Regular languages and finite automata.

Suppose \( \Sigma \) is a finite alphabet.

Recall the following basic operations on languages \( L, K \subseteq \Sigma^* \):

**Concatenation**

\[
LK = \{ xy \mid x \in L, y \in K \}
\]

**Union**

\[
L \cup K = \{ x \mid x \in L \cup x \in K \}
\]

**Kleene star**

\[
L^* = L^0 \cup L^1 \cup L^2 \cup \cdots
\]

\[
= \{ w_1 w_2 \cdots w_n \mid n \geq 0 \text{ and } w_i \in L \}
\]

**Definition** The class of regular languages in \( \Sigma \) is the smallest class \( L \) of languages containing \( \emptyset, \Sigma^* \) (but \( \Sigma \neq \emptyset \)) and such that if \( L, K \in L \), then also \( LK \), \( L \cup K \), \( L^* \in L \).

**Example** All finite languages are regular. Since

\[
\{a, b, \ldots, a_n \} = \{a, b, \ldots, a_{n-1} \} \cup \{a, b, \ldots, a_{n-2} \} \cup \cdots \cup \{a \}
\]
Definition. A determinstic finite state automaton, DFA, consists of a finite directed graph \( G \), where \( V \) is a finite set of vertices and \( E \in V^2 \) is a set of directed edges, along with

(1) a distinguished start state \( s_0 \in V \),

(2) a set \( A \subseteq V \) of accepting states,

(3) a labeling \( l : E \to \mathcal{P}(\Sigma) \). For any \( \sigma \in \Sigma \), there is exactly one edge \( (s, t) \in E \) originating at \( s \) and with label \( \sigma \), i.e., \( \sigma = l(s, t) \).

We call \( V \) the set of states of the automaton.

Given an automaton \( \mathcal{M} \) as above and a string \( w = a_1 a_2 \cdots a_n \in \Sigma^* \), we say that \( \mathcal{M} \) accepts \( w \) if the unique edge path \( (s_0, a_1, a_2, \ldots, a_n) \) in \( \mathcal{M} \) originating at \( s_0 \) and with edges \( l(s, a_i) \) terminates at an accepting state.

Given \( \mathcal{M} \), we let \( L(\mathcal{M}) \) be the language consisting of the strings accepted by \( \mathcal{M} \). We also say that \( \mathcal{M} \) recognizes \( L(\mathcal{M}) \).
Example: We draw a DFA by a diagram in the plane as a usual directed graph with labeled edges. Moreover, the accepting states are indicated by double circles.

\[ \begin{align*}
    & a & \rightarrow & s & \rightarrow & b & \rightarrow a \\
    & b & \rightarrow & s & \rightarrow & b & \rightarrow a
\end{align*} \]

Thus, all has exactly two states and the only accepting state is \( s \). We see that \( \Sigma = \{ a, b \} \) and

\[ L(\text{all}) = \{ w \mid |w|_b = \text{# of occurrences of } b \text{ in } w \text{ is even} \} \]

Similarly, \( \Sigma = \{ a, b, c \} \)

\[ \begin{align*}
    & a & \rightarrow & c & \rightarrow & a & \rightarrow & c & \rightarrow & a, b, c & \rightarrow & a, b, c \\
    & b & \rightarrow & c & \rightarrow & b & \rightarrow & c & \rightarrow & a, b, c & \rightarrow & a, b, c \\
    & c & \rightarrow & a & \rightarrow & c & \rightarrow & a & \rightarrow & a, b, c & \rightarrow & a, b, c
\end{align*} \]

\[ L(\text{all}) = \{ wvcv, bwcv \mid w \in \{ a, b \}^*, v \in \{ a, b, c \}^* \} \]
Definition. A generalized non-deterministic finite state automaton, $\varepsilon$-NFA, is a directed finite graph $\mathcal{A} = (V, E)$ along with

1. a non-empty set of start states $S \subseteq V$,
2. a set of accepting states $A \subseteq V$,
3. a labeling $L : E \rightarrow 2^\Sigma \cup \{\varepsilon\}$.

