

Ian Schwartz

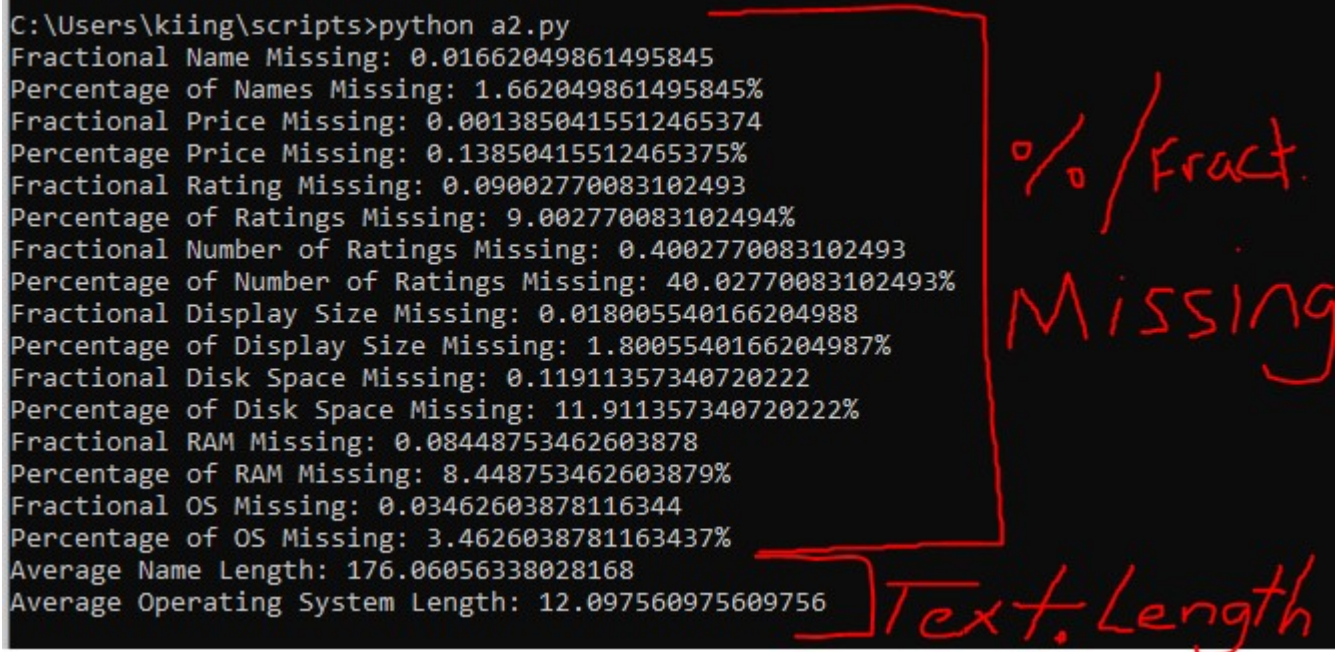
07/15/2023

COMPSCI 767 – Assignment 2 Discussion

Context: Searched 'laptop' on Amazon.com

Attributes & Classification (Set 'S'):

- Name -- -- --Textual
- Price -- -- --Numeric
- Rating -- -- --Numeric
- Number of Ratings --Numeric
- Display Size (inches) --Numeric
- Disk Size (GB) --Numeric
- RAM (GB) --Numeric
- Operating System --Textual

A terminal window showing the output of a Python script. The output lists various missing data statistics for different attributes. A red bracket on the right side of the terminal groups the first 14 lines, with a handwritten note "% / Fract. Missing" next to it. Another red bracket on the right side groups the last two lines, with a handwritten note "Text. Length" next to it.

```
C:\Users\kiing\scripts>python a2.py
Fractional Name Missing: 0.01662049861495845
Percentage of Names Missing: 1.662049861495845%
Fractional Price Missing: 0.0013850415512465374
Percentage Price Missing: 0.13850415512465375%
Fractional Rating Missing: 0.09002770083102493
Percentage of Ratings Missing: 9.002770083102494%
Fractional Number of Ratings Missing: 0.4002770083102493
Percentage of Number of Ratings Missing: 40.02770083102493%
Fractional Display Size Missing: 0.018005540166204988
Percentage of Display Size Missing: 1.8005540166204987%
Fractional Disk Space Missing: 0.11911357340720222
Percentage of Disk Space Missing: 11.911357340720222%
Fractional RAM Missing: 0.08448753462603878
Percentage of RAM Missing: 8.448753462603879%
Fractional OS Missing: 0.03462603878116344
Percentage of OS Missing: 3.4626038781163437%
Average Name Length: 176.06056338028168
Average Operating System Length: 12.097560975609756
```

