Lecture 09: Optimality of One-time Pad & Limitations

Overview

- **Optimality.** In today's lecture we shall see that one-time pad is *essentially optimal* (in what exact sense, we shall describe shortly)
- **Limitation.** We shall also characterize the *exact* knowledge that is leaked if one-time pad is used to encrypt two messages

Class of Private-key Encryption Algorithm

For simplicity of proof and clarity of the intuition, we shall consider the class of all private-key encryption algorithms with the following restrictions

1. The key-generation algorithm Gen() outputs a secret key sampled uniformly at random from the set $\mathcal{K}$
2. The encryption algorithm $\mathsf{Enc}_{\mathsf{sk}}(m)$ is deterministic

Figure: Restrictions on Private-key Encryption.

Graph of a Private-key Encryption

Suppose $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a private-key encryption scheme that satisfies the two restrictions in Figure 1. We construct the following bipartite graph

- The left partite set is the set of all message $\mathcal{M}$
- The right partite set is the set of all cipher-texts $\mathcal{C}$
- Given a message $m \in \mathcal{M}$ and a cipher-text $c \in \mathcal{C}$, we add an edge (m, c) labeled sk, if we have $c = \mathsf{Enc}_{\mathsf{sk}}(m)$

This is the *graph corresponding to the encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$

Characterizing Correctness in the Graph Representation

Claim

Given distinct $m, m' \in \mathcal{M}$ and $c \in \mathcal{C}$ the following cannot happen.

There exists an $\mathsf{sk} \in \mathcal{K}$ such that (m, c) and (m', c) are both labeled sk.

Proof.

- Suppose distinct m, m' and c such that $\mathsf{Enc}_{\mathsf{sk}}(m) = \mathsf{Enc}_{\mathsf{sk}}(m') = c$.
- Consider Bob's view
- Bob knows sk and c.
- So, Bob cannot distinguish the case when "c is the encryption of m using sk" from the case when "c is the encryption of m' using sk"
- So, the scheme is not correct

Claim

For every cipher text c, there exists $i \in \{0, 1, 2, \dots\}$ such that it receives exactly i edges from every message $m \in \mathcal{M}$

Proof Outline.

- Consider any cipher text $c \in \mathcal{C}$
- By security, the following quantity is $\mathbb{P}[M = m]$ for all $m \in \mathcal{M}$

$$\mathbb{P}\left[M = m \middle| C = c\right] = \frac{\mathbb{P}[M = m, C = c]}{\mathbb{P}[C = c]}$$
$$= \mathbb{P}[M = m] \frac{\mathbb{P}\left[C = c \middle| M = m\right]}{\mathbb{P}[C = c]}$$

- This implies that $\mathbb{P}\left[C = c \middle| M = m\right]$ is identical to $\mathbb{P}[C = c]$ for all m

Characterizing Security in the Graph Representation II

- Note that
  $\mathbb{P}[C = c] = \sum_{m' \in \mathcal{M}} \mathbb{P}\left[C = c | M = m'\right] \cdot \mathbb{P}\left[M = m'\right]$
- Therefore, we can interpret the quantity $\mathbb{P}[C = c]$ as an average of all the entries in the set $A = \left\{\mathbb{P}\left[C = c | M = m'\right], \text{ for } m' \in \mathcal{M}\right\}$, where the entry $\mathbb{P}\left[C = c | M = m'\right]$ has weight $\mathbb{P}\left[M = m'\right]$
- Security states that the "average" is identical to all the elements in the set A
- So, all the elements in the set A are identical
- Note that $\mathbb{P}\left[C = c | M = m\right] = n_{m,c}/|\mathcal{K}|$, where $n_{m,c}$ is the number of edges between the message m and cipher text c
- Therefore, for a fixed cipher-text c we have $n_{m,c} = n_{m',c}$, for $m, m' \in \mathcal{M}$

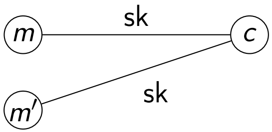

Figure: Correctness rules out this case.

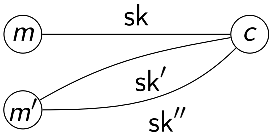

Figure: Security rules out this case.

Shannon's Theorem I

Theorem

Let (Gen, Enc, Dec) *be a private-key encryption scheme that satisfies the restrictions in Figure 1. If this scheme is correct and secure, then* $|\mathcal{K}| \geqslant |\mathcal{M}|$.

Shannon's Theorem II

Proof

- If possible let (Gen, Enc, Dec) be a secure private-key encryption scheme that satisfies the constraints in Figure 1 such that $|\mathcal{K}| < |\mathcal{M}|$
- Consider the graph of this private-key encryption scheme
- There are $|\mathcal{K}|$ edges incident on each message $m \in \mathcal{M}$
- There are a total of $|\mathcal{M}|$ messages in the left set
- The total number of edges, therefore, is $|\mathcal{M}| \cdot |\mathcal{K}|$
- Security implies that every cipher text receives equal edges from each message. So, the number of edges incident on any cipher text is 0, or $|\mathcal{M}|$, or $2|\mathcal{M}|$, or ...
- Since there are $|\mathcal{M}| \cdot |\mathcal{K}| > 0$ edges in total, there is one cipher text c that receives $|\mathcal{M}|$, or $2|\mathcal{M}|$, or $3|\mathcal{M}|$, ... edges.
- Now, the cipher text c receives edges from every message in $\mathcal{M}$. But there are only $|\mathcal{K}| < |\mathcal{M}|$ distinct labels.