Given an $\varepsilon$-NFA $\mathcal{A}$ and a string $w \in \Sigma^*$, we say that $\mathcal{A}$ accepts $w$ if there is an edge path $(e_1, \ldots, e_m)$ and labels $b_i \in L(e_i)$ (where $b_i$ can be $\varepsilon$) such that

$$w = b_1 b_2 \cdots b_m$$

Example. We indicate the start states by an $\circ$.

$$L(\mathcal{A}) = 2^1 \left(2b\right)^n c^m \mid n \geq 0, \; m \geq 0 \; ?$$
Suppose $L \subseteq \Sigma^*$ is a language and $w \in \Sigma^*$. We define the \textit{cone type} of $w$ in $L$ by

$$\text{cone}_L(w) = \{ x \in \Sigma^* \mid wx \in L \}$$

and let the \textit{cone types} of $L$ be

$$\text{Cone}(L) = \{ \text{cone}_L(w) \mid w \in \Sigma^* \}$$

\textbf{Note:} The set of cone types of $L$ are the cones \textit{of all words} $w \in \Sigma^*$, not just $L \subseteq \Sigma^*$.

\textbf{Example:} Let $L = \{ a^n b^n \mid n \geq 0 \} \subseteq \Sigma^* a b \Sigma^*$. Then

$$\text{cone}_L(b^2) = \emptyset, \quad \text{cone}_L(e) = L, \quad \text{cone}_L(a^2) = \{ a^0 b^0 \}$$

Note also that $\text{cone}_L(a), \text{cone}_L(a^2), \ldots$ are all distinct.

\textbf{Definition:} For $L \subseteq \Sigma^*$ a language define an \textit{equivalence relation} $\equiv_L$ on $\Sigma^*$ by

$$w \equiv_L v \iff w \in L \text{ and } v \text{ have the same cone type, } \text{ i.e., } \text{cone}_L(w) = \text{cone}_L(v).$$
Lemma: Let $M$ be an $e$-NFA and let $L = L(M)$ be the language accepted by $M$. Then $L$ has only finitely many equivalence classes, i.e., $M$ has only finitely many classes.

Proof: We define another equivalence relation $\approx$ on $E^*$ as follows:

$$w \approx v \iff \text{for any start state } s \text{ and arbitrary state } t \text{ of } M, \text{ there is an edge path from } s \text{ to } t \text{ with edge label } w \text{ at } t \text{ and only if there is an edge path from } s \text{ to } t \text{ with edge label } v.$$

Note that if $M$ has $n$ states, then $\approx$ has at most $2^n$ classes (for each edge $(s, t)$, we have to respond to a yes-no question).

Claim: If $w \approx v$ then also $wx \approx xv$.

For suppose that, e.g., $x \in C_{e_2}^M(w)$. Then there is an edge path beginning at a start state $s$ and terminating at an accepting state $t$, with edge label $wx$, let $t_1$ be any state along this edge path such
that the edge path arises at $t_1$ with label $w$.
Then there is an alternative edge path $e_i \ldots e_p$ beginning at $s$ and terminating at $t_2$ with label $v$. It follows that there is an edge path from $s$ to $t_2$ with label $vx$, where $x \in \text{cone}_2(v)$. Similarly, $\text{cone}_2(v) \subseteq \text{cone}_2(w)$.

**Lemma** Suppose $L \subseteq \Sigma^*$ is a language with finitely many cone types. Then there is a DFA $M$ with $L(M) = L$.

**Proof** Let $\Delta_1, \ldots, \Delta_n$ be the finitely many cone types of $L$ and note that for $w, x \in \Sigma^*$ and $a \in \Sigma^*$, if $wxa \in L$ then also $wya \in L$.

