Distinguishability

**Recall** A *deterministic finite automaton* is a five-tuple $M = (S, \Sigma, T, s_0, F)$ where

- $S$ is a finite set of “states”,
- $\Sigma$ is an alphabet — the “input alphabet”,
- $T : S \times \Sigma \to S$ is the “transition function”,
- $s_0 \in S$ is the “initial state”,
- $F \subseteq S$ is the set of “final” or “accepting” states.

We define the multi-step transition function $T^* : S \times \Sigma^* \to S$ as follows.

1. For any $s \in S$, $T^*(s, \Lambda) = s$.
2. For any $s \in S$, $x \in \Sigma^*$ and $a \in \Sigma$,
   \[ T^*(s, xa) = T(T^*(s, x), a). \]

A string $x \in \Sigma^*$ is *accepted* by $M$ if
\[ T^*(s_0, x) \in F. \]

The language *recognized* by $M$, denoted $L(M)$, is the set of strings accepted by $M$. That is,
\[ L(M) = \{ x \in \Sigma^* \mid T^*(s_0, x) \in F \}. \]
Distinguishing Strings

The use of a DFA to recognize an infinite language depends on the ability to adequately distinguish strings from one another without remembering everything about them.

**Definition**  For any language $L$ over $\Sigma$, and any $x, y, z \in \Sigma^*$, we say $x$ and $y$ are *distinguished by $z$ wrt $L$* if exactly one of $xz$, $yz$ is in $L$. That is, 

$$xz \in L \text{ iff } yz \notin L.$$ 

Similarly, we say that $x$ and $y$ are *distinguishable wrt $L$* if there is some $z \in \Sigma^*$ that distinguishes them.

**Example**  Consider the language

$$L = (\{0, 1\}\{0, 1\})^*.$$ 

The strings 0 and 01 are distinguishable wrt $L$. In fact, any string in $\{0, 1\}^*$ distinguishes them. The strings $\Lambda$ and 01 are indistinguishable wrt $L$. (In fact, for this language, strings $x$ and $y$ are distinguishable iff exactly one of them belongs to $L$.)
Distinguishing Strings — Continued

For any language $L$ over $\Sigma$, and any $x, y, z \in \Sigma^*$, we say $x$ and $y$ are distinguished by $z$ wrt $L$ if exactly one of $xz, yz$ is in $L$. That is,

$$xz \in L \text{ iff } yz \notin L.$$ 

Similarly, we say that $x$ and $y$ are distinguishable wrt $L$ if there is some $z \in \Sigma^*$ that distinguishes them.

---

**Example**  Consider the language

$$L = \{0, 1\}^* \{01\}.$$ 

The strings 1 and 10 are distinguishable wrt $L$. They are distinguished by only one string: 1.
**Distinguishability Lemma**  For any DFA

\[ M = (S, \Sigma, T, s_0, F), \]

for any \( x, y \in \Sigma^* \), if \( x \) and \( y \) are distinguishable wrt \( L(M) \), then

\[ T^*(s_0, x) \neq T^*(s_0, y). \]

**Proof.** Consider any \( x, y \in \Sigma^* \) s.t. \( T^*(s_0, x) = T^*(s_0, y) \). No \( z \in \Sigma^* \) can distinguish \( x \) and \( y \) wrt \( L(M) \). To see this, consider any \( z \in \Sigma^* \).

\[
T^*(s_0, xz) = T^*(T^*(s_0, x), z) \quad \text{(previous result)}
\]
\[
= T^*(T^*(s_0, y), z) \quad \text{(} T^*(s_0, x) = T^*(s_0, y) \text{)}
\]
\[
= T^*(s_0, yz) \quad \text{(previous result)}
\]

Consequently,

\[ T^*(s_0, xz) \in F \iff T^*(s_0, yz) \in F, \]

which shows that \( xz \in L(M) \) iff \( yz \in L(M) \).
**Distinguishability Theorem**  For any language $L$ over $\Sigma$, if there are $n$ strings over $\Sigma$ s.t. each is distinguishable from all the others wrt $L$, then any DFA that recognizes $L$ has at least $n$ states.

**Proof.** Assume that $x_1, x_2, \ldots, x_n$ are all distinguishable from one another wrt $L$. Assume that DFA $M = (S, \Sigma, T, s_0, F)$ recognizes $L$. By the Distinguishability Lemma, since any two distinct strings from $x_1, x_2, \ldots, x_n$ are distinguishable wrt $L$, we can conclude that each of the states $T^*(s_0, x_1), T^*(s_0, x_2), \ldots, T^*(s_0, x_n)$ is distinct. Hence, $M$ has at least $n$ states.

