Regular sets

A set is regular if some Finite State Automata (FSA) exists that recognizes it.

- Regular sets are closed under:
  - Union
  - Intersection
  - Complement
  - Concatenation
Converting an NFA to an equivalent DFA

Idea: in the DFA, keep track of the set of states that the NFA could be in.

More formal description

Given an NFA, N, create a DFA whose:
- states are the powerset of the NFA
- start state is the set of all states reachable from N's start state
- final states are those states containing N's final states
- arrows are of form: from a state in DFA, on a symbol go to all states that can be reached from that symbol in set of states in NFA (make sure to follow ε-arrows)
- If desired, remove all unreachable states
  - (Or, create the states on an as-needed basis)
Concatenating two regular sets

A string from the first set followed by a string from the second set
- Idea: guess where the first string ends
- Example: L = binary strings divisible by 4, M = every 0 immediately followed by 2 1s

Regular Expressions

Recognize the same sets as DFA and NFA.
- Equivalent

What is a regular expression?
- Used for pattern matching
  - grep
  - perl
  - etc.
- $(abc)^*(d|e)fgh$
  - Matches any number of repeated abc, followed by either d, or e, then followed by fgh

Three operations:
- Concatenation: regular expressions appear next to each other
- Union: The | specifies a choice among alternative regular expressions
- Kleene star: represents 0 or more repetitions of a regular expression

Recursive definition:
- RE =
  - symbol from alphabet
  - $\epsilon$
  - $\emptyset$
  - RE₁:RE₂
  - (RE₁)'
  - (RE₁ | RE₂)
Examples

Binary strings divisible by 4

Binary strings with each 0 immediately followed by two 1s

Binary strings with each 0 immediately preceded by two 1s

Binary string divisible by 4 followed by string with an even number of 0s

Strings of length 0

Strings over \{a, b, c\} of length 5

Empty set

Relationship between RE and FSA

Any RE can be converted to an equivalent NFA

- Recursive construction
  - For a symbol:

  - For concatenation:

  - For union:

  - For Kleene star:
Example

Convert \((ab|(b^*))^c\) to a FSA

Converting NFA to RE

First, convert to Generalized NFA (GNFA).

Then convert GNFA with \(k\) states to one with \(k-1\) states
  - \(k-1\) to \(k-2\)
  - ...
  - \(3\) to \(2\)

Then, convert GNFA with \(2\) states to regular expression

What is GNFA
  - NFA with:
    - transitions labeled with regular expressions
    - start state
      - no arrows in
      - arrows out to every other state
    - single accept state
      - no arrows out
      - arrow in from every other state
    - other states
      - Single arrow between every pair of states (except start, accept)
Converting NFA to GNFA

Add new start state
- $\epsilon$ transition to old start state
- appropriate transitions to other states

Add new final state
- $\epsilon$ transitions from old final states
- appropriate transitions to other states

Add new transitions
- $\emptyset$ transitions where no transitions exist (other than from new accept or to new start)

Example

Converting GNFA<sub>k</sub> to GNFA<sub>k-1</sub>

Pick a state $d$ to rip out of the GNFA<sub>k</sub>
- not start or final state

Patch up all other pairs of states
- If label from $i$ to $j$ was $RE_i$ new label is $(RE_i | RE_d(RE_d^*RE_d))$

Example:
Example converting RE to FSA

Want RE for binary strings not divisible by 3