Case Study Data Science / ML Engineer

Context

Within the world of financial transactions, it's very likely that data contains re-occurring behaviours and patterns. After all, transactions between parties contains mostly the same data:

- As consumer, you buy goods within similar categories and in similar price points
- As webshop owner, you sell a fixed inventory that slowly changes over time as you build your business
- Between businesses, monthly subscriptions get processed

If we would capture this data using rule-based logic, the task would be very hard and error prone. The solution would use static logic and thresholds, but the underlying data can drift, the thresholds are very specific for each use-case, and this will result in a high number of false positives.
Machine Learning models have the potential on various fronts to solve some of these problems. Given that the input data for fraudulent transactions is highly skewed towards transactions being non-fraudulent, but some fraudulent patterns can be described in a very obvious way, there are models that can help spot fraudulent transactions and greatly improve the accuracy of the monitoring process.

Assignment: "Work on a solution that tries to find fraudulent transactions in the given data set"

The data that you will be given comes from a simulation called AMLSim [1]. It is a simulation of financial transactions with known money laundering patterns embedded. The AMLSim project is intended to provide a multi-agent based simulator that generates synthetic banking transaction data together with a set of known money laundering patterns - mainly for the purpose of testing machine learning models and graph algorithms.

There are two data sets available: one with 10.000 involved accounts, and one with 100.000 involved accounts. More involved accounts mean a larger data set is generated. You are free to choose which data set you want to choose, they are both included in the assignment.

[1] https://github.com/IBM/AMLSim
[2] https://github.com/IBM/AMLSim/wiki

Assignment approach

Since this is a case study, and to keep the time requirement realistic, we don't require you to provide a full end-to-end implementation. What you should think about as a case study, is to provide us with information around your process in a way that is explainable and easy to follow.

To this end we have prepared a Jupyter Notebook that uses the PyCaret library (https://pycaret.gitbook.io/docs) to get you started with the analysis and model building.

There is no "correct" approach for the case study, but we think we can broadly define three ways of approaching this problem, depending on your strengths as engineer:

1. Data science: Focus on data analysis, data quality, statistics, emerging patterns in the data that we can already identify, data completeness, and interpretation of the existing models

2. Data engineering: Orient around feature engineering. You can conceivably create more relevant features for the given data set than what we currently include in our example. Try to construct certain features that enhance the expressive power of our models.

3. Machine Learning: Focus the actual technical implementation, which models are suitable for the task, on which data, but disregarding some context around it like model lifecycle, all possible models that could work, and data analysis

When discussing the case with us, depending on which of the above formats you choose, we will be looking for information around specific topics such as:

- What did your initial data analysis look like, what did you learn about the problem from looking into its context?
- Did you do any exploration in terms of data quality and/or distributions, and what does this mean for the features that could be extracted from the data?
- What did your consideration for choosing some model architecture look like? There are many models available, how did you narrow down to a specific set of models and what prompted you to select this/these model(s)?
- How did you implement your model (which libraries etc) and how did you evaluate the outcome of the model?
- What is your thought process about deploying this model in production, collecting metrics, maintaining its lifecycle?

Hand-in format

You are expected to program your solution with Python as your main programming language. Feel free to use other auxiliary scripting or programming languages, if the main language being used is Python.

You'll hand in the results of the analysis in human-readable format. This can be only on the engineering level, using for instance a Jupyter Notebook to document your analysis steps, which you then walk us through. It can also be a short slide-deck with a summary of your findings. Both options are fine, and a matter of preference.
When you use other libraries to implement something, it's greatly appreciated if you share a Docker setup so we can reproduce your build. Also, when you hand in your work, feel free to use a Git repository in your Github account or something similar.

On the day of the assessment interview, you'll present your findings to us in this way, so make sure the format is interpretable.

Data structure

The Github page for AMLSim contains a clear instruction on the CSV schemas of the relevant output files. A whole data dump is generated for you in these two data sets, which also include analysis images and auxiliary CSV files for potential use.

https://github.com/IBM/AMLSim/wiki/Data-Schema-for-Input-Parameters-and-Generated-Data-Set#output-data-schema-definition

Questions / remarks?

If you have any questions or remarks about the assignment, don't hesitate to contact us. We encourage discussion over incorrect implementation ☺