Ad Hoc Analysis

Overview

DigiFoods, Inc. is a Filipino start-up that offers marketing consultancy services to small to mid-sized businesses in the food and beverage industry. One of their clients is Lola Tamis, a former JSEC enterprise that has since turned into a real business with five different physical branches.

Lola Tamis contracted DigiFoods to run and analyze their loyalty program (where customers effectively exchange their purchase activity data for perks and discounts). DigiFoods only had application developers, so they hired you to handle the analysis half of this project.

Context

DigiFoods was able to collect users' transactional data for six months. DigiFoods now needs to parse some important metrics from the transactional data and present them to Lola Tamis.

Lola Tamis wants DigiFoods to describe the transactions that were captured by the loyalty program. For the most part, they do not have strict requirements: they will be satisfied with a pivot table, though they would of course be pleasantly surprised by deeper insights. At minimum, they want to have a breakdown of the count of each item sold per month, and they also want to have a breakdown of the total sale value per item per month.

Lola Tamis also wants a table whose columns are each month covered by the transactional data and whose rows are the following metrics:
1. Repeaters - the number of customers from the current month who also purchased in the previous month. This metric is 0 for the earliest month in the transactional data.
2. Inactive - the number of customers in the total set of transactions up to and including the current month who have purchase history but do not have a purchase for the current month. This metric is 0 for the earliest month in the transactional data.

3. Engaged - the number of customers in the total set of transactions up to and including the current month who have consistently purchased every single month.

Lola Tamis thinks that there may be more useful activity metrics in addition to these, but they did not provide DigiFoods with a brief on their additional thoughts.

One of your programmer friends notes that the "grain" of the transaction table may be problematic. The data of Lola Tamis appears to be presented in a granularity of one transaction per row. She notes that it will be much easier to work with the data if it is first transformed into a granularity of one "line item" per row. (For example, if a transaction contains two line items "Red Beans,10" and "Green Beans,5", then each line item should be its own row. They should not be bundled into one row.)

Requirements

DigiFood gave you this single [transaction file](#) to work with.

DigiFood requires you to use the Python data ecosystem (NumPy, Pandas, Matplotlib, etc.) with minimal [hardcoding](#) to generate the reports. They foresee that they will be able to do similar analyses for other, much larger clients. They want your work to serve as a reference artifact for these larger jobs, which cannot be done through spreadsheets. As such, they also need you to use the Python data ecosystem appropriately (i.e., using vectorized methods instead of raw Python for loops as much as possible).

Deliverables

There are three core deliverables for this test. Be concise.

1. A one-page document explaining your approach to the problem. Use font size 12, 1.5 spacing.
2. A short deck containing your visuals and tables.
3. A public GitHub repository containing all the code used to generate the visuals and tables. For your benefit, you may want to include Jupyter notebooks with fully-rendered charts and tables to ensure that I see your results.

Supporting documentation is welcome but not necessary and will not be used as a basis for grading.

Include a properly accomplished [Certificate of Authorship](#).

Standards

Performance	Multiplier (divided by 4)	Description
A	4	Provided an output that is industry-ready and fulfills the spirit of the requirements almost perfectly. The output accomplishes requirements in a way that is justifiably better than what the client suggested. The methods used made full use of the characteristics of the Python data ecosystem.
B+	3.5	Provided an output that has some enhancements over the stated requirements. The output directly tackles some of the desired improvements over the minimum requirements discussed by the client. The methods used were appropriate uses of the Python data ecosystem.
B	3	Provided what was required. Nothing more, nothing less. The methods used were acceptable.
C+	2.5	Provided most of what was required with some minor deficiencies. The output fulfills most of the functional requirements of the client, but it suffers from some bugs and/or problems. The methods used were questionable.
C	2	Provided an output with major deficiencies. The output has difficulty fulfilling the requirements of the client, and in some cases, the output cannot fulfill the requirements at all. The methods used treated the Python data ecosystem tools as mere Python lists and dictionaries.
D	1	Provided an output that, despite the programmer's best efforts, cannot be construed as a submission to this project.
F	0	No submission by the hard deadline and/or provided an output that does not show any level of effort from the programmer at all.