^Based on the above:

Report of Average Length for Textual Attributes:

- Name – 176 character average
- Operating System – 12 character average

Missing Values (via python a2.py script):

Fractional Name Missing: 0.01662
Percentage of Names Missing: 1.662%

Fractional Price Missing: 0.00138
Percentage Price Missing: 0.1385%

Fractional Rating Missing: 0.09
Percentage of Ratings Missing: 9.00%

Fractional Number of Ratings Missing: 0.40023
Percentage of Number of Ratings Missing: 40.023%

Fractional Display Size Missing: 0.01800
Percentage of Display Size Missing: 1.800%

Fractional Disk Space Missing: 0.1191
Percentage of Disk Space Missing: 11.91%

Fractional RAM Missing: 0.08449
Percentage of RAM Missing: 8.449%

Fractional OS Missing: 0.03463
Percentage of OS Missing: 3.463%

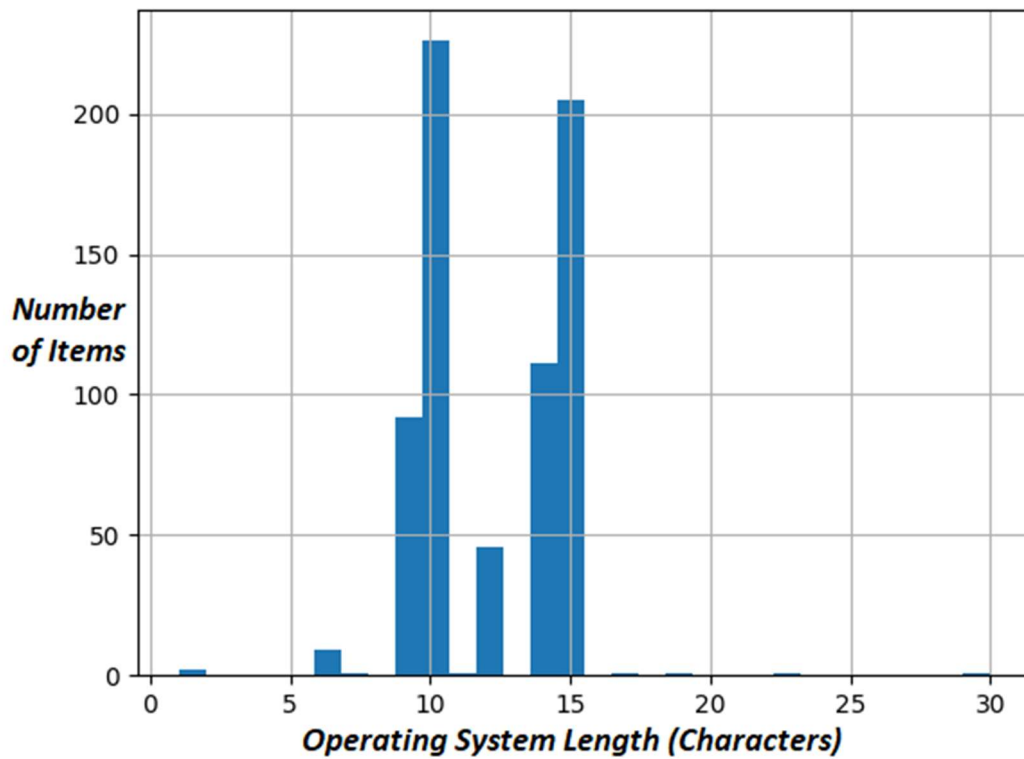
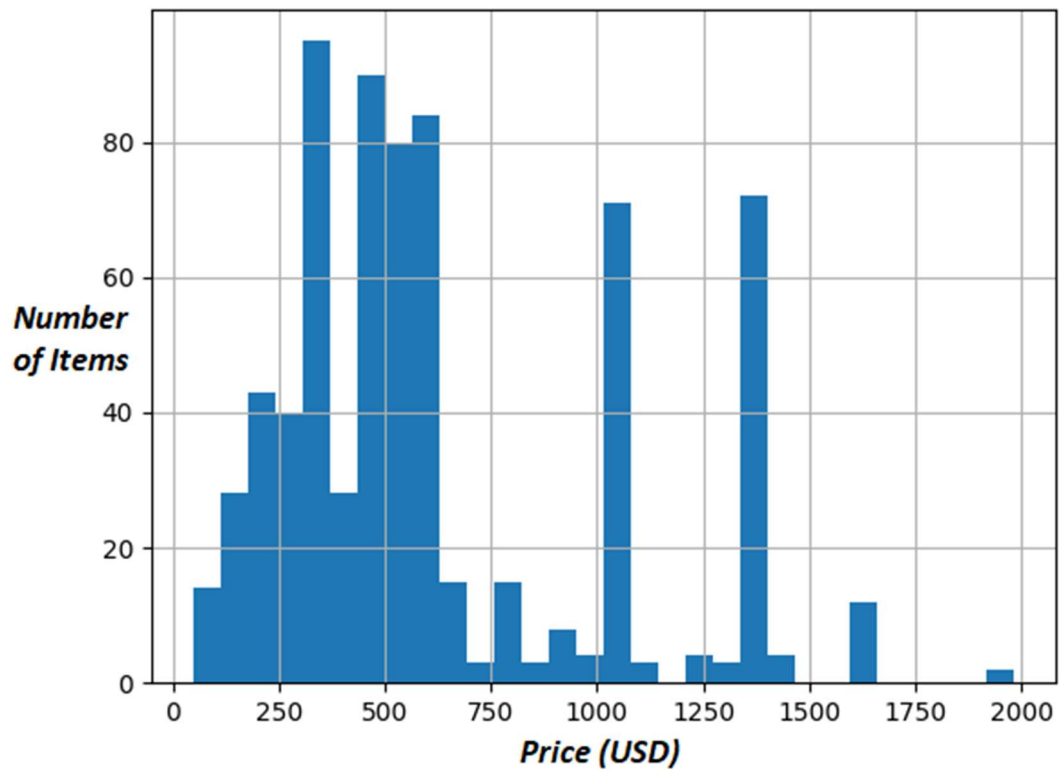
Solutions to Fill Missing Values:

There are 3 solutions I immediately think of:

1. You can scrap the information entirely – that is the entire tuple/record along with corresponding attributes for that specific item.
 - a. Pro would be more 'realistic' data overall
 - b. Con here is you lose out on data
2. You could fill in the missing value with the median or mean value corresponding to that attribute.
 - a. Pro is the inclusion of more data points/more data (specifically for the item in question)
 - b. Con is that it really isn't accurate data since it was essentially made up. This could be more accounted for if you look at the same exact item on multiple websites, because there would be a way then to include standard deviation.
3. You could simply leave the column empty, or delete whatever incorrect value (e.g. non-numeric in a numeric column), then use the percentage missing to factor into standard deviation perhaps?

If the values often have to be filled for machine learning in subsequent steps (as noted by the prompt question), then option #2 would likely be the best possible option. You wouldn't want to fill the missing value with any random value/outlier as that would skew data. As far as feasible methods for myself are concerned, I think this is the only real option. If the attribute is textual, you would have to figure out which is which. If they share the same amount of characters, you would have to develop a separate counter then, and manually look at the data to confirm.

