Homework Assignment 2

"ML Fundamentals & Implementation"

Course: Machine Learning in Marketing

Due date: 06/12/2020

Part 1 Machine learning metrics.

Create your own implementations for the following four metrics:

- ROC-AUC score (Area Under the Receiver Operating Characteristic Curve)
- Binary cross-entropy
- Silhouette score
- R2 score

Do *not* use existing libraries that contain metric implementations (e.g., `sklearn`). The idea is that you implement solutions "manually," using only `pandas` and/or `numpy`. The notebook `02-metrics.ipynb` contains code that creates data you can use for testing your implementation.

For each metric, (1) briefly explain in a few sentences what the metric evaluates, and (2) suggest two suitable applications in a marketing context.

Bonus: Create a Python package for your metrics functions. Open a pull request with your code on Github (folder `(mlim)/exercises/solutions/$GROUP/exercise-2`).

Part 2 Clustering.

Implement a pipeline that clusters Instacart customers based on the data you observed about their past purchases. Use the Instacart data set from Exercise 1 (Part 3).

Start with exploratory data analysis (EDA). Then derive features from the raw data sets. For each feature, explain what motivated you to build the feature (e.g., why do you believe that the feature is useful for distinguishing customers).

When implementing the clustering pipeline, follow the best practices we discussed in lectures/exercises 03 and 04. For example, structure your code into pipeline stages, use a configuration file, tune hyperparameters, etc. Compute meaningful metrics that evaluate the performance of your clustering implementation.

Bonus: Study the clusters. Can you find (statistically) significant differences between the customer groups?

Part 3 Predict time of customers' next order.

In exercise 03 we looked at Dr. D's grocery store. Remember that Dr. D hired the consulting firm Accidenture to help him cluster customers. Now he wants to give you a chance to win his next ML contract.

Dr. D owns another store, a hypermarket, that stocks almost 50,000 products. Dr. D wants to run a marketing campaign that targets customers who visit his shop infrequently. As a first step he needs a machine learning implementation that predicts whether a customer will visit the shop again within the 14 days after the last purchase. Dr. D then plans to provide a 10% discount (applied to the basket amount at the checkout) to customers who will not visit the store in the 14 days after their last purchase. The discount can be redeemed if customers spend at least 25 Euros on their shopping trip. Printing and sending the coupon to customers costs 1 Euro. Dr. D tasked you to develop the ML model that helps him predict who will not visit his shop in the next 14 days.

Before Dr. D deploys your solution to production, you need to provide predictions that he will benchmark (using the AUC metric). The data are available on Moodle:

`orders.parquet`	Information about orders.
`baskets.parquet`	Products purchased in each order.
`aisles.parquet`	Products' aisles: IDs and descriptions.
`departments.parquet`	Products' departments: IDs and descriptions.
`products.parquet`	Product master: IDs, descriptions, links to aisles and departments.
`prediction_index.parquet`	See below.

The file `prediction_index.parquet` serves as input for making your predictions. Your task is to predict the ***probabilities*** that the customers will make a purchase in the next 14 days for all customer-order combinations.

user_id	order_number	yhat
1	11	?
2	15	?
3	13	?
4	6	?
5	5	?

Upload your solution (`e02-$GROUP.parquet`) on Moodle (Assignment 02) by noon, Dec 6. The leaderboard will be published at the end of the exercise on Dec 7.

Bonus: Can you think of a metric for evaluating the quality of your models that is better suited for Dr. D's goals than AUC?