Issue # 18: Numerator Quantity Imputation Method Validation

Ron Huang

November 21, 2025

Contents

1 USDA Data Quality

In this document, we examine the validity of quantity imputed through method discussed in Issue 13. The validation is conducted as follows:

1. Merge USDA Branded Food Dataset with Nielsen 2020 penal
2. In item table, compute the errors and absolute errors as percentage of package weights in Nielsen data for each product defined by a UPC
3. In household-month panel table, compute monthly quantity purchased by both actual packages weights and imputed packages weights. Compute errors and absolute errors as percentage of the actual total quantity purchased for each household-month

Some descriptive results of attempting to merge USDA and Nielsen 2020 data:

- Matched 367 out of 76,122 items with USDA package weight data.
- Matched 37 out of 7,538 brands with USDA package weight data.
- Matched 25,110 out of 76,122 items from brands with USDA package weight data.
- Mean percentage error between USDA size and Nielsen size: 1,973.43%; median percentage error: 0.03%.
- Mean percentage absolute error between USDA size and Nielsen size: 1,977.07%; median percentage absolute error: 0.04%.
- Number of items with less than 5% absolute error: 352 out of 367 (95.91%).
- Number of items with 0% absolute error: 55 out of 367 (14.99%).
- Number of UPCs with USDA package weight data: 7,673 out of 2,269,796 purchases (0.34%)

In summary, only a small portion of soft drink products are successfully matched with Nielsen data by UPC. However, we can see that the quality of the quantity for matched items are very high, with some exceptions that have very high error rates. Therefore, two major issues here:

- Small number of matched products
- Outliers in errors of matched quantity

To address the first issue, this could be due to the variation in the format of UPCs in USDA and Nielsen datasets. Currently, I only did a simple normalization of UPCs in both datasets by adding zeros in front of UPCs not of length 14, which enables about 300 products matched with weights. However, in Nielsen dataset, the UPCs are cleaned and normalized to 12 digits and excludes the 'check digits', which required some math and formula to recover; the UPCs in USDA dataset is much messier and of various length, and some have check digit and some do not. This will require more efforts to deal with. Some documentation of UPCs in Nielsen here in Section 3.2.

For the second issue, there are three potential causes. The first is incorrect unit size in Nielsen data, as we can see in the first row of Table 1 where the unit size is only 0.007 (actually the unit as CT is also unusual). In this case, we can only trust USDA. The second cause is the multi-packing issue in USDA dataset. The package size in Nielsen is always container size $\times$ number in a pack (`multi` column in Nielsen data); this is also usually the case in USDA dataset, but not always (as in row 3 to 7 of Table 1). This multi-packing issue could be related to the third cause, which is the versioning of UPC. Product information can be updated for the same UPC over time, and this aspect is well recorded in Nielsen. In USDA dataset, there is a time dimension showing when an UPC was available and modified. However, my current understanding is that the dataset might only record the product information of the last modification, so I am not sure if we can check the historic product information of the same UPC in USDA as we can do it in Nielsen.

More details about USDA Branded Food Dataset can be found here. It turns out that Nielsen is actually one of the partners that helped construct this USDA data, so ideally, with some more data wrangling and conversion we might be able to find a way to recover and align it more closely with Nielsen data.

2 Imputation Quality

The previous section account for one source of the error, which is the external dataset's quality. If the statistics for matched items in the previous section (Number of items with less than 5% absolute error: 352 out of 367 (95.91%)) seems promising and convincing to people about the data quality, we can for now assume the error due to data quality is 0 and estimate errors from imputation. This is done by the following process:

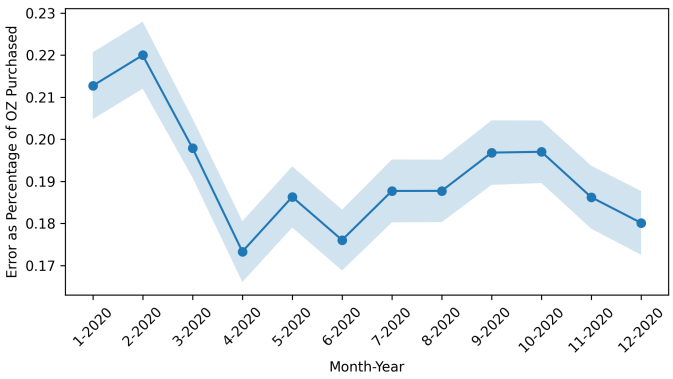
1. Take, for example, 80% of package weights from the purchase panel out of Nielsen data by random.
2. Conduct the imputation using the rest 20% of package weights as the externally matched data.