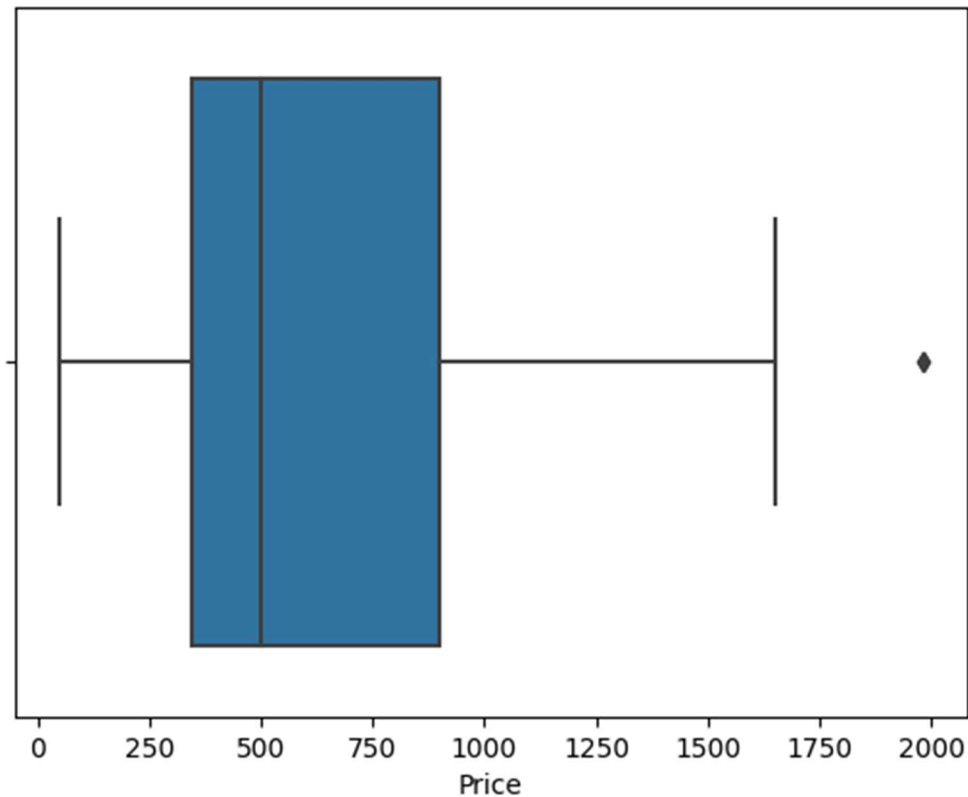
Histograms / Outliers:



Based on the above histograms, I believe the only outliers for the 'Price' category would be those above ~\$1400, as the values of ~\$1300 and ~1075 seem significantly populated, though they are still a distance from what looks like the bell curve.

Looking at the operating system length, I would say that the '6' character bar is the lowest I would go in terms of number of items before it becomes an outlier. I say this partially because I believe the '6' character bar to belong to 'Mac OS' (including the space), which is found in a fair number of items (and is a 'significant' brand so-to-speak.)

I also accidentally made a boxplot, and since I have it I figured I would include it (x-axis is Price in USD):



Other:

- I do not believe my textual attribute columns contain any synonyms. I think the closest thing to that would be the variation in Windows OS versions (Windows 10 S, Windows 11 S, Windows 11 Pro, Windows 11 Home, etc.), but since those are all technically different, I don't believe they are synonymous.
- For some of my items, I had an issue with some other value going into the incorrect column. For instance, I would find 'List: ' in my 'Price' column as well as my 'Number of Ratings' column which was quite odd. This is absolutely a problem which ran into when attempting to create a histogram for Number of Ratings (NOT pictured above), where it would give me an error essentially because there was a 'non-numeric' value it was picking up.
 - To fix this, I simply wrote a portion of my a2 script which removes non-numeric values from my 'Number of Ratings' column.
- I believe all my attributes follow a certain format (no dates or anything), minus the few values that are simply incorrect.

Software:

- Mostly just Python language for scripting, using Visual Studio Code
- Used MS Paint to quick edit in the axis titles, since my program spit the histograms out as PNG files