# Группировка фото

В процессе изучения новой модели, используемой для верификации лиц, наши разработчики столкнулись с необходимостью понять насколько эта модель хорошо работает. Для этого из интернета был скачан большой набор из N фотографий, но проблема в том, что все эти фото не размечены. Другими словами, не указано какая фото к какой личности относиться. Для дальнейшего анализа, было решено сначала разобраться с выборкой фотографий и просматривать все пары фотографий, чтобы их разметить, очень долго. Такой подход будет требовать O(N2) просмотров человеком. Поэтому было решено воспользоваться моделью для группировки фотографий в подвыборки (кластеры), которые просмотреть на предмет проверки гипотезы, что они принадлежат одной личности.

Итак, модель принимает на вход фотографию и выдает дескриптор, представляющий собой M-мерный вектор. Для каждой пары дескрипторов можно посчитать сходство по косинусному расстоянию (см. [sklearn-функцию](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html#sklearn.metrics.pairwise.cosine_similarity) для N дескрипторов). Вычисленное таким образом сходство, представляет собой число от -1 до 1 (чем больше значение, тем больше объекты похожи). Выбирая порог T∊[-1;1] для каждой пары дескрипторов (фотографий), можно отнести их к одному кластеру или к разным.

**Требуется** по данному набору дескрипторов и порогу разбить выборку на кластеры.

**Входные данные.**

В первой строке указаны три числа N, M и T через пробел. - количество фотографий в выборке. В следующих N строках описываются дескрипторы каждой фотографии. Каждый дескриптор представляет собой M вещественных чисел разделенных однократным пробелом..

**Выходные данные**

Количество кластеров, на которые была разбита выборка.

**Пример**

| **Input** | **Output** |
| --- | --- |
| 3 3 0.6  1.2 0.5 0.4  -1.2 -0.5 0.7  1.2 0.5 -0.4 | 3 |
| 3 3 0.8  1.2 0.5 0.4  -1.2 -0.5 0.7  1.2 0.5 -0.4 | 2 |

*Комментарий. Если фото А похоже на фото B, фото B похоже на фото C, а фото C не похоже на A, то все три фото находятся в одном кластере.*