Scenario: Good morning! You're a user experience engineer at Netflix. A product goal is to design customized home pages for groups of users who have similar interests. You task your team with designing an algorithm for producing a clustering of users based on their movie interests. Your team implements two algorithms that produce different clusterings. How do you decide which one to use? What feedback do you give the team in order to help them improve? Clearly, you will need to use math.

Definition: The set of movie ratings over n movies is R_n, where each element of R_n is a n-tuple with each entry in the tuple one of $\{-1, 0, 1\}$.

Definition: A **partition** of a set A is a set of non-empty, disjoint subsets $A_1, A_2, \cdots, A_n$ such that $A_1 \cup A_2 \cup \cdots \cup A_n = A$.

Idea: A **clustering** is a partition of the elements in a set with the goal of grouping "similar" elements. The definition of "similar" can change based on the problem domain.

Conventions for today: We will use $U = \{r_1, r_2, \cdots, r_t\}$ to refer to an arbitrary set of user ratings (we'll pick some specific examples to explore) that are a subset of R_5. We will be interested in creating partitions $C_1, \cdots, C_m$ of U. We'll assume that each user represented by an element of U has a unique ratings tuple.

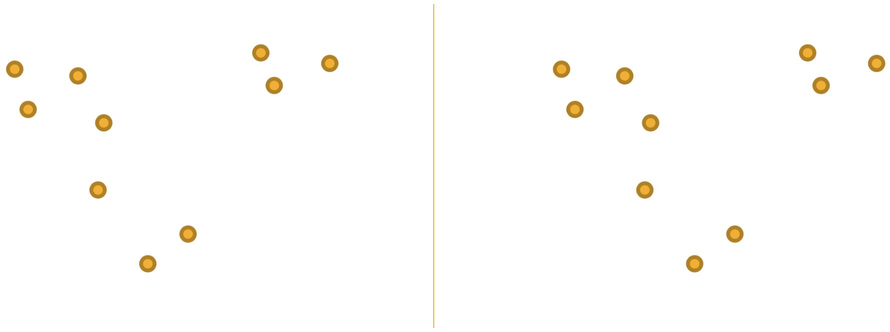
Idea: One way to measure similarity is with functions that measure **distance** between elements.

Definition: The distance between two ratings is defined by d:

$$d(\ (x_1, \ldots, x_n), (y_1, \ldots, y_n)\ ) = \sum_{1 \leq i \leq n} |x_i - y_i|$$

Consider $x = (1, 0, 1, 0, 1)$, $y = (1, 1, 1, 0, 1)$, $z = (-1, -1, 0, 0, 0)$, $w = (0, 0, 0, 1, 0)$.

What is $d(x, y)$? $d(x, z)$? $d(z, w)$?



Definition: For a cluster of ratings $C = \{r_1, r_2, \cdots, r_n\} \subseteq U$, the **diameter** of the cluster is defined by:

$$diameter(C) = \max_{1 \le i,j \le n} (d(r_i, r_j))$$

Consider $x = (1, 0, 1, 0, 1)$, $y = (1, 1, 1, 0, 1)$, $z = (-1, -1, 0, 0, 0)$, $w = (0, 0, 0, 1, 0)$.

What is $diameter(\{x, y, z\})$? $diameter(\{x, y\})$? $diameter(\{x, z, w\})$?

diameter works on single clusters. One way to aggregate across a clustering $C_1, \cdots, C_m$ is ______________________________

Can we easily minimize the sum of diameters?

How can we express the idea of **many elements within a small area**? Key idea: "give credit" to small diameter clusters with many elements.

What is the most useful advice to give the team? Put another way, what is **one number** they can focus on improving, where you (the team leader) understand how that number is calculated.