Lecture 4: Sets

CAB203 Discrete Structures

Matthew McKague

Queensland University of Technology

matthew.mckague@qut.edu.au

Outline

Readings

Readings for this lecture:

- Pace: 4.1 to 4.5
- Lawson: 3.1

Readings for next lecture:

- Pace: 3.1 to 3.3

Outline

Set theory (informal)

We talk about groups, collections, and sets frequently:

- The people who live in Brisbane
- Alice, Bob and Charlie
- My fingers and toes
- Fruits
- Undergraduate students who take the bus to school

Set theory is the mathematical theory of sets.

Set theory

- A *set* S is a collection of items (*elements*)
- No order to elements, and we don't count multiples
- One basic property, *membership*:

$$\begin{array}{ll} x \in S & x \text{ is in } S \\ x \notin S & x \text{ is not in } S \end{array}$$

- For $x \in S$ we also say x *is an element of* S or x *is a member of* S

Set theory is subtle. Check out Russel's paradox and ZFC.

Defining sets

We can define sets by:

- listing elements:

$$SMALLPRIMES = \{1, 2, 3, 5, 7\}$$

- setbuilder notation:

$$SQUARES = \{x \in \mathbb{Z} : x = y^2 \text{ for some } y \in \mathbb{Z}\}$$

- a list with implied pattern: (this is generally a bad idea)

$$EVENS = \{2, 4, 6, 8, \ldots\}$$

Some sets

- Integers: $\mathbb{Z} = \{\cdots - 2, -1, 0, 1, 2 \ldots\}$
- Positive integers: $\mathbb{Z}^+ = \{1, 2, 3, \ldots\}$
- Non-negative integers: $\mathbb{Z}_{\geq 0} = \{0, 1, 2, 3, \ldots\}$
- Rationals: $\mathbb{Q} = \left\{ \frac{x}{y} : x, y \in \mathbb{Z} \right\}$
- Real numbers (decimals): $\mathbb{R}$
- Set of numbers 1 through n: $[n] = \{1, \ldots, n\}$
- $\{x \in \mathbb{Z} : x | 60\}$

More sets

Sets don't have to be numbers! Some more sets:

- Empty set (no elements): $\emptyset = \{\}$
- Set of students in CAB203
- { apple, orange, banana }
- Sets don't have to be the same "type": $\{apple, \sqrt{2}, \emptyset\}$
- Sets can include other sets: $\{\{1\}, 2, \emptyset\}$

Membership

We can check whether an element x is in a set S by:

- Checking whether x is in the list, if S is given explicitly
- Checking whether x satisfies the conditions, if S is given in setbuilder notation
- Checking whether x satisfies the implied condition, for sets given like $\{1, 2, \ldots\}$

Membership examples

- $15 \in \{1, 3, 5, \dots\}$ because it follows the implied condition (odd numbers)
- $1 \in \{1, 2, 3\}$ because it is in the explicit list
- $12 \in \{x \in \mathbb{Z} : x|60\}$ because 12 is an integer and 12 divides 60

Why are implied conditions bad?

- Maybe $6 \in \{2, 4, \dots\}$ because it follows the implied condition (even numbers)
- Maybe $6 \notin \{2, 4, \dots\}$ because it does not follow the implied condition (powers of 2)

We can interpret the pattern in multiple ways, so there is no unique answer to this question. We should avoid implied sets!

Equality of sets

Two sets are considered to be equal if they contain the same elements. I.e. $S = T$ when

- Every element $x \in S$ is in T
- Every element $x \in T$ is in S

Are these the same?

- $\{1, 2, 3\}$ is the same set as $\{3, 2, 1\}$ (order doesn't matter)
- $\{1, 1, 1\}$ is the same set as $\{1\}$ (don't count multiples)
- $\{x \in \mathbb{Z} : x^2 = 4\}$ is the same as $\{2, -2\}$

Size of sets

The size of a set is notated like $|S|$, and counts the number of *distinct* elements in the set

- $|\{1, 2, 3\}| = 3$
- $|\{1, 1, 1\}| = 1$
- $|\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}| = 4$

The sizes of infinite sets, like $\mathbb{Z}$, are counted using cardinal numbers, which are a generalisation of $\mathbb{Z}^+$. $|\mathbb{Z}| = \aleph_0$. $|\mathbb{R}| = 2^{\aleph_0}$

Subsets

We say that A is a *subset* of B if every $x \in A$ is in B. We write

$$A \subseteq B.$$

A *proper subset* of B is a subset of B that is not equal to B. We write

$$A \subset B.$$

Note: $A \subset B \equiv A \subseteq B \wedge A \neq B$.