**Distinguishability Corollary**  For any language $L$ over $\Sigma$, if there are infinitely many strings over $\Sigma$ s.t. each is distinguishable from all the others wrt $L$, then there is no DFA that recognizes $L$. Consequently $L$ is not regular.
**Distinguishability Theorem**  For any language $L$ over $\Sigma$, if there are $n$ strings over $\Sigma$ such that each is distinguishable from all the others wrt $L$, then any DFA that recognizes $L$ has at least $n$ states.

For any $n \in \mathbb{N}$, let

$$L_n = \{0, 1\}^* \{1\} \{0, 1\}^n.$$ 

So, in words, $L_n$ is the set of all binary strings that end with a 1 followed by exactly $n$ symbols.

**Claim**  For any $n \in \mathbb{N}$, any DFA that recognizes $L_n$ has at least $2^{n+1}$ states.

As you might imagine, we will prove this by constructing a set of $2^{n+1}$ binary strings that are all distinguishable from one another wrt $L_n$.

Then we can apply the Distinguishability Theorem.

Before proving this claim, let’s construct DFA’s for $L_1$ and $L_2$...
A DFA for $\{0, 1\}^* \{1\} \{0, 1\}^1$
A DFA for \( \{0, 1\}^* \{1\} \{0, 1\}^2 \)
**Claim**  For any \( n \in \mathbb{N} \), any DFA that recognizes
\[
L_n = \{0, 1\}^* \{1\}\{0, 1\}^n.
\]
has at least \( 2^{n+1} \) states.

**Proof.**  First notice that any two distinct strings over \( \{0, 1\} \) of length \( n + 1 \) are distinguishable wrt \( L_n \).

Indeed, any two such strings \( x, y \) differ on the \((k + 1)\)st character, for some \( k \) (\( 0 \leq k \leq n \)).

Assume wlog that the \((k + 1)\)st character of \( x \) is 1 and the \((k + 1)\)st character of \( y \) is 0. So
\[
x \in \{0, 1\}^k \{1\}\{0, 1\}^{n-k} \subseteq \{0, 1\}^* \{1\}\{0, 1\}^{n-k}
\]
and
\[
y \in \{0, 1\}^k \{0\}\{0, 1\}^{n-k} \subseteq \{0, 1\}^* \{0\}\{0, 1\}^{n-k}.
\]

Consequently, \( x \) and \( y \) are distinguished by the string \( 1^k \), with \( x1^k \in L_n \), while \( y1^k \notin L_n \).

Finally, since there are \( 2^{n+1} \) distinct strings over \( \{0, 1\} \) of length \( n + 1 \), we conclude by the Distinguishability Theorem that any DFA that recognizes \( L_n \) has at least \( 2^{n+1} \) states.
\{ 0^n1^n \mid n \in \mathbb{N} \} \text{ is not recognized by any DFA}

**Claim** The language \( L = \{ 0^n1^n \mid n \in \mathbb{N} \} \) is not recognized by any DFA.

We prove this by showing that there is an infinite set of binary strings that are all distinguishable from one another wrt \( L \). (The result then follows by the Distinguishability Corollary.)

**Proof.** There are infinitely many strings of the form \( 0^n \ (n \in \mathbb{N}) \), and all are distinguishable from one another wrt \( L \).

Indeed, for any \( m, n \in \mathbb{N} \), if \( m \neq n \), then \( 0^m \) and \( 0^n \) are distinguished wrt \( L \) by \( 1^n \), since \( 0^m1^n \not\in L \) while \( 0^n1^n \in L \).

It follows by the Distinguishability Corollary that no DFA recognizes \( L \).
Indistinguishability wrt $L$: an “equivalence relation” on strings

**Definition**  For any $L \subseteq \Sigma^*$, let $I_L$ be the binary relation on $\Sigma^*$ s.t.
for all $x, y \in \Sigma^*$,

$$xI_L y \iff x \text{ and } y \text{ are indistinguishable wrt } L.$$ 

Recall: $x$ and $y$ are distinguishable wrt $L$ iff there is a $z \in \Sigma^*$ s.t.

$$xz \in L \text{ iff } yz \notin L.$$ 

So $xI_L y$ iff, for all $z \in \Sigma^*$, $xz \in L$ iff $yz \in L$.