Let $M$ have states $\Delta_1, \ldots, \Delta_n$ and put an arrow $\Delta_i \xrightarrow{a} \Delta_j$ if for some
any \( w \in \Sigma^* \) with \( \text{come}_2(w) = \Delta_i \), we have
\( \text{come}_2(wa) = \Delta_j \). Thus a state \( \Delta_i \) is accepting
if \( \exists \Delta_0 \) and \( s = \text{come}_2(\varepsilon) \) is the unique
start state. All is clearly deterministic and
\( L(\Delta_0) = L \).

\[ \square \]

**Lemma** Let \( \Delta_0 \) be a DFA. Then the language
\( L(\Delta_0) \) is regular.

**Proof** Let \( V \) be the finite set of states of \( \Delta_0 \)
and let \( t_0, t_1 \in V \) be arbitrary. For any
\( X \subseteq V \), let
\[ G(X, t_0, t_1) = \{ w \in \Sigma^* \mid \text{there is an edge path from } \]
\( t_0 \) to \( t_1 \) only passing through
states in \( X \) and having
label \( w \} \).

By induction on \( |X| \), we show that for any \( t_0, t_1 \),
the language \( G(X, t_0, t_1) \) is regular.

\( |X| = 0 \): In this case \( X = \emptyset \) and so \( w \in G(X, t_0, t_1) \)
if and only if \( w = a \) for some \( a \in \Sigma \) for
which \( \exists w \) is an edge \( t_0 \xrightarrow{a} t_1 \).
So \( G(X, t_0, t_1) \) is a (finite) subset of \( \Sigma \).
and hence is regular.

$|X| = n+1$: Assume the result holds for all subsets of $X$ of size $n$ and assume $|X| = n+1$.

Then we have that

$G(X, t_0, t_1) = \left( \bigcup_{q \in X} G(X \cup q, t_0, t_1) \right) \cup \left( \bigcup_{q \in X} G(X \cup q, t_0, q) \circ G(X \cup q, q, t_1) \right)$

which is regular by the induction hypothesis.

Clearly, the right hand side is contained in $G(X, t_0, t_1)$.

Conversely, suppose $w \in G(X, t_0, t_1)$ and consider the edge path from $t_0$ to $t_1$ with label $w = a_1 a_2 \ldots a_n$:

$$t_0 \rightarrow q_1 \rightarrow q_2 \rightarrow \ldots \rightarrow q_{n-1} \rightarrow q_n \rightarrow t_1$$

Thus, if $|w| = n+1$, we have $w \in \bigcup_{q \in X} G(X \cup q, t_0, t_1)$.

Otherwise, note that

- $a_1 \in G(X \cup q_1, t_0, q_1)$
- If $q_1$ is the last occurrence of $q_1$ among $q_1, \ldots, q_{n-1}$, then
\[ a_{i+1} \cdots a_n \in G(X; s_0, q_i, t_i) \]

- if \( q_i = q_f = q_1 \) then \( i \leq f \) and \( q_f \neq q_1 \)
- for all \( i < l < f \), then

\[ a_i a_{i+1} \cdots a_{l-1} \in G(X; s_0, q_i, t_i) \]

Thus, \( w = a_1 \cdots a_n \) belongs to

\[ G(X; s_0, t_0, q_1) \circ G(X; s_0, q_1, t_1) \circ G(X; s_0, q_1, t_1) \]

So also \( L(\mathcal{M}) = \bigcup_{q \in A} G(V; s_0, q) \) is regular.

\[ \square \]

**Theorem:** TFAE for a language \( L \) over a finite alphabet \( \Sigma \)

(a) \( L \) is regular

(b) \( L = L(\mathcal{M}) \) for some DFA \( \mathcal{M} \)

(c) \( L = L(\mathcal{M}) \) for some \( \varepsilon \)-NFA \( \mathcal{M} \)

(d) \( \text{Cone}(L) \) is finite.

This dual equivalence of regular languages and languages recognized by finite automata is known as the *Kleene theorem*, while the equivalence with (d) is the *Myhill–Nerode theorem*.
Proof. We have already proved (c) $\Rightarrow$ (d) $\Rightarrow$ (b) $\Rightarrow$ (a). So we need only prove that regular languages are of the form $L(M)$ for $\varepsilon$-NFA $M$.

