

Coding_Exercise

September 30, 2019

1 Data

The data file `example_data.csv` contains realistic but fictional rows of bank transaction data for a single individual. The data is described by 5 columns with the following definitions:

1. `user_id`: UUID for the individual's bank account
2. `made_on`: date the transaction was registered in the database
3. `original_description`: transaction description provided by the bank
4. `amount_cents`: transaction amount in cents
5. `income`: whether the transaction is credit or debit

2 Problem Statement

A person who uses their bank account actively will have several regularly recurring transactions, due to bill payments, salary payments, subscriptions or habits in purchasing behavior. The objective is to create code that takes bank transaction data as input and returns a data structure which summarizes a single user's regularly recurring transactions as output. The definition of a regularly recurring transaction is any debit or credit transaction to or from the same entity that happens at regular intervals. An example of the desired output of the final code is shown below:

```
In [13]: recurring = get_recurring_transactions(transactions)
         recurring

[
  {
    "entity_name": "Fake Company Salary",
    "income": true,
    "period_days": 28.0,
    "typical_amount_cents": 85000
  },
  {
    "entity_name": "Phone Subscription",
    "income": false,
    "period_days": 27.5,
    "typical_amount_cents": 3499
  },
  {
```

```
    "entity_name": "Weekly Friday Mcdonalds",
    "income": false,
    "period_days": 7.0,
    "typical_amount_cents": 479
  }
]
```

The `entity_name` is the common description of the regularly occurring transactions, `income` again refers to whether the transaction is debit or credit, `period_days` denotes the expected interval in days between consecutive transactions for that entity, and `typical_amount_cents` denotes the typical amount in cents. The `example_data.csv` is provided as a sample for you to develop your understanding of the problem and test out your solution, however it is expected that the solution will generalize to unseen data (i.e. same format data for a new user).

3 Guidelines

There is no single right or wrong answer to this problem. A good solution is the one that attempts to return the correct output as outlined above, and clearly discusses the process along the way, clearly outlining any assumptions you are making. Present your solution using an ipython jupyter notebook with markdown text and code comments to explain your reasoning/methodology. You may use any python packages that you think will aid the solution, but if they are outside of the normal `numpy/scipy/sklearn` please give a short description of them and reference where they can be downloaded.