1. Методами машинного обучения (не статистическими тестами) показать, что разбиение на трейн и тест репрезентативно.

Ответ: Т.к. разбиение на train и на test уже зафиксировано, то данное разбиение репрезентативно, если целевая переменная имеет такое же распределение для train и test.

2. Есть кластеризованный датасет на 4 кластера (1, 2, 3, 4). Бизнес аналитики посчитали, что самым прибыльным является кластер 2. Каждый клиент представлен в виде 10-тимертного вектора, где первые 6 значений транзакции, а оставшиеся: возраст, пол, социальный статус (женат (замужем)/неженат (не замужем)), количество детей. Нужно поставить задачу оптимизации для каждого клиента не из кластера 2 так, чтобы увидеть как должен начать вести себя клиент, чтобы перейти в кластер 2.

Задача оптимизации для каждого клиента не из кластера 2 будет заключаться в нахождении оптимальных значений для его вектора с целью максимизации его прибыли и перехода в кластер 2.

Математическая постановка задачи оптимизации: Пусть у нас есть клиент $i, i = 1, 2, \dots, N$, где N - общее количество клиентов в датасете, и x - его 10-мертный вектор, состоящий из транзакций ($x_1, x_2, \dots, x_6$), возраста (x_7), пола (x_8), социального статуса (x_9) и количества детей (x_{10}).

В первую очередь скажем, что последние 4 параметра не стоит как-то менять потому что конкретный клиент физически не станет старше или младше, не сможет увеличить или уменьшить число детей, сменить пол и т.п. Т.е. нас интересуют только данные о транзациях.

Так как мы используем метод kmeans, то мы знаем центры каждого кластера. Таким образом если мы хотим чтобы клиент имел наибольшую выгоду для нас, то нужно чтобы **точка с признаками клиента в гиперпростанстве была максимально близка к центру 2 кластера**

Тогда задача оптимизации для каждого клиента i можно сформулировать следующим образом:

$$||x - y|| - \lambda||y - c_2|| \to \min_{y, x_1, \dots, x_6}$$

λ>0- гиперпараметр, который контролирует баланс между притяжением к кластеру 2 и отталкиванием от кластеров 1, 3 и 4
c_2 - это центр второго кластера.

- $f(y) = 2$ (т.е. y находится во втором кластере)

Тут y - точка второго кластера, которая также является переменной, по которой мы оптимизируем. Для того, чтобы мы стягивали x к границам кластера 2 мы вычитаем такую норму, чтобы y был как можно дальше от центра кластера 2, при этом сам кластер не покидал. Обратим внимание, что при решении задачи оптимизации мы учитываем лишь шесть переменных - параметров транзакции, т.е. таким образом мы не заставляем клиента менять пол, количество детей и т.п.

Всего таких задач оптимизации ставим $N - |I_2|$ штук. Т.е. всего наблюдений без учета наблюдений из кластера 2.

3. Что лучше 2 модели случайного леса по 500 деревьев или одна на 1000, при условии, что ВСЕ параметры кроме количества деревьев одинаковы?

Ответ: Так как на случайный лес влияет случайность, то при одинаковых random_state случайный лес с 1000 деревьями будет лучше, так как 2 одинаковых леса по 500 усредненные между собой - это по сути один лес по 500. Если предположить, что при увеличении количества деревьев качество растет, то 1000 деревьев будет лучше.

4. В наличии датасет с данными по дефолту клиентов. Как, имея в инструментарии только алгоритм kmeans получить вероятность дефолта нового клиента.

Ответ: Имея в инструментарии kmeans мы имеем возможность считать расстояния между точками. При появлении новой точки в пространстве признаков мы можем найти расстояния от нее до других точек и выбрать k ближайших расстояний. Так как наши данные разделены на два класса алгоритмом kmeans (был дефолт/не было дефолта), то мы можем посчитать:

$$prob = \frac{with\_default}{k}$$

Где $with\_default$ - количество точек из k ближайших у которых класс "был дефолт".
Для большей справедливости можно придумать какую-то взвешанную сумму, выбирая веса обратно пропорциональные расстоянию. Но тут прийдется каждый раз пересчитывать коэффициент перед суммой чтобы при условии что $k = with\_default$ вероятность была равна 1.

В случае если класетров много, можно считать такие вероятности принажлежности к каждому кластеру. Т.е. считать число объектов среди k ближайших отнесенных к каждому кластеру и дальше делать $argmax$ чтобы найти кластер имеющий наибольшее

количество ближайших точек среди k выбранных .

5. Есть выборка клиентов с заявкой на кредитный продукт. Датасет состоит из персональных данных: возраст, пол и т.д. Необходимо предсказывать доход клиента, который представляет собой непрерывные данные, но сделать это нужно используя только модель классификации.

Ответ: На самом деле часто возникает потребность оценивать непрерывные величины задачами классификации. Если визуализировать таргет мы увидим, что у нас образуются большие кластеры на определенных уровнях зарплаты, например, скорее всего довольно большой кластер будут образовывать люди с з.п. 10к, 20к, 30к, 40к, ... Т.е каждые 10 тысяч рублей. Таким образом вместо задачи регрессии мы можем поставить задачу многоклассовой классификации, рассматривая 50 классов. с зарплатами от 10к до 500к. Более высокие зарплаты являются редкостью и скорее всего будут выбросами в нашей задаче. Далее обучаем модель классификации, например, градиентный бустинг на деревьях решений (например таким инструментом пользуются специалисты из компании ЦФТ).

Данный подход можно считать некоторой аппроксимацей непрерывному пдходу регрессии. Скорее всего точность у такого подходу будет хуже, но в каком-то смысле так мы сможем решить задачу с приемлемой точностью