost words and benefit words selected by each participant. One way to view this data is through a scatterplot

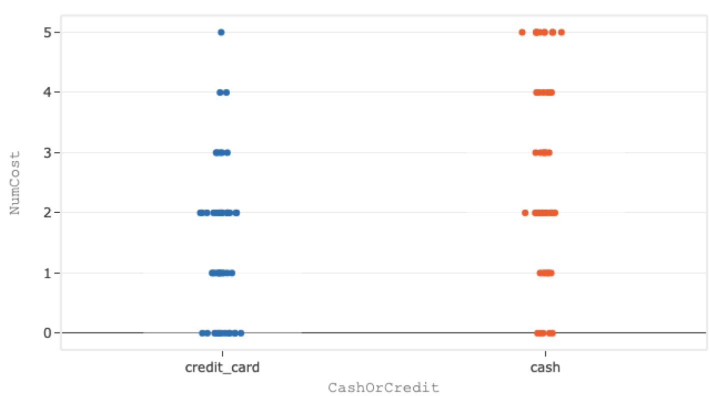


Due to overplotting, opacity is used to distinguish points with a few participants (light gray) and many participants (dark gray). In the first scatterplot you see can many participants near the boundaries (5 benefit words, 0 cost words), and (0 benefit, 5 cost words). Applying a color and shape encoding reveals more information, that most of the (5,0) participants are in the credit condition and the (0,5) are mostly in the cash condition.



Switching to faceted strip plots reveals that in (5,0) and (0,5) there is only one outlier.





Repeated Regions of word responses

Back in the tabular visualization sorting by **NumBen** and **NumCost** is a convenient method to surface the (0,5) group in the table. Examining the word responses reveals that most of the participants in this group produced the same word across multiple different trials.

FOOT	NAP	SPOOK	RECOVERY	0	5
FOOT	NAP	SPOOK	RECOVERY	0	5
FOG	NAG	SPOKEN	RECOMMEND	0	5
FOND	NAP	SPOOK	RECOVERY	0	5
FOOT	NAP	SPOOK	RECOVERY	0	5
FOOT	NAP	SPOOK	RECOVERY	0	5
FOOT	NAP	SPOOK	RECOVERY	0	5
FOOT	NAP	SPOOK	RECOVERY	0	5
FOOL	NAP	SPOOK	RECOVERY	0	5
FORT	NAP	SPOT	RECONSTRUCT	0	5
FOOL	NAP	SPOOK	RECOVERY	0	5

Case Study: “Clean” Dataset

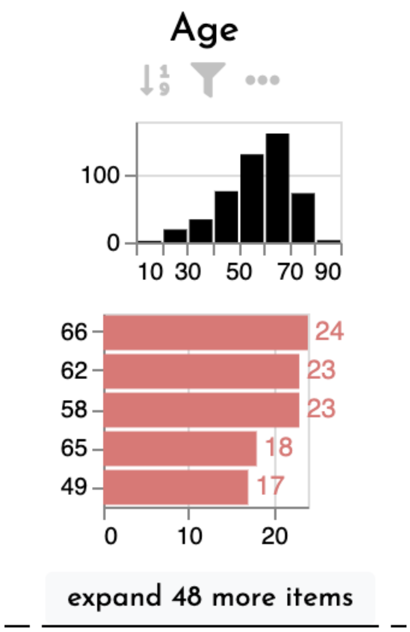
Paper: https://pubmed-ncbi-nlm-nih-gov.ezproxy.lib.utah.edu/28783718/

Dataset Source:
https://www.cbioportal.org/study/summary?id=metastatic_solid_tumors_mich_2017

We assume that this data is not fraudulent. We picked it by randomly selecting a medium-sized dataset listed on the first page of https://www.cbioportal.org/.

