Introduction

A marketplace is being attacked by bots that produce fake clicks and leads. The marketplace reputation might be affected if sellers get tons of fake leads and receive spam from bots. On top of that, these bots introduce noise to our models in production that rely on user behavioural data. We need to save Adevinta's reputation detecting these fake users.

To do so, we have a dataset of logs of a span of five minutes. Each entry contains the user id (**UserId**), the action that a user made (**Event**), the category it interacted with (**Category**) and a column (**Fake**) indicating if that user is fake (1 is fake, 0 is a real user). An example of how the data looks like:

UserId	Event	Category	Fake
XE321R	click_ad	Motor	1
ZE458P	send_email	Motor	0
XE321R	click_ad	Motor	1

An internal team in Adevinta studied the behaviour of these bots and concluded that:

- A bot tends to produce more clicks in an a window period of time than a real user
- While real users interact with just a small subset of categories in a window time, bots behaviour are not centered into specific categories and interact with a vast majority of them.
- The distribution of clicks for a given bot tends to be skewed towards clicks like *click_ad* and *send_email*, because the former can bias our algorithms and the latter aims to annoy our sellers.

Instructions

- You will find with this email a dataset (csv.file)
- Your task is to create code that :
    - takes as input another .csv file with the same structure (but without the “Fake” columns)
    - outputs a .csv file with 2 columns : “UserId” and “is_fake_probability”
    - recommend a threshold to use for classification
- Programming language : Python is recommended, Scala is accepted
- Please submit a running piece of code, documented as you please (in the code, separate docs, document by tests, whatever makes the code understandable, exploration code)
- We are interested in 2 main aspects, in order of importance :

    - Production code quality : project structure, tests... The code should also run on different platforms (Linux, macOS, …)
    - ML knowledge : feature engineering, model training and evaluation. You can use any ML library you want. We are not looking for cutting edge algorithms, don’t worry if the results are sub-optimal
- If you have ideas on how to improve the solution further, please write that down as well
- Spend approximately 3 hours on the task

File provided :
https://github.mpi-internal.com/jordi-esteve/ml-academy-ds/blob/master/Datasets/fake_users.csv