Data Science Engineering Exercise

At Klaviyo, our mission is to improve every human interaction via data science. In the last decade, the amount of data we have access to has exploded. And yet, we aren't using that data to its full potential -- data is difficult to access, it's not easily queryable, it's not easy to get insights from. The promise of data and data science is its ability to help us make better decisions leveraging computing power and storage that's beyond the capacity of our individual brains.

We have a lot of data. Our customers send us billions of events representing user and customer behaviors (e.g. logging into an app, buying something, visiting a website, etc.) a month via our API that we validate, normalize and store in realtime. We store and aggregate data in a datastore that allows us to execute low latency queries with high throughput -- so asking questions is fast. On top of that datastore, we're now starting to build algorithms to test hypotheses and discover patterns in the data.

For this exercise, we're going to consider the case of predicting a user or customer's future actions. We'll use making a purchase as the action, however the action to predict could be anything -- user sessions on a website, attending an event, etc.

The action we want to predict in this exercise is if a customer will make a purchase within the next six months. More precisely, for every customer of a company, our model should output the probability that the customer will make at least one purchase in the next six months. We want to be able to evaluate the accuracy of this prediction.

Your goal is **not** to create a model that this prediction. **Rather, your goal is to create a framework to evaluate models**.

Data

The attached zip file contains three CSV files. Each CSV corresponds to a single company and lists transactions over time. Each CSV has the following format:

CustomerID	**Timestamp**	**PurchaseValue**
hash_value	2018-05-31 23:25:47+00:00	100.00

The first column is the **CustomerID** and it's possible to have multiple rows with the same **CustomerID**. The second column is a UTC timestamp and the third column is the monetary value of the purchase. You can assume that each file contains a comprehensive list of transactions from the minimum to maximum dated transactions in the file.

What You Build

You will create a framework to evaluate models. Think of the user of your system as an analyst who wants to know how well a model works and compare different models.

Specifically, we want you to build:

- A small python framework that will allow evaluating models using the provided data.

- Two dummy models that predict the probability a customer makes a purchase in the next six months. (These models do not need to perform in any meaningful way. Their only purpose is to test and demonstrate your framework.)

- A Jupyter notebook that pulls everything together and shows evaluating your dummy models on the provided data using your framework.

We also want you to address (but not implement!) the following question in your notebook:

- How would you approach the problem if you had data from 100,000 companies rather than three companies?

Send us your code and notebook.

Purpose

We would like to see you achieve the following:

- Come up with a useful and trustworthy way to evaluate model performance. In other words, an organization should be able to use your framework and make an informed decision about which model to choose and what to expect from that model. This is getting at your ability to think like a data scientist. However, since this is an engineering exercise, we're expecting a common sense approach rather than any particular statistical solution.

- Develop a simple and clean framework that will be easy to extend to more models and datasets. This is getting at your engineering skills and ability to write high-quality code.

Again, we are not asking you to come up with an actual reasonable model.