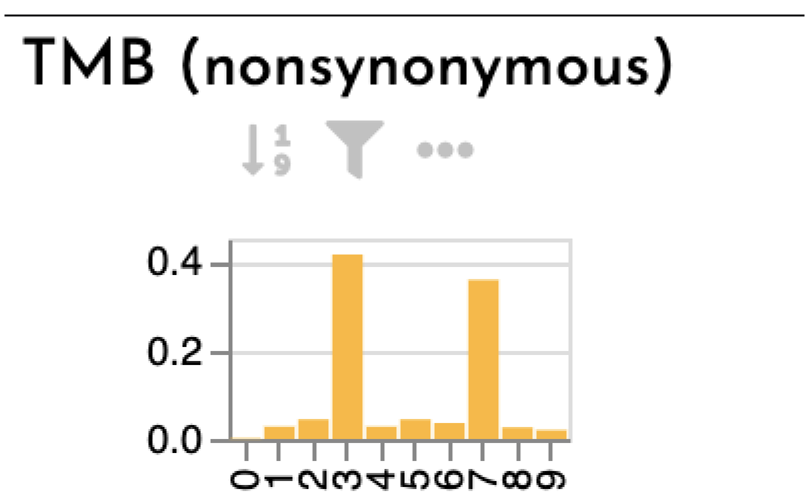
Duplicate Numbers

The first numerical column is age, which records the age of the patient as a whole number. Here we can see that duplicate ages are present. However, the number of duplicates are not alarming. There are roughly 100 possible values, and 500. If the ages were uniformly distributed you would expect 5 duplicates of every age. However, as we can see the ages are not uniform, with a peak between 50-70 years of age. The most frequently duplicated values are in this common range. This is an expected pattern.



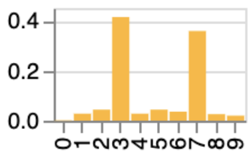
Trailing Digits

For **Column TMB (nonsynonymous)** there is a strange pattern for the trailing digits at first glance:



There is a very high frequency of 3s and 7s. However, a quick review of the first few numbers below gives a reasonable explanation for this. The numbers appear to be fractions converted to numbers.

TMB (nonsynonymous)



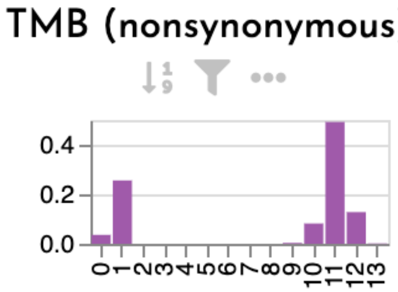
203.8333333330000
101.1333333330000
80.0666666667000
74.0000000000000
59.5666666667000
57.3333333333000
50.2333333333000
49.7666666667000
47.6333333333000

With some extra effort, it can be seen that these are all fractions of 30:

- 26/30 = 0.833333
- 4/30 = 0.133333
- 2/20 = 0.066666
- 0/30 = 0.00000
- 17/30 = 0.56666
- 10/30 = 0.33333
- 7/30 = 0.233333
- 23/30 = 0.766666
- 19/30 = 0.633333
- …

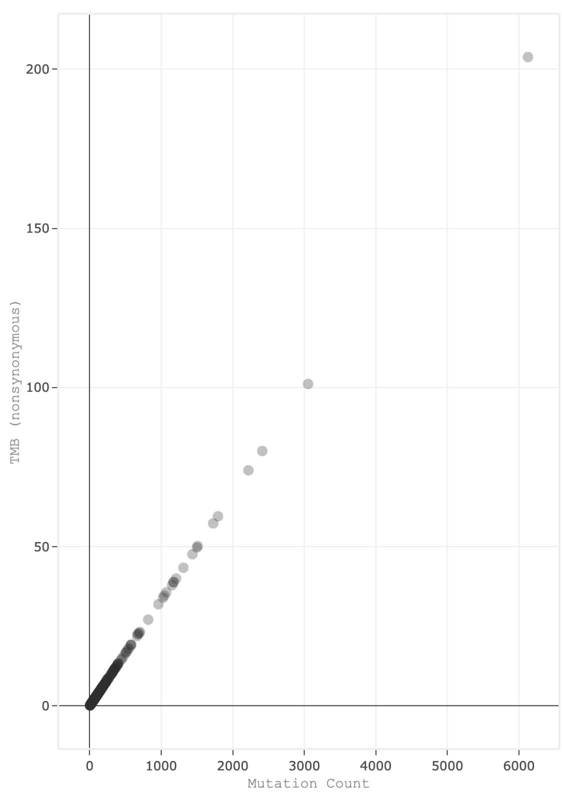
Precision

Similarly, precision may appear strange at first, but can be explained by the same phenomenon.



Domain expectations

It is not possible for me to effectively determine if data breaks domain expectations in an unexpected way. That said, I did notice that two variables appear to be perfectly correlated, specifically **Mutation Count** and **TMB (nonsynonymous)**



However, after a quick google search, I found this article https://www-ncbi-nlm-nih-gov.ezproxy.lib.utah.edu/pmc/articles/PMC7710563/

Which defines TMB as “... the number of somatic mutations per megabase of interrogated genomic sequence, varies across malignancies.” So it does seem reasonable that the values are directly related to each other.