For any $L \subseteq \Sigma^*$, $I_L$ is an “equivalence relation”.

We won’t study this notion independently this semester, although it is very useful.

An equivalence relation on a set “partitions” the set — that is, it divides the set into disjoint subsets — and these disjoint subsets are called “equivalence classes”...
**Equivalence classes wrt $L$**

For any $x \in \Sigma^*$, we will write $[x]$ to stand for the set

$$\{ y \in \Sigma^* \mid xL_{}y \}.$$

So, in words, $[x]$ is the set of all strings that are indistinguishable from $x$ wrt $L$.

We call $[x]$ the *equivalence class* of $x$ (wrt $L$).

Some nice properties of equivalence classes wrt $L$:

0. For all $x \in \Sigma^*$, $x \in [x]$.
1. For all $x, y \in \Sigma^*$, $[x] \cap [y] = \emptyset$ or $[x] = [y]$.
2. For all $x \in \Sigma^*$, $x \in L$ iff $[x] \subseteq L$.
3. $\{ [x] \mid x \in \Sigma^* \}$ is a partition of $\Sigma^*$. That is,
   (a) the elements of the set are disjoint subsets of $\Sigma^*$, and
   (b) their union is $\Sigma^*$.
4. $\{ [x] \mid x \in L \}$ is a partition of $L$. That is,
   (a) the elements of the set are disjoint subsets of $L$, and
   (b) their union is $L$. 
Example  Consider the language

\[ L = ((0 + 1)(0 + 1))^* . \]

Even length strings are indistinguishable wrt \( L \).
Similarly, odd length strings are indistinguishable wrt \( L \).

Hence,

\[ I_L = \{ (x, y) \mid x, y \in \{0, 1\}^*, |xy| \text{ is even} \} . \]

So

\[ [\Lambda] = [00] = [01] = [0000] = \{ x \mid x \in \{0, 1\}^*, |x| \text{ is even} \} . \]

And

\[ [0] = [1] = [010] = [11111] = \{ x \mid x \in \{0, 1\}^*, |x| \text{ is odd} \} . \]
**Observation**  Consider the language

$$PAL = \{ x \mid x \in \Sigma^*, x = x^R \},$$

where $x^R$ stands for the reverse of $x$. It turns out that if $|\Sigma| > 1$, then

$$I_{PAL} = \{ (x, x) \mid x \in \Sigma^* \},$$

because all strings over $\Sigma$ are distinguishable from each other wrt $PAL$.

Therefore, for all $x \in \Sigma^*$,

$$[x] = \{x\}.$$

**BTW:** To see that all strings are distinguishable wrt $PAL$, take any two strings $x, y$ over $\Sigma$. Consider two cases.

**Case 1:** $|x| = |y|$. Then $xx^R \in PAL$, while $yx^R \notin PAL$.

**Case 2:** $|x| \neq |y|$. Wlog assume that $|x| < |y|$. Let $z$ be a string over $\Sigma$ s.t. (i) $|xz| = |y|$, and (ii) $xz \neq y$. (For condition (ii), we need the fact that $|\Sigma| > 1$.) Then $xz(xz)^R \in PAL$, while $yz(xz)^R \notin PAL$. Indeed, since $|y| = |xz|$, $yz(xz)^R$ cannot belong to $PAL$ unless $y = xz$, which by choice of $z$ is not the case.
Minimal DFA Theorem

**Theorem** For any language $L$ over $\Sigma$, let

$$S_L = \{ [x] \mid x \in \Sigma^* \} , \quad F_L = \{ [x] \mid x \in L \}$$

and let $T_L : S_L \times \Sigma \to S_L$ be the unique function s.t. for all $x \in \Sigma^*$ and $a \in \Sigma$, $T_L([x], a) = [xa]$.

If $S_L$ is finite, then

$$M_L = (S_L, \Sigma, T_L, [\Lambda], F_L)$$

is a DFA that recognizes $L$. Moreover, no DFA that recognizes $L$ has fewer states than $M_L$.

**Corollary** (Myhill-Nerode Theorem 1957-58) A language $L$ is regular iff there are finitely many equivalence classes of $I_L$.

**Proof sketch.** The left-to-right part follows from Kleene’s Theorem and the Distinguishability Corollary. The right-to-left part follows from the Minimal DFA Theorem and Kleene’s Theorem.