The set of all subsets of a set S is called the *power set* of S, written $P(S)$.

$$P(\{1,2\}) = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$$

Subset examples

- For any set S, $\emptyset$ and S are subsets of S.
- $\{1, 2, 3\}$ is a proper subset of $\{1, 2, 3, 4, 5\}$
- $\mathbb{Z}^+ \subset \mathbb{Z}$
- $\{2x : x \in \mathbb{Z}\} \subset \mathbb{Z}$
- $\{1, 2, 3\}$ is not a subset of $\{2x : x \in \mathbb{Z}\}$
- $\{apple, banana\}$ is a subset of $\{apple, carrot, banana\}$

Set equality again

Using subsets we obtain an equivalent definition of set equality.
$S = T$ when both:

- $S \subseteq T$
- $T \subseteq S$

This is sometimes easier to work with.

More setbuilder notation

There are two general forms that we can use for setbuilder notation:

- *Set comprehension:* Specify a subset of elements that match some condition.

$$\{x \in S : \phi(x)\}$$

$$SQUARES = \{x \in \mathbb{Z} : x = y^2 \text{ for some } y \in \mathbb{Z}\}$$

- *Replacement:* Apply a function to each member of a set and collect the results.

$$\{f(x) : x \in S\}$$

$$SQUARES = \{x^2 : x \in \mathbb{Z}\}$$

Neither of these constructions can create larger sets than you start with.

Setbuilder examples

- $\{x \in \mathbb{Z} : x = 2y \text{ for some } y \in \mathbb{Z}\}$ is every integer that is an integer multiple of 2, i.e. the even numbers, including negative even numbers and 0
- $\{2x : x \in \mathbb{Z}\}$ is everything that is twice an integer, so again the even numbers.
- $\{x : 2x \in \mathbb{Z}\}$ is not proper setbuilder notation. We have a condition for x, but we don't know what subset to take x from.
- $\{x \in \mathbb{R} : 2x \in \mathbb{Z}\}$ is the numbers that double to an integer, i.e. $\{\cdots -1.5, -1, -0.5, 0, 0.5, 1 \dots\}$.

A more general set-builder notation combines replacement and comprehensions as in $\{x^2 : x \in \mathbb{Z}, 0 \leq x \leq 15\}$. These are like replacements with extra conditions on variables.

Set operations

Just like numbers, sets have operations:

- Union: everything in *either* set

$$A \cup B = \{x : x \in A \ \vee \ x \in B\}$$

(not proper set-builder notation! More on this later ...)

- Intersection: everything in *both* sets

$$A \cap B = \{x \in A \ : x \in B\}$$

- Difference: remove items in one set from the other

$$A \setminus B = \{x \in A : x \notin B\}$$

Examples

Set $A = \{1, 2, 3\}$ and $B = \{1, 3, 5\}$. Then:

- $A \cup B = \{1, 2, 3, 5\}$
- $A \cap B = \{1, 3\}$
- $A \setminus B = \{2\}$

Unions again

Our definition of union was not proper setbuilder notation!

$$A \cup B = \{x : x \in A \text{ or } x \in B\}$$

- Looks like set comprehension, but we didn't specify where x is drawn from
- We are trying to build a larger set, but setbuilder notation cannot do this
- Instead, we define $A \cup B$ to be the smallest set S such that $A \subseteq S$ and $B \subseteq S$.

Universe sets

- The *universe* U is the set of all elements that we care about
- U depends on the context
- All elements are assumed to be members of the universe
- Allows us to define *complements*

$$\overline{S} = \{x \in U : x \notin S\}$$

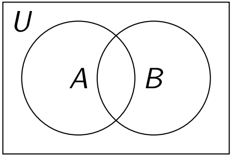
- Can use set comprehensions without specifying the set to draw elements from
- Examples with $U = \mathbb{Z}^+$:

$$\begin{aligned} COMPOSITES &= \overline{PRIMES} \\ EVENS &= \{x : x = 2y\} \\ ODDS &= \overline{EVENS} \end{aligned}$$

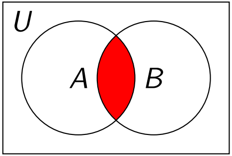
Venn diagrams

Venn diagrams are used for showing the relationship between sets.