3. Compute errors and absolute errors as percentage of actual total quantity purchased for each household-month observations.

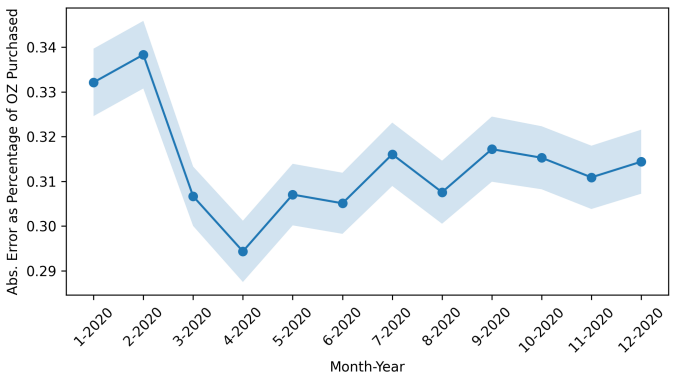
Figure 3 shows the monthly average error and absolute error as percentage of actual household-month consumption, which is around 20% and 30%

Figure 1: Monthly Average Errors as percentage of Actual household-month quantity: 80% missing data

(a) Monthly Average Errors



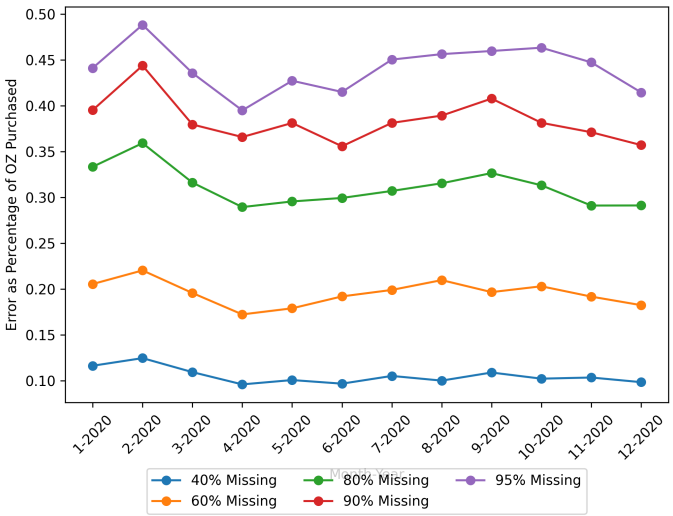
(b) Monthly Average Errors Abs. errors



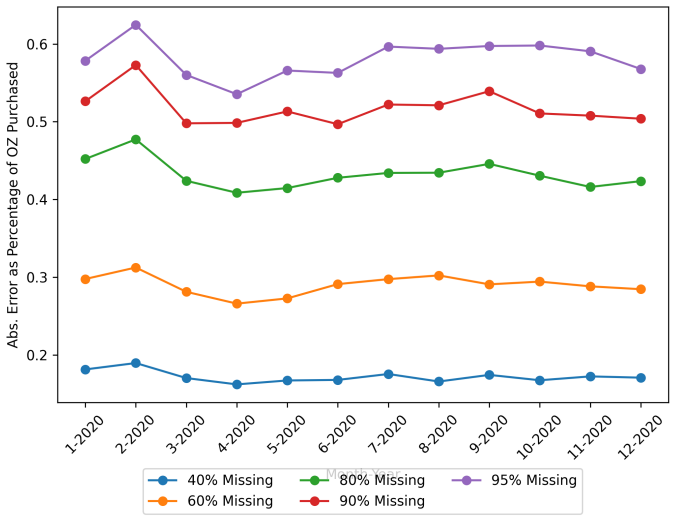
Intuitively the quality of imputation would depend on how much missing data there are. We can repeat this simulation by replace 80% with different values, as follows. The absolute error is around 60% when 95% of package weights are missing at random

Figure 2: Monthly Average Errors as percentage of Actual household-month quantity: 80% missing data

