Вам дана таблица секвенирования ДНК набора пациентов. Положения в секвенированном сегменте ДНК помечены как метилированные («1») или неметилированные («0»).
Однако метилирование ДНК может различаться для одного и того же материала от одного и того же пациента: первая позиция иногда может быть измерена как метилированная, а иногда нет.
Для каждого пациента мы предоставляем вам данные о том, сколько раз мы сталкивались с определенным паттерном метилирования.
Каждый пациент также помечен как больной раком (у них есть префикс `p_cncr`) или пациент контрольной группы (у них есть префикс `p_cntrl`).

Задача минимум (15 баллов) проста:
1. Для каждого пациента создать базу данных с 20 переменными, каждая из которых содержит долю случаев метилирования определенной позиции в ДНК (должны получиться числа от 0 до 1);
2. Также добавьте 20 * 19 * 4 / 2 = 760 переменных для частот того, как часто две определенные позиции устанавливаются в шаблон "11", "01", "10" или "00".
3. Возьмите логарифм от полученных значений.

Основное задание (20 баллов) состоит в проведении следующих операций:

[0] Для нужд этого задания требуется взять логарифм от значений, найденных в предыдущем задании.

[1] В этом задании от вас потребуется реализовать функцию, принимающую на вход набор переменных и считающую ни их основе точность какой-либо модели-классификатора для различения здоровых и больных раком пациентов.

- Для подсчета точности следует использовать leave-one-out cross validation: вы обучаете модель на всех данных кроме одного пациента и проверяете корректность её предсказания на этом пациенте. Эта процедура повторяется для каждого пациента отдельно, после чего возвращается доля пациентов, статус (болен/не болен) которых был корректно угадан.
- В качестве модели-классификатора вы можете использовать любой разумный метод, но рекомендуется использовать логистическую регрессию. В R она считается функцией `glm()` с параметром `family = "binomial"`, а в python можно использовать следующий пакет: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
- Не обязательно передавать в функцию именно данные переменных. Вы можете отдельно создать глобальный dataframe, а функции передавать лишь названия/метки/позиции переменных.

[2] Напишите функцию, проверяющую точность предсказания (функцией из

пункта 1) для всех пар двойных переменных, посчитанных в предыдущем задании.
Эта функция должна уметь распаралеливать данную работу на количество потоков, переданное ей в качестве аргумента.
Должна быть возможность продемонстрировать разницу во времени выполнения при разном количестве используемых потоков (здесь, ради экономии времени, допускается использование не всех пар двойных переменных).

Дополнительное задание (25 баллов):

Допустим замер произвольных мест в днк является слишком дорогостоящей процедурой. Но пусть дана следующая последовательность днк:
GTGATTGTGGGCACAGCTACAAAAC1GGGGTTGGATAAGTC2GCTC3GCC4GGGGCC5GAGGGCC6GTCTC7GTGC8GGGGGC9GGGGAAGGGGC10GTGAGGC11GC12GGAGATGG13GAGAAAAC14GCTAACCC15GC16GTTCTTGATGGGAGGCC17GC18GTCC19TGGGAGATGGGGGT20AGC
Если здесь после буквы стоит число - то эта буква днк соответствует очередной из наших 20 позиций. Остальные буквы сами по себе нас не интересуют. Сами цифры никаких позиций не занимают и лишь маркируют определенные буквы.
Пусть также для нас является "дешёвым" измерить сразу 3 "куска" этой последовательности по следующим правилам:

- Каждый кусок должен иметь длину между 17 и 22 буквами включительно;
- Между двумя разными кусками должен быть зазор в хотя бы 2 буквы.

Допустим мы выбрали куски "GTGATTGTGGGCACAGCTACA", "C1GGGGTTGGATAAGTC2GC" и "C18GTCC19TGGGAGATGGGGGT20AGC" - в них присутствуют позиции 1, 2, 18, 19 и 20. Это значит что мы можем "дешево" измерить переменную, составленную при фиксировании этих 5-и позиций. Так как каждая позиция бможет быть метилированной или нет, то на основе 5-и позций мы получаем 2^5 = 32 разных переменных.

Найдите всё переменные, которые можно "дешево" посчитать на основе указанных правил. Сколько их (необходимо чтобы по ним у вас были хоть какие-то данные)? Используя функцию из пункта 1, найдите переменную, на основе которой можно делать наилучшие предсказания среди переменных, составленных из 6-и и менее фиксированных мест.