Shannon's Theorem III

- So, using the pigeon-hole principle, the cipher text c is connected to two distinct messages using the same secret key
- Therefore, this scheme is not correct!

Comments on the Graph Notation

- Every private-key encryption scheme can be represented using the graph representation
- The canonical key generation algorithm, outputs a random $\mathsf{sk} \xleftarrow{\$} \mathcal{K}$
- The canonical encryption of m using the secret-key sk is the cipher-text c such that (m, c) is labeled sk
- The canonical decryption of a cipher-text c using the secret-key sk is the message m such that (m, c) is labeled sk
- The canonical encryption and decryption algorithms *exist* but need not be *efficient*

One-time Pad is Optimal

- Let $(G, \circ)$ be a group
- **Correctness Condition.** There does not exist $m \neq m'$ and sk such that $m \circ \mathsf{sk} = m' \circ \mathsf{sk}$ (you proved this in HW0)
- **Security Condition.** There does not exists $m, \mathsf{sk} \neq \mathsf{sk}'$ such that $m \circ \mathsf{sk} = m \circ \mathsf{sk}'$ (you will prove this in HW2)
- And we have $|\mathcal{K}| = |\mathcal{M}|$ (Inequality is Tight!)

Limitations I

- As the name suggests, you cannot send two messages using the same secret-key
- Suppose Alice computes $c_1 = m_1 \circ \mathsf{sk}$ and $c_2 = m_2 \circ \mathsf{sk}$ and sends (c_1, c_2) to Bob
- Obviously Bob can decrypt both the cipher-texts using the secret-key sk
- However the security is lost

1. Suppose the adversary thinks that the first cipher-text is an encryption of the message $\widetilde{m_1}$
2. Then the secret-key that explains this pair of message and cipher-text is $\widetilde{\mathsf{sk}} = \mathsf{inv}(\widetilde{m_1}) \circ c_1$
3. This implies that the second cipher-text encrypts the message $\widetilde{m_2} = c_2 \circ \mathsf{inv}(c_1) \circ \widetilde{m_1}$

Limitations II

- That is, conditioned on the first message, the second message is fixed (In a secure two-message encryption scheme, we expect that the second message is independently distribution even conditioned on the first message)

Limitations III

An Example:

- Consider the $(\mathbb{Z}_{26}, +)$ group
- Suppose we encrypt $c_1 = m_1 + \text{sk}$ and $c_2 = m_2 + \text{sk}$
- Seeing the cipher-texts (c_1, c_2), the adversary knows that the two messages are on the form $(\widetilde{m_1}, c_2 - c_1 + \widetilde{m_1})$
- The second message is fixed conditioned on $\widetilde{m_1}$

Limitations IV

Another Example

- Consider the $(\mathbb{Z}_2^n, +)$ group (here "$+$" is the coordinate-wise addition modulo 2)
- Here $c_1 - c_2$ determines which bits of m_1 and m_2 are identical/different

Horrible Encryption

- Suppose someone picks $\mathsf{sk} \xleftarrow{\$} \mathbb{Z}_{26}$
- And encrypts an entire well-formed english sentence $m_1 m_2 \dots m_n$ as $(m_1 + \mathsf{sk})(m_2 + \mathsf{sk}) \dots (m_n + \mathsf{sk})$
- Given this cipher-text an adversary can compute the frequency-list of alphabets in the cipher-text to guess the sk such that the frequency-list matches the one for well-formed English sentences

Additional Reading

- Additional Reading: Caesar cipher, Cryptanalysis of Caesar cipher, Vigenère Cipher, Kasiski Method, Index of coincidence

Simulation Argument: Advanced Reading I

- Suppose we are working with the group $(\mathbb{Z}_{26}, +)$
- Suppose the adversary sees two cipher-texts $c_1 = m_1 + \mathsf{sk}$ and $c_2 = m_2 + \mathsf{sk}$
- We want to claim that the adversary learns only $(m_1 - m_2)$!

Simulation Argument: Advanced Reading II

- The adversary can, at least, compute $(m_1 - m_2)$
- But, how to argue that it learns only $(m_1 - m_2)$?

Simulation Argument: Advanced Reading III

- We proceed by using a *Simulation Argument*
- Suppose we want to state that the adversary learns only "–blah–"
- Then, we construct a polynomial-time algorithm, called the *simulator*, that takes as input "–blah–" and its outputs has the same distribution as the adversary's view
- For example, in this case "–blah–" is "$\Delta = (m_1 - m_2)$," and the simulator needs to output the view of the adversary (C_1, C_2)

- Consider the algorithm below

Sim(Δ) :

1. Sample $x \xleftarrow{\$} G$
2. Output $(x, x - \Delta)$

- The distribution of the output of this algorithm is identical to the distribution of (C_1, C_2)