- Universe U and two overlapping A and B

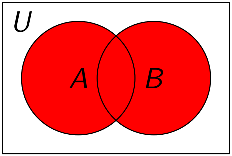


- $A \cap B$

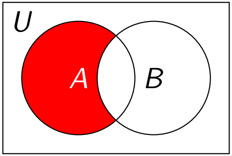


More Venn diagrams

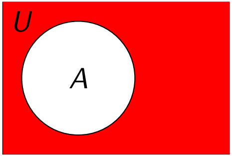
- $A \cup B$



- $A \setminus B$

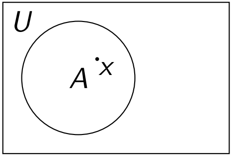


- $\overline{A}$

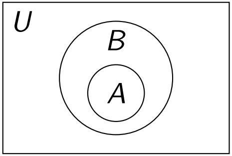


Even more Venn diagrams

- $x \in A$



- $A \subset B$



Uses of set theory in computer science

- basic vocabulary for describing many structures
- models of databases
- tools for understanding how many algorithms treat data

Characteristic vectors

If the universe is not too big, sets can be represented as bit strings (characteristic vectors):

- Let $n = |U|$
- Number the elements like $U = \{e_1, e_2, \ldots e_n\}$
- $\chi_S = \chi_{S1}\chi_{S2}\ldots\chi_{Sn}$ where

$$\chi_{Sj} = \begin{cases} 0 & e_j \notin S \\ 1 & e_j \in S \end{cases}$$

- Eg. $U = \{1, 2, 3\}$ then $\chi_{\{1,3\}} = 101$, $\chi_{\{2\}} = 010$.
- Set operations like $\cup, \cap$ become bitwise operators

Sets in Python

```
>>> S = {1,2,3}; T = {1,3,5}        # Braces define sets
>>> print(S)
{1, 2, 3}
>>> S.add(4); print(S)              # Sets can be changed
{1, 2, 3, 4}
>>> S.remove(4); print(S)
{1, 2, 3}

>>> 1 in S                     # Testing membership
True
>>> 4 in S
False
```

Sets in Python (2)

```
>>> S = {1,2,3}; T = {1,3,5}

>>> S.issubset({1,2})            # testing for subsets
False
>>> S <= {1,2}
False
>>> S.issubset({1,2,3,4,5})
True
>>> S <= {1,2,3,4,5}
True
```

Sets in Python (3)

```
>>> S = {1,2,3}; T = {1,3,5}

>>> S.union(T)
{1, 2, 3, 5}
>>> S | T              # OR is union
{1, 2, 3, 5}
>>> S.union(T, {8,9}, {10, 11}) # multiple union
{1, 2, 3, 5, 8, 9, 10, 11}
>>> someSets = [S, T, {8,9}, {10, 11}]
>>> set.union(*someSets)    # splat operator
{1, 2, 3, 5, 8, 9, 10, 11}

>>> S.intersection(T) # splat and multiple sets work too
{1, 3}
>>> S & T              # AND is intersection
{1, 3}

>>> S - T              # set difference S \ T
{2}
```

Sets in Python (4)

```
>>> S = {1,2,3}; T = {1,3,5}

>>> S.isdisjoint(T)                # is S & T empty?
False

>>> len(S)                         # size of S
3

>>> type({})                       # {} gives empty dictionary
<class 'dict'>
>>> type(set())                    # empty set
<class 'set'>
```

More sets in Python

Python uses a combined comprehension/replacement notation for set-builder notation

```
>>> S = {0, 2, 4, 6, 8}
>>> def p(x): return x % 2 == 0
...
>>> def f(x): return x * 5
...
>>> { f(x) for x in S if p(x) }
{0, 40, 10, 20, 30}
>>> {s * 5 for s in range(0,10) if s % 2 == 0}
{0, 40, 10, 20, 30}
```

Note: the `if` is optional, the `for` is not optional.

Outline

Axiomatic approach

Set theory, like most mathematics, uses an *axiomatic* approach:

- Define a minimal set of *axioms* which are taken to be true
- Use logic to derive consequences of the axioms

The accepted formal definition of set theory is *Zermelo-Fraenkel* set theory.

Zermelo-Fraenkel set theory

ZF set theory is a set of seven axioms which define how sets behave. Informally:

1. **Two sets are equal if they contain the same elements.** I.e. the *only* property of sets is the elements they contain.
2. **Every set S other than $\emptyset$ contains at least one element y, and S and y are disjoint**. So sets can't be members of themselves, and there is only one empty set.
3. **If S is a set and $\phi(x)$ is a formula, then there is a set that contains exactly the elements of S that satisfies $\phi(x)$.** We can use set comprehensions like $\{x \in S : \phi(x)\}$.