(a) Monthly Average Errors



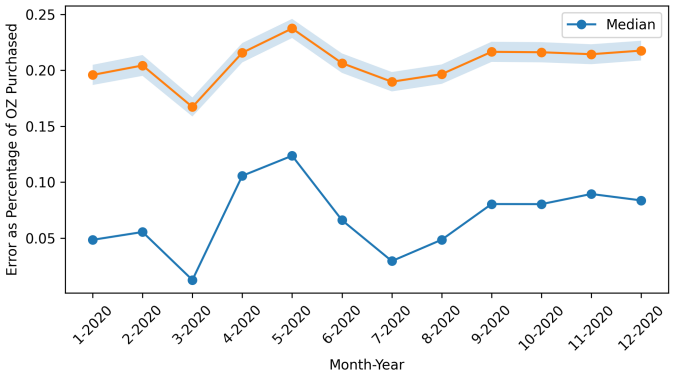
(b) Monthly Average Errors Abs. errors



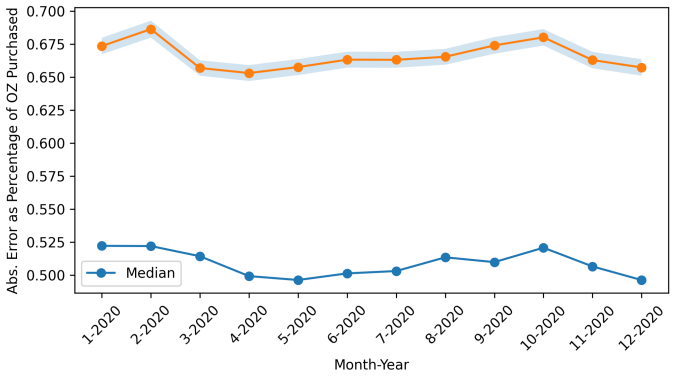
We can repeat the imputation with the current, though a small portion, matched USDA quantity to roughly approximate the data quality error + imputation error. Here I dropped extreme errors that could be due to incorrect size in Nielsen data (size being to small so divided by it blow up the error). The absolute error is around 67%.

Figure 3: Monthly Average Errors as percentage of Actual household-month quantity: USDA Matched Quantity

(a) Monthly Average Errors



(b) Monthly Average Errors Abs. errors



3 Appendix

3.1 Matched Quantity

Table 1: Soft Drink with Matched Quantity from USDA - Headers

UPC	Brand	Size	Unit	Fl oz	Multi	USDA Weight	USDA Fl oz	Abs Diff %
00857157004173	HAPI WATER	1.0	CT	0.007	1	48 fl oz / 1.5 Quart / 1.4 L	48.000	6856.14
00007203670453	FUN POPS	1.0	CT	0.007	36	90 oz / 2.55 kg	90.000	356.14
0002114061399	CTL BR DT	12.0	OZ	12.000	1	144 fl oz / 4.26 L	144.048	11.00
0002114061407	CTL BR R	12.0	OZ	12.000	1	144 fl oz / 4.26 L	144.048	11.00
0002114061401	CTL BR DT	12.0	OZ	12.000	1	144 fl oz / 4.26 L	144.048	11.00
0002362736217	CTL BR R	12.0	OZ	12.000	1	1 L / 33.8 fl oz / 1.05 Quart	33.814	1.82
0002362736218	CTL BR R	12.0	OZ	12.000	1	1 L / 33.8 fl oz / 1.05 Quart	33.814	1.82
0078000220161	SUNDROP R	12.0	OZ	12.000	12	12 fl oz	12.000	0.92
0012000000164	PEPSI R	12.0	OZ	12.000	12	12 fl oz	12.000	0.92
0078000053425	A & W DT	16.9	OZ	16.900	6	16.9 fl oz / 500 mL	16.907	0.83
0078000012194	SEVEN UP R	16.9	OZ	16.900	6	16.9 fl oz / 500 mL	16.907	0.83
0078000148482	CANADA DRY DT	16.9	OZ	16.900	6	16.9 fl oz / 500 mL	16.907	0.83
0078000122231	SEVEN UP DT	16.9	OZ	16.900	6	16.9 fl oz / 500 mL	16.907	0.83
00854652006213	TICKLE WATER DT	8.0	OZ	8.000	4	240 mL	8.115	0.75
00854652006077	TICKLE WATER DT	8.0	OZ	8.000	4	240 mL	8.115	0.75
0007789019354	CTL BR R	11.0	OZ	11.000	4	1.32 L	44.635	0.014
0007789019345	CTL BR R	11.0	OZ	11.000	4	1.32 L	44.635	0.014
00007203698611	CTL BR R	10.0	OZ	10.000	1	10 fl oz / 300 mL	10.144	0.014
00007203698610	CTL BR DT	10.0	OZ	10.000	1	10 fl oz / 300 mL	10.144	0.014
00007203698613	CTL BR R	10.0	OZ	10.000	1	10 fl oz / 300 mL	10.144	0.014