Since one can easily build $\varepsilon$-NFA recognizing any finite language, it suffices to show that if $L, K$ are recognized by DFA, then also $L \cap K$ and $L^*$ are recognized by $\varepsilon$-NFA.

So suppose $M_0 = (V_0, E_0, s_0, A_0, l_0)$ and $M_1 = (V_1, E_1, s_1, A_1, l_1)$ are deterministic finite automata recognizing $L$ and $K$ respectively. That is, $V_i$ are the states, $E_i \subseteq V_i \times V_i$ the directed edges, $s_i$ the start state, $A_i$ the accepting states and $l_i : E_i \rightarrow 2(\Sigma) \setminus \emptyset$ the labeling. Why? $V_0 \cap V_1 = \emptyset$.

Let us first build an $\varepsilon$-NFA recognizing $L \cap K$:

Set $N = (V_0 \cup V_1 \cup \Sigma \cup \varepsilon, E_0 \cup E_1 \cup \Sigma(t_0, s_0) \cup \Sigma(t_0, s_1) \cup \varepsilon, A_0 \cup A_1, l)$

where $l(x) = l_0(x)$ whenever $x \in E_0$, $l(x) = l_1(x)$ whenever $x \in \Sigma$, and $l(t_0, s_0) = l(t_0, s_1) = \varepsilon$. 
Thus, at the first stage of the computation, 
without reading any input at all, 
N has to decide to feed the input to either \( M_0 \) or \( M_1 \), so \( L(N) = L \cup K \).

Now let us construct \( N \) to recognize \( L K \).

\[ N = (V_0 \cup V_1, E_0 \cup E_1, V, A_0, A_1, l) \]

where \( l(e) = l_1(e) \) for \( e \in E_1 \) and \( l(q, \delta) = \delta \) for \( q \in A_0 \). Thus, on input \( w \), \( N \) begins with a computation in \( M_0 \) and can leave any accepting state of \( M_0 \) jump to \( q_1 \) and continue with a computation in \( M_1 \). So \( N \) accepts \( w \) if and only if \( w = xy \), where \( x \in L \) and \( y \in K \). Thus, \( L(N) = L K \).
Finally, let $L^*$ be a new state and set

$$N = (V_0 \cup \{t_0\}, E_0 \cup \{(t_0, \varepsilon)\}, \varepsilon, \varepsilon, t_0, A_0 \cup \{t_0\}, \varepsilon)$$

where $l(e) = l_0(e)$ for $e \in E_0$, $l(t_0, s_0) = \varepsilon$,

$$l(q, t_0) = \varepsilon \quad \text{for all } q \in A_0.$$

Again, $L(N) = L^*$.

Remark. We can of course also show that regular languages are recognized by DFA by instead using the Myhill-Nerode Theorem, i.e., by showing that they only have finitely many equivalence classes.

For example, suppose $L$ and $K$ have finitely many equivalence classes and let $\Delta_1, \ldots, \Delta_n$ be the equivalence classes of $K$. Now set

$$W \cong V \iff W \cong_{\Delta_i} V$$

for any $i \leq n$, where $W$ has a decomposition $w = x_1$ with $x_1 \in L$ and $\cong_{\Delta_i} y \Delta_i$ if and only if $V$ has.
Clearly, \( \sim \) is an equivalence relation with finitely many classes (note that the second part is implied by the first question). Moreover, suppose that \( w \sim v \) and that \( w \in LK \). Then either we can write \( w = xy \), where \( y \in K \) and \( x \in L \), whenever also \( v = x'y \) and thus \( \nu = x'y \in LK \), so we can write \( w = xy \), where \( x \in L \) and \( y \in K \). In the second case, let \( \Delta_i = \text{cove}_K(y) \) and note that then \( v \) has a decomposition as \( v = st \), where \( s \in L \) and \( \text{cove}_L(s) = \Delta_i \). It then follows that \( t \in K \), whence \( \nu = stw \in LK \).

In any case, \( \nu \in LK \), so \( \text{cove}_{LK}(w) \subseteq \text{cove}_{LK}(\nu) \).

By symmetry, \( w \nu \nu \). So \( \omega \) defines a \( L \)-cycle and hence \( LK \) has finitely many \( \text{cove} \) types.

