This week's highlights

- Define clustering as an application of relations and partitions
- Practice working with definitions using numerical and logical expressions
- Review and practice with key concepts and examples from the quarter

Lecture videos

Week 10 Day 1 YouTube playlist

Week 10 Day 2 YouTube playlist

Monday March 8

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:

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

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:

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

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.

Wednesday March 10

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. Your manager tasks you with designing an algorithm for producing a clustering of users based on their movie interests, with the following constraints:

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 $\{-1, 0, 1\}$. The distance between two ratings is defined by d:

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

$U = \{r_1, r_2, \cdots, r_t\}$ is a set of user ratings, and $U \subseteq R_5$. Assume that each user represented by an element of U has a unique ratings tuple. A candidate clustering is $C_1, \cdots, C_m$ that is a **partition** of U: set of non-empty, disjoint subsets of U whose union equals U. We compare candidate clusterings by computing a metric, e.g. min cluster density or average cluster density, where density relates number of ratings in a cluster with the maximum distance between pairs of elements in the cluster.

Definition: A binary relation E on U is an **equivalence relation** means it is reflexive, symmetric, and transitive.

$\forall x \in U($_____$)$, $\forall x \in U\, \forall y \in U\, ($_____$)$, and $\forall x \in U\, \forall y \in U\, \forall z \in U\, ($____

An **equivalence class** of an element $x \in U$ for an equivalence relation E on the set U is the set

$$[x]_E = \{s \in U | (x, s) \in E\}$$

The set of equivalence classes of E is $\{[x]_E \mid x \in U\}$.

Theorem: Given an equivalence relation E on set U, $\{[x]_E \mid x \in U\}$ is a partition of U.

Proof

- To show: For each $a \in U$, $[a]_E \neq \emptyset$, and for each $a \in U$, there is some $b \in U$ such that $a \in [b]_E$.
- To show: For each $a, b \in U$, $(\ (a, b) \in E\ ) \rightarrow (\ [a]_E = [b]_E\ )$
- To show: For each $a, b \in U$, $(\ (a, b) \notin E\ ) \rightarrow (\ [a]_E \cap [b]_E = \emptyset\ )$

$$E_{proj} = \{(\ (x_1, x_2, x_3, x_4, x_5), (y_1, y_2, y_3, y_4, y_5)\ ) \in U \times U \ \mid \ (x_1 = y_1) \wedge (x_2 = y_2) \wedge (x_3 = y_3)\}$$
$$E_{dist} = \{(u, v) \in U \times U \ \mid \ d(u, v) \leq 2\}$$
$$E_{circ} = \{(u, v) \in U \times U \ \mid \ d((0, 0, 0, 0, 0), u) = d((0, 0, 0, 0, 0), v)\}$$

	E_{proj}	E_{dist}	E_{circ}
Example $(u, v) \in E_{\_\_}$			
Example $(u, v) \notin E_{\_\_}$			

Claim: ____ is not an equivalence relation.

Proof:

The partition of U defined by _____ is:

The partition of U defined by _____ is:

What are some properties of the densities of the clusters made by these equivalence relations?

Tips for future classes

From the CSE 20 TAs and tutors

- In class
  - Go to class
  - Show up to class early because sometimes seats get taken/ the classroom gets full and then you have to sit on the floor
  - There's usually a space for skateboards/longboards/eboards to go at the front or rear of the lecture hall
  - If you have a flask water bottle please ensure that its secured during a lecture and it cannot fall - putting on the floor often leads to it falling since people sometimes cross your seats.
  - Take notes - it's much faster and more effective to note-take in class than watch recordings after, particularly if you do so long-hand
  - Resist the urge to sit in the back. You will be able to focus much better sitting near the front, where there are fewer screens in front of you to distract from the lecture content
  - If you bring your laptop to class to take notes / access class materials, sit towards the back of the room to minimize distractions for people sitting behind you!
  - On zoom it's easy to just type a question out in chat, it might be a little more awkward to do so in person, but it is definitely worth it. Don't feel like you should already know what's being covered
  - Always check you have your iclicker[1] on you. Just keep it in your backpack permanently. That way you can never forget it.
  - Don't be afraid to talk to the people next to you during group discussions. Odds are they're as nervous as you are, and you can all benefit from sharing your thoughts and understanding of the material

[1] iclickers are used in many classes to encourage active participation in class. They're remotes that allow you to respond to multiple choice questions and the instructor can show a histogram of responses in real time.