3.2 UPC Format and Conversion

Note 2: Explanation of upc_ver_uc
The characteristics associated with a UPC code sometimes change in the data. In some cases, the same UPC code can refer to completely different products. In others cases, just a few characteristics may have changed. Finally, NielsenIQ may have reclassified the product into a different module and none of the characteristics actually changed
For example, a characteristic may change due to a temporary or permanent change in a product characteristic (e.g. the size of the product changes). However, in some cases the change may simply be due to NielsenIQ-generated fields (e.g. a UPC code may have been assigned to a new Module Code).
Instead of losing information or imposing our own interpretations, we assigned a UPC version to indicate the different versions. Therefore, to merge the product characteristics into the purchase file, one should use both UPC and UPC_ver_uc as the merging variables.

20. What standard is used for the UPC variable? Why can't I merge another UPC dataset with the UPC's found in the Consumer Panel Data?

- The UPC codes (UPC and UPC_ver_uc) in the consumer panel data do not include the check digits. The codes are made up of the EAN (International Article Number), with the check digit dropped.

- In order to merge the UPCs of the consumer panel data with external data containing other formats and types of UPCs, researchers can generate the check digits. It is important to note that the consumer panel data promotes all UPC-Es (short form codes) to UPC-As, minus the check digit. See Section VIII: Other for more details about steps for merging consumer panel UPC codes with external datasets.

65. Is it possible to generate the check digit to add to the consumer panel UPC codes (UPC and UPC_ver_uc)?

- Although the check digit does not identify a product but rather to validate data entry, some external resources do retain these numbers. Therefore, if looking to compare these variables to external data, researcher s may want to compute the check digit of a UPC (UPC-A, EAN-8, or EAN-13). To do so, follow this algorithm:

  1) Starting at the right most digit excluding the check digit, sum all the digits moving left, multiplying every other number by 3.
  2) Take this result module 10.
  3) If the result is not 0, subtract from 10. This is your check digit; otherwise, 0 is your check digit.

  An example for EAN-13 7895144603049 (789514460304 with no check digit):

  1) 7+8*3+9+5*3+1+4*3+4+6*3+0+3*3+0+4*3 = 3*(4+3+6+4+5+8) + (0+0+4+1+9+7) = 111
  2) 111 mod 10 ≡ 1
  3) 10 - 1 = 9, which checks.

  A final note of concern is that EAN-8 and UPC-E may collide. While UPC-E are not stored in the consumer panel data, check with your data sources to be sure whether this may occur and how to handle it.

    - EAN-13, a 13 digit code, is a superset of UPC-A recently adopted to alleviate crowding of the space available to manufacturers to list the UPC code for an increasingly large range of products.
    - EAN-8 is an 8 digit code that serves an analogous purpose to UPC-E.

- It is possible to convert between these standards in certain circumstances, except for EAN-8. As mentioned, EAN-13 is a superset of UPC-A. You can always convert a UPC-A to EAN-13 by simply prepending a 0 to the UPC-A code. EAN-13 can only be converted to UPC-A when its first digit is 0 by dropping the 0. Be careful to account for whether your data contain a check digit to know whether the EAN-13 should be 12 or 13 digits. When stored as a numeric type, you should see no difference between an equivalent UPC-A and EAN-13 as their check digits are the same.

- Converting between UPC-A and UPC-E is also possible, however only a subset of UPC-A's can be converted to UPC-E's. Although it is possible to detect when a UPC-A is eligible to be converted to UPC-E, there is no way to know if the manufacturer used a UPC-A or UPC-E by just looking at the code. Thus, in an external data source, you may find that the product is a UPC-A or a UPC-E. Because all UPC-E's can be converted to UPC-A's, it is our recommendation that all UPC-E's be promoted to UPC-A's/EAN for maximum compatibility. In the consumer panel data, all UPC-E's are promoted to UPC-A's, minus the check digit.

- A further complication exists with regard to UPC-E's. UPC-E's are sometimes referred to as 6 digit and sometimes as 8 digit codes. The difference is that the 8 digit version includes a leading 0 and the UPC-A calculated check digit is appended.

- To convert a UPC-E to UPC-A, use the following table where the UPC-E is "abcdef" (alternatively "0abcdefC") and C is the UPC-A computed check digit:

f	UPC-A
0-2	0abf0000cdeC
3	0abc00000deC
4	0abcd00000eC
5-9	0abcde0000fC