**Exercise:** Show that if \( L \uparrow K \) have finitely many \( \text{cove} \) types, then also \( L \cup K \) and \( L^* \) have finitely many \( \text{cove} \) types.

Before giving more examples of regular languages, we give some indication of their limitations.
Pumping lemma

Let \( L \subseteq \Sigma^* \) be a regular language. Then there is an integer \( n \geq 1 \) such that any word \( w \in L \) with \( |w| > n \) can be expressed as

\[
w = xyz, \quad y \in \Sigma^+, \quad x, z \in \Sigma^*
\]

where \(|x| \leq n\) and \(xy^iz \in L\) for all \( i \geq 0\).

Proof
First suppose \( L \) is recognized by a DFA \( M \) with \( n \) states. Then if \( w \) is any word of length \( |w| > n \) then there is an edge path labeled by \( w \) which will have to pass through some state \( q \) more than once. Let \( x \) be the shortest prefix of \( w \) at which a state \( q \) is reached for the second time. Clearly \(|x| \leq n\). Also, let \( y \) be the shortest non-empty string \( s \) on edge paths with label \( xy \) arriving to \( q \). Then so does \( xy^iz \) for all \( i \geq 0 \) and \( xy^iz \in L \) for all \( i \geq 0 \).

Example: The languages \( \{a^n b^n \mid n \geq 0\} \) and \( \{a^n \mid n \text{ is a prime}\} \) are not regular.
The subsequence ordering

**Definition.** Let $< \subseteq$ be a strict partial ordering on a set $X$, i.e., $<$ is transitive and irreflexive. We say that $<$ is a well-quasi-ordering (wqo) if

(i) any antichain is finite, i.e., in any infinite sequence $(x_i)_{i \in \mathbb{N}}$ there are $i \neq j$ such that $x_i = x_j$ or $x_i < x_j$,

(ii) there is no infinite descending chain, i.e., no infinite sequence $(x_i)_{i \in \mathbb{N}}$ with $x_i > x_{i+1} > x_{i+2} > \cdots$ ($<$ is well-founded).

**Exercise.** Let $x \leq y$ if $x < y$ or $x = y$.

Show that $<$ is wqo if and only if for any infinite sequence $(x_i)_{i \in \mathbb{N}}$ there are $i < j$ with $x_i < x_j$.

[Hint: Use Ramsey's theorem.]

**Definition.** Suppose $w = a_1 a_2 \cdots a_n$ and $x$ are words in $\Sigma$. We say that $w$ is a subsequence of $x$ if there are words $y_0, \ldots, y_n \in \Sigma^*$ with $x = y_0 a_1 y_1 a_2 y_2 \cdots a_n y_n$ and $w \neq x$. Write $w < x$ to denote this.
Proposition Let \( \Sigma \) be a finite alphabet.
Then \((\Sigma, \prec)\) is a well-graded ordering.

Proof assume towards a contradiction that there is some sequence \((x_i)_{i \in \mathbb{N}}\) s.t. \(i < j \implies x_i \not\approx x_j\) and call any such sequence bad.

We inductively construct a bad sequence as follows:

Let \(y_1 \in \Sigma^*\) be a word of shortest length beginning an infinite bad sequence.

Let \(y_2 \in \Sigma^*\) be a word of shortest length ending \(y_1\) and beginning an infinite bad sequence.

Let \(y_3 \in \Sigma^*\) be a word of shortest length beginning an infinite bad sequence, etc.