  - Certain classes will podcast the lectures, just like Zoom archives lecture recordings, at podcast.ucsd.edu
  - If they aren't podcasted, and you want to record lectures, ask your professor for consent first

- Office hours, tutor hours, and the CSE building
  - Office hours are a good place to hang out and get work done while being able to ask questions as they come up
  - Office hour attendance is typically much busier in person (and confined to the space in the room)
  - Get to know the CSE building: CSE B260, basement labs, office hours rooms
  - Know how to get in to the building after-hours

- Libraries and on-campus resources
  - Look up what library floors are for what, how to book rooms: East wing of Geisel is open 24/7 (they might ask to see an ID if you stay late), East Wing of Geisel has chess boards and jigsaw puzzles, study pods on the 8th floor, free computers/wifi
  - Know Biomed exists and is usually less crowded
  - Most libraries allow you to borrow whiteboards and markers (also laptops, tablets, microphones, and other cool stuff) for 24 hours
  - Take advantage of Dine with a prof / Coffee with a prof program. It's legit free food / coffee once per quarter.
  - When planning out your daily schedule, think about where classes are, how much time will they take, are their places to eat nearby and how you can schedule social time with friends to nearby areas
  - Take into account the distances between classes if they are back to back

- Final exams
  - What are 8am finals? Basically in-person exams are different
  - Don't forget your university card during exams

- Blue books for exams (what they are, where to get them)
- Seating assignments for exams and go early to make sure you're in the right place (check the exits to make sure you're reading it the right way)
- Know where your exam is being held (find it on a map at least a day beforehand). Finals are often in strange places that take a while to find

CSE department course numbering system

Lower division

- CSE 12, Basic Data Structures and Object-Oriented Programming
- CSE 15L (2 units), Software Tools and Technique Laboratory
- CSE 20 or Math 15A, Introduction to Discrete Mathematics
- CSE 21 Mathematics for Algorithms and Systems
- CSE 30, Computer Organization & Systems Programming

Upper division

- Advanced Data Structures and Programming: CSE 100
- Theory and Algorithms: CSE 101, CSE 105
- Software Engineering: CSE 110, CSE 112
- Systems/Networks: CSE 120 or CSE 123 or CSE 124
- Programming Languages /Databases: CSE 130 or CSE 132A
- Security/Cryptography: CSE 107 or CSE 127
- AI / Machine Learning/ Vision/ Graphics: CSE 150A or CSE 151A or CSE 151B or CSE 152A or CSE 158 or CSE 167
- Hardware / Architecture: CSE 140/ CSE 140L Components and Design Techniques for Digital Systems Architecture, CSE 141 / CSE 141L Introduction to Computer Architecture and CSE 141L Project in Computer Architecture (2 units), CSE 142 / CSE 142L Comp Arch Software Perspective

Friday March 12

Convert $(2A)_{16}$ to

- binary (base ____)
- decimal (base ____)
- octal (base ____)
- ternary (base ____)

Claim: every integer greater than 1 is a product of primes
Proof:

Let $W = \mathcal{P}(\{1, 2, 3, 4, 5\})$. Consider the statement

$$\forall A \in W \ \forall B \in W \ \forall C \in W \ ((A \cap B = A \cap C) \to (B = C))$$

Negate the statement and decide whether it or its negation is true.

The bases of RNA strands are elements of the set $B = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{U}\}$. Each of the sets below is described using set builder notation. Rewrite them using the roster method.

- $\{s \in S \mid s \text{ has length } 2\}$

- $\{s \in S \mid \text{the leftmost base in } s \text{ is the same as the rightmost base in } s \text{ and } s \text{ has length } 3\}$

- $\{s \in S \mid \text{there are twice as many } \texttt{A}\text{s as } \texttt{C}\text{s in } s \text{ and } s \text{ has length } 1\}$

Certain sequences of bases serve important biological functions in translating RNA to proteins. The following recursive definition gives a special set of RNA strands: The set of RNA strands $\hat{S}$ is defined (recursively) by

Basis step:	$\texttt{AUG} \in \hat{S}$
Basis step:	If $s \in \hat{S}$ and $x \in R$, then $sx \in \hat{S}$

where $R = \{\texttt{UUU}, \texttt{CUC}, \texttt{AUC}, \texttt{AUG}, \texttt{GUU}, \texttt{CCU}, \texttt{GCU}, \texttt{UGG}, \texttt{GGA}\}$.

Each of the sets below is described using set builder notation. Rewrite them using the roster method.

- $\{s \in \hat{S} \mid s \text{ has length less than or equal to } 5\}$

- $\{s \in S \mid \text{there are twice as many } \texttt{C}\text{s as } \texttt{A}\text{s in } s \text{ and } s \text{ has length } 6\}$

The set of linked lists of natural numbers L is defined by

Basis step:	$[] \in L$
Basis step:	If $l \in L$ and $n \in \mathbb{N}$, then $(n, l) \in L$

The function $length : L \to \mathbb{N}$ that computes the length of a list is

Basis step:	$length([]) = 0$
Basis step:	If $l \in L$ and $n \in \mathbb{N}$, then $length((n, l)) = 1 + length(l)$

Prove or disprove: the function *length* is onto.

