Целью этой работы является переход от **unsupervised DL** -> к -> **supervised DL**, а именно – ***предсказание вероятности для каждого клиента быть распознанным как мошенник***.

Вы помните что output нашего проекта на SOM являлась матрица *frauds*, содержавшая тех клиентов которые принадлежали категории ***outliers*** (выделяющихся), соответственно имеющих вероятность быть распознанными как мошенники.

Чего не было в нашем проекте на SOM так это **предсказывания** того что клиент может оказаться мошенником. Поскольку данная проблема требует предиктивной модели, мы стремимся перевести наш unsupervised model в supervised model. Также давайте определимся с тем что по итогам проекта SOM у нас есть **binary output** – fraud or NOT fraud. В данной же работе нужно вывести **вероятность** быть распознанным как мошенник, соответственно это ….? Определите тип вашей проблемы выбрав одно из двух:

1. Регрессия
2. Классификация

Итак мы определили нашу проблему, что нам нужно сделать далее? Определить нашу матрицу характеристик. Назовем ее *customers*. Напомню, что до этого наш ***dataset*** был поделен на **X** and **y**. Где **'X'** содержал часть датасета на котором мы построили SOM, а '**y'** содержал информацию о том была ли одобрена кредитка клиенту или нет. Мы сознательно разделили данный dataset подобным образом. Теперь же нам нужно поделить наш dataset иначе. Ваша задача:

Определите какие характеристики будут важны для нашей предиктивной модели, а какие нет. Оставьте только нужные в customers. Вашим новым датасетом будет матрица customers размером 690x15.

В supervised DL есть dependent variable and independent variables. Зависимые переменные это переменные исход которых зависит от одной или совокупности нескольких других переменных. Например вероятность того мошенник клиент или нет может быть зависима от нескольких других переменных:

$$P(f) = \{A1, A2, A3 \dots An\}$$

P(f) в данной формуле – probability of being fraud. Сет *{A1, A2, A3 … An}* это независимые друг от друга переменные. В данном случае P(f) зависит от нашего сета.

Ваша задача:

Создайте зависимую переменную в виде вектора вероятностей для всех 690 клиентов. Назовите ее is_fraud (Figure 1).

Подсказки:

1. *Создайте вектор из нулей с помощью numpy*
2. *С помощью цикла итеррируйте через dataset и найдите id, а затем индекс в массиве тех клиентов которые были переведены в список frauds.*
3. *С помощью индекса обновите соответствующий элемент вектора is_fraud*

Следующим этапом станет соединение SOM с ANN. У вас уже есть все необходимое для этого. Исходный код для ANN возьмите в соответствующей работе из списка предыдущих.

Подсказки:

1. *Уберите ненужные части (все до масштабирования данных)*
2. *Не забудьте поменять входные данные (X_train на свои новые данные)*
3. *Уберите X_test из масштабирования.*
4. *Архитектура будет та же, хотя число слоев, как и другие характеристики можно уменьшить. Попробуйте подобрать. Accuracy будет ~ 0.97 , loss ~ 0.19.*
5. *Делайте prediction на опять же своих новых данных, не X_test.*
6. *В конце у вас будет вектор из вероятностей, соедините (concatenate) его с вектором client IDs (Figure 2).*
7. *Отсортируйте по возрастанию*

is_fraud - Массив NumPy

	0
39	0
40	0
41	0
42	1
43	0
44	0
45	0
46	0
47	0
48	0
49	0
50	0
51	0
52	0
53	0
54	1
55	0
56	0

Figure 1.

y_pred - Массив NumPy

	0	1
0	15608916.…	0.008180
1	15600975.…	0.008880
2	15611973.…	0.009246
3	15611409.…	0.010099
4	15752344.…	0.010280
5	15688059.…	0.011422
6	15735106.…	0.012152
7	15686670.…	0.012392
8	15643056.…	0.012784
9	15801473.…	0.012932
10	15593959.…	0.013019

Figure 2.