Zermelo-Fraenkel set theory (2)

4. **If $S_1, S_2, \ldots$ are sets, then there is a set which contains all of the elements of every S_j.** This allows us to form unions.
5. **If S is a set and $f(x)$ is a function on S, then there is a set which contains $f(x)$ for every $x \in S$.** We can use replacements like $\{f(x) : x \in S\}$
6. **Define $S_0 = \emptyset$ and $S_j = \{S_{j-1}\}$. Then there is a set T which contains every S_j.** There is some set which contains an infinite number of elements.
7. **For a set S, there is a set containing every possible subset of S.** This allows us to form power sets.

Example: defining replacement from ZF axioms

The ZF axioms don't quite give us replacements, but we can get it from axioms 1, 3 and 5.

- Start with a set S and a function $f(x)$.
- Use axiom 5 to obtain a set A which contains $f(x)$ whenever $x \in S$.
- Create a formula $\phi(y)$ to say that there is a $x \in S$ such that $f(x) = y$
- Use axiom 3 to obtain a set A' which contains every $y \in A$ such that $\phi(y)$ is true.

So far we have shown that there is at least one set, namely A', that is what we want.

Example: defining replacement from ZF axioms

Next we need to show that there is *only one* set which fits the criteria. Left as an exercise to the reader.

Set theory's place in mathematics

Set theory is not complicated, but it is considered the foundation of all mathematics

- Sets are a common language used throughout mathematics
- Everything else in mathematics can be built out of sets

Example: $\mathbb{Z}_{\geq 0}$ from set theory

We can construct the set of non-negative integers just from sets! We'll use Peano's axioms, which only require us to define two things:

- What is 0
- A successor function $S(\cdot)$ that takes a number to the next one, i.e. $S(1) = 2$, $S(2) = 3$ ...

We define the empty set, $\emptyset$, as 0 and define a successor function by $S(n) = n \cup \{n\}$. Like so,

- "0" $:= \emptyset = \{\}$
- "1" $:= S($"0"$) =$ "0" $\cup \{$"0"$\} = \{\emptyset\} = \{\{\}\}$
- "2" $:= S($"1"$) =$ "1" $\cup \{$"1"$\} = \{\emptyset, \{\emptyset\}\} = \{\{\}, \{\{\}\}\}$

  $\vdots$
- $S(n) = n \cup \{n\}$

Outline

Syllogisms

Syllogisms are a kind of *deductive reasoning* about sets.

- Start with some *premises* (statements), which are taken to be true
- Apply a valid form of argument
- Draw a conclusion
- If the premises are true then it is logically impossible for the conclusion to be false

Syllogism example

Example:

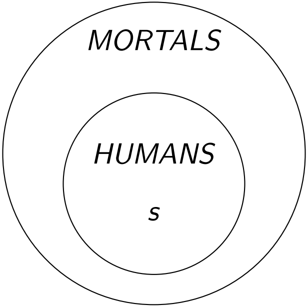
All humans are mortal. (*major premise*)
Socrates is human. (*minor premise*)

Socrates is mortal. (*conclusion*)

Set theoretic notation

We can frame syllogisms in set-theoretic language

$$\frac{\begin{array}{l} HUMANS \subseteq MORTALS \\ s \in HUMANS \end{array}}{s \in MORTALS}$$



Syllogism types

We can abstract the argument to a *syllogism type*:

$$\begin{array}{c} A \subseteq B \\ x \in A \\ \hline x \in B \end{array}$$

This is a *valid type* meaning that it works for any A, B, x as long as the premises are true.

Invalid types

Most argument types are *not* valid:

$$\begin{array}{l} x \in A \cap B \\ y \in B \\ \hline y \in A \end{array}$$

$$\begin{array}{l} \text{Einstein was a genius and a physicist} \\ \text{I am a physicist} \\ \hline \text{I am a genius} \end{array}$$

$$\begin{array}{l} \text{Many flowers smell nice} \\ \text{Rafflesia is a kind of flower} \\ \hline \text{Rafflesia smells nice} \end{array}$$

Invalid type, but correct conclusion

An invalid argument does not mean the conclusion is wrong!

$$\frac{\text{All humans are mortal} \quad \text{Socrates is mortal}}{\text{Socrates is human}}$$

It happens that Socrates was human, but this is an invalid form.
Same form with different premises:

$$\frac{\text{All mothers are mortal} \quad \text{Socrates is mortal}}{\text{Socrates is a mother}}$$

More valid types

There are 24 valid syllogism types. A sampler:

All trees are plants ($A \subseteq B$)
Some trees are tall ($A \cap C \neq \emptyset$)

Some plants are tall ($B \cap C \neq \emptyset$)

All cats are mammals ($A \subseteq B$ and $A \neq \emptyset$)
All cats are carnivores ($A \subseteq C$)

Some mammals are carnivores ($B \cap C \neq \emptyset$)

Out of 256 possible syllogism types, only 24 are valid.