Thus, \(\Sigma\) is finite, there is an infinite subsequence, say \((y_{n_i})_{i \in \mathbb{N}}\), with constant first letter, e.g. \(y_{n_i} = z n_i\). Note again that \(z n_i \not\approx z n_j\) for \(i < j\) and \(z n_i < y_{n_i}\), so

\[ y_1, y_2, \ldots, y_{n_1}, z n_1, z n_2, \ldots \]

is also an infinite bad sequence, but with \(\|z n_1\| < \|y_{n_1}\|\) contradicting the minimality of \(y_1, y_2, \ldots, y_{n_1}, y_{n_1}, y_{n_1+1}, \ldots\). \(\square\)
**Definition** Let \((X, \prec)\) be a strict partial ordering and \(B \subseteq Y \subseteq X\) subsets. We say that \(B\) is a basis for \(Y\) if
\[
Y = \{ x \in X \mid \exists z \in B \, z \preceq x \}\text{.}
\]

**Exercise** Show that \((X, \prec)\) is wpo if and only if any \(Y \subseteq X\), which is closed upwards, i.e., \((y \preceq x \land y \in Y) \implies x \in Y\), has a finite basis.

**Corollary** Let \(L \subseteq \Sigma^*\) be a language closed under subsequences, i.e., if \(x \in L\) and \(x \preceq y\), then \(y \in L\).

Then \(L\) has a finite basis \(B = \{ x_1, \ldots, x_n \}\),
\[
\text{or } L = \{ y \in \Sigma^* \mid x_i \preceq y\ \text{for some } i = 1, \ldots, n \}.
\]

**Theorem** Let \(L \subseteq \Sigma^*\) be any language closed under subsequences. Then \(L\) is regular.

**Proof** Let \(B = \{ x_1, \ldots, x_n \}\) be a finite basis for \(L\) and let \(C\) be the finite set of all prefixes of elements in \(B\).
\[
\text{set } w \preceq v \iff \text{for any } x \in C\text{, } x \preceq w \text{ it and only if } x \preceq v\text{.}
\]
Then $\forall \omega \in \Sigma^* \setminus L$, and since $L$ has only finitely many classes, so does $\Sigma^* \setminus L$, whence $L$ is regular.

**Theorem.** Let $L \subseteq \Sigma^*$ be regular, then so is

$\Sigma^* \setminus L$.

**Proof.** Note that $\text{cone}_{\Sigma^* \setminus L}(w) = \Sigma^* \setminus \text{cone}_L(w)$, so if $L$ has only finitely many cone types, the same holds for $\Sigma^* \setminus L$.

**Corollary.** Let $L \subseteq \Sigma^*$ be a language closed under taking subsequences. Then $L$ is regular.

**Proof.** Just note that $\Sigma^* \setminus L$ is closed under subsequences, so $\Sigma^* \setminus L$ and thus also $L$ are regular.

**Example.** Let $\Sigma = \{0,1, \ldots, 9\}$ and let $L \subseteq \Sigma^+$ be the set of all prime numbers written in base 10 and $K = \{ x \in \Sigma^+ | \exists y \in L \ y \leq x \}$. Then $K$ has a finite basis $B = \{ x_1, \ldots, x_n \}$, where $x_1, \ldots, x_n$ are actually prime numbers written in base 10.
Example: Let \( S = \{2, 1, 2^r\} \) and let \( L \subseteq S^+ \) be the set of all prime numbers written in base 3. Then \( 2, 10, 111 \) is a basis for
\[
\sup(L) = \exists y \in S^+ \mid \exists x \in L \quad x \cdot y = y.
\]
To see this, note first that 2 \( \sim \) 2 \( \sim \) 10 \( \sim \) 111 (where 2, 3, 13 are in base 10). So 2, 10, 111 \( \in L \).

Also, suppose \( x \in L \). Then if 2 \( \not\mid x \), we have \( x \in S_0, S^+ \), and 2 \( \not\mid \) but \( 10 \mid x \), also \( x \in 0^*1^* \). Now, unless \( x \) represents the number 0, \( x \) has no leading 0 and hence \( x \in 1^* \).

So suppose \( x \in 1^* \). Then as neither 1, nor 11 represent prime numbers, we must have \( x \in 111^* \), etc., \( 111 \leq x \).

Notation: Let \( x \equiv y \) if \( x \) is a prefix of \( y \).

Theorem: Suppose \( L \) is a regular language.

Then \( \text{pref}(L) = \{x \in S^* \mid \exists y \in L \quad x \equiv y\} \)

is regular too.

Proof: Exercise. \( \square \)