Prove or disprove: the function *length* is one-to-one.

Suppose A and B are sets and $A \subseteq B$:
True or False? If A is infinite then B is finite.

True or False? If A is countable then B is countable.

True or False? If B is infinite then A is finite.

True or False? If B is uncountable then A is coutable.

Review quiz questions

1. **Monday** Please complete the CAPE and TA evaluations. Once you have done so complete the custom feedback form for this quarter: `https://forms.gle/Gfwy8ABEXsLEvLbQ7`

   Then, (we're using the honor system here), write out the statement "I have completed the end of quarter evaluations" and you'll receive credit for this question.

2. **Monday** Select all and only the partitions of $\{1, 2, 3, 4, 5\}$ from the sets below.

   (a) $\{1, 2, 3, 4, 5\}$

   (b) $\{\{1, 2, 3, 4, 5\}\}$

   (c) $\{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}\}$

   (d) $\{\{1\}, \{2, 3\}, \{4\}\}$

   (e) $\{\{\emptyset, 1, 2\}, \{3, 4, 5\}\}$

3. **Monday** Consider movie ratings of 4 movies, where each rating is a 4-tuple with each component in $\{-1, 0, 1\}$. The Manhattan distance between ratings (x_1, x_2, x_3, x_4) and (y_1, y_2, y_3, y_4) is

$$d(\ (x_1, x_2, x_3, x_4), (y_1, y_2, y_3, y_4)\ ) = \sum_{1 \le i \le 4} |x_i - y_i|$$

   The diameter of a collection of ratings $C = \{r_1, \ldots, r_n\}$ is defined by

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

   (a) Calculate $\mathit{diameter}(\{(-1, -1, -1, -1), (0, 0, 0, 0), (1, 1, 1, 1)\})$

   (b) Calculate $\mathit{diameter}(\{(-1, 0, 0, 1), (0, 1, 0, -1), (1, 0, 0, -1)\})$

   (c) What's the greatest diameter of a collection that has exactly two ratings 4-tuples?

   (d) What's the least diameter of a collection that has exactly two ratings 4-tuples?

4. **Wednesday** Select all and only the correct statements about an equivalence relation E on a set U:

(a) $E \in U \times U$

(b) $E = U \times U$

(c) $E \subseteq U \times U$

(d) $\forall x \in U \ ([x]_E \in U)$

(e) $\forall x \in U \ ([x]_E \subseteq U)$

(f) $\forall x \in U \ ([x]_E \in \mathcal{P}(U))$

(g) $\forall x \in U \ ([x]_E \subseteq \mathcal{P}(U))$

5. **Wednesday** Fill in the blanks in the following proof that, for any equivalence relation R on a set A,

$$\forall a \in A\ ([a]_R \neq \emptyset)\ \wedge\ \forall a \in A\ \exists b \in A\ \ (a \in [b]_R)$$

Proof: Since this statement is a **(a)**_____ we need to prove two goals.

Goal 1: we need to show $\forall a \in A\ ([a]_R \neq \emptyset)$ *Proof of Goal 1*: Towards a **(b)**_____ consider an arbitrary element a in A. We will work to show that $[a]_R \neq \emptyset$, namely that $\exists x \in [a]_R$. By definition of equivalence classes, we can rewrite this goal as $\exists x \in A\ (\ (a, x) \in R\ )$. Towards a **(c)**_____, consider $x = a$, an element of A by definition. By **(d)**_____ of R, we know that $(a, a) \in R$ and thus the existential quantification has been proved.

Goal 2: we need to show $\forall a \in A\ \exists b \in A\ \ (a \in [b]_R)$ *Proof of Goal 2*: Towards a **(e)**_____ consider an arbitrary element a in A. By definition of equivalence classes, we can rewrite the goal as $\exists b \in A\ (\ (b, a) \in R\ )$. Towards a **(f)**_____, consider $b = a$, an element of A by definition. By **(g)**_____ of R, we know that $(a, a) \in R$ and thus the existential quantification has been proved.

Since both goals have been proved, the statement has been proved. □

Consider the following expressions as options to fill in the two proofs above. Give your answer as one of the numbers below for each blank a-c. You may use some numbers for more than one blank, but each letter only uses one of the expressions below.

i universal quantification
ii existential quantification
iii conditional
iv conjunction
v disjunction
vi exclusive or
vii biconditional
viii exhaustive proof
ix proof by universal generalization
x proof of existential using a witness
xi proof by cases
xii direct proof
xiii proof by contrapositive
xiv proof by contradiction
xv reflexivity

xvi symmetry

xvii transitivity