

# レジュメの書き方

齋藤 翔太

2024 年 4 月 8 日

## 発表の目的

プレゼンテーションの基本的な考え方と、それに基づくレジュメの書き方を伝えることが本発表の目的である。

## 目次

1	プレゼンテーションの基本	2
2	レジュメを書くときのポイント	2
3	発表内容ごとの具体例	3
3.1	定理とその証明の書き方 . . . . .	3
3.2	アルゴリズムの書き方 . . . . .	4
4	L <sup>A</sup> T <sub>E</sub> X について	5
5	まとめ	6
	付録	7
	付録 A 形式的な注意	7
	付録 B 参考文献の書き方	7

## 1 プレゼンテーションの基本

レジュメに限らず、どんな媒体を用いるにしても、良いプレゼンテーションを行うためには以下の手順にしたがってストーリーを考える必要がある。

### 手順 1

目的を明確にする。

具体例をあげると以下ようになる。

本発表 プレゼンテーションの基本的な考え方とレジュメの書き方を伝えること。

学部 3 年生のゼミ 担当範囲のテキストの内容を理解してもらうこと。担当範囲のうち、自分では理解できなかった部分を正確に伝え、議論すること。

学部 4 年生のゼミ 自分の研究の進捗を正確に報告し、今後の研究の進め方についてのアドバイスをもらうこと。

### 手順 2

聴衆は誰か考える。

プレゼンテーションの良し悪しはそのときの聴衆にとってわかりやすかったかどうかで決まる。発表の目的と聴衆の前提知識を突き合わせることで、何をどのような順番でどんな側面から説明すべきかが決まる。具体例をあげると以下ようになる。

学部 3 年生のゼミ 齋藤研究室の学生全体。

4 年生の卒論発表会 情報学部の他の教員。

### 手順 3

発表時間に応じて盛り込む内容を決定する。

発表時間とプレゼンテーションの目的を達成するのに重要なものを考慮して、発表に盛り込む。具体例は以下ようになる。

本発表 発表時間は 15 分程度。優先度は高い順に、プレゼンテーションの基本、レジュメを書くときのポイント、発表内容ごとの具体例。

## 2 レジュメを書くときのポイント

レジュメを作成するときは、特に以下の点に注意する。他の形式的な注意点は付録にまとめてあるので、それらも参照のこと。

- 正確に書く
  - － 式で書く
  - － 擬似コードで書く

- 変数がとりうる値の範囲を書く（例えば、「 $x$  を  $x$  とおく」よりも「 $x \in \mathbb{R}$  とおく」等と書く）
- 写像（関数）なら始集合と終集合を書く（例えば、「関数  $f$ 」よりも「関数  $f: \mathbb{R} \rightarrow \mathbb{R}$ 」等と書く）
- $\sum_{i,j} x_i y_j$  より  $\sum_{i=1}^n \sum_{j=1}^m x_i y_j$  等と書く
- 解釈の仕方が 1 通りしかないような文章にする
- 直観的にわかるように書く
  - 図で書く
  - フローチャートを使う
  - 式で書く
  - 箇条書きで書く
  - 擬似コードで書く
  - 重要なところほど目立たせる
    - \* 枠囲みを使う
    - \* 太字にする
    - \* 下線を引く
  - 本質的でない議論を脚注<sup>\*1</sup>や付録に回す
- 具体例を書く
  - 新たに定義された概念について、変数に値を代入してイメージをつかむ
  - 変数に値を代入して定理を確かめる
  - 変数に値を代入したもとで、アルゴリズムの挙動を追う
- わからなかったところを明確に書く
  - わかったふりをしない
  - 「式 (a) から式 (b) への変形がわからなかった」等、具体的であるほど良い

### 3 発表内容ごとの具体例

#### 3.1 定理とその証明の書き方

定理とその証明を述べるという発表も多い。その際には以下に注意する。

定理とその証明の書き方

- 定理は枠囲みを使って強調する
- 定理の主張そのものが複雑な場合には、その定性的な意味を端的に述べる
- 証明が長い場合には、あらかじめその証明の構成をフローチャート等を用いて説明する

---

<sup>\*1</sup> これが脚注である。

### 3.1.1 定理とその証明の書き方の具体例

定理（相互情報量とエントロピーの関係）

確率変数  $X$  と  $Y$  について、以下の関係式が成り立つ。

$$I(X; Y) = H(X) - H(X|Y) \quad (1)$$

式 (1) の左辺  $I(X; Y)$  は  $Y$  を知ったときに  $X$  に関して得られる情報量を表し、式 (1) の右辺は  $X$  の不確かさ  $H(X)$  から  $Y$  を知ったとしても残っている  $X$  の不確かさ  $H(X|Y)$  を引き算したものを表している。すなわち、相互情報量は不確かさの減少量で表現できることを表している。

証明：相互情報量  $I(X; Y)$  を次のように変形する。

$$I(X; Y) \stackrel{(a)}{=} \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{XY}(x, y) \log_2 \frac{P_{XY}(x, y)}{P_X(x)P_Y(y)} \quad (2)$$

$$= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{XY}(x, y) \log_2 \frac{P_{X|Y}(x|y)}{P_X(x)} \quad (3)$$

$$= - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{XY}(x, y) \log_2 P_X(x) + \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{XY}(x, y) \log_2 P_{X|Y}(x|y) \quad (4)$$

$$= - \sum_{x \in \mathcal{X}} P_X(x) \log_2 P_X(x) - \left( - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_{XY}(x, y) \log_2 P_{X|Y}(x|y) \right) \quad (5)$$

$$\stackrel{(b)}{=} H(X) - H(X|Y) \quad (6)$$

ここで、(a) は相互情報量の定義、(b) はエントロピーと条件付きエントロピーの定義による。ゆえに、 $I(X; Y) = H(X) - H(X|Y)$  が成り立つ。□

## 3.2 アルゴリズムの書き方

アルゴリズムを記述するときのポイントは以下の通りである。

アルゴリズムの書き方

- 入力、出力、処理をすべて明記する
- 擬似コードを使って書くとなお良い
- 具体例による説明を添える

### 3.2.1 アルゴリズムの書き方の具体例

消失訂正 sum-product 復号法を例に説明すると以下ようになる。消失訂正 sum-product 復号法は、消失通信路上で LDPC 符号を復号する際によく用いられる復号法である。以下のアルゴリズムやその具体例の説明は [1] を参考にしたものである。

初期化 受信語  $y \in \{0, 1, ?\}^n$  の各成分  $y_i$  値に応じて、対応する変数ノード  $v_i \in V$  の変数ノード値  $n_{v_i}$  を以

下のように初期化する。

$$n_{v_i} = \begin{cases} 1, & y_i = 0 \\ -1, & y_i = 1 \\ 0, & y_i = ? \end{cases} \quad (7)$$

変数ノード処理 変数ノード  $v_i \in V$  から隣接するチェックノード  $c_j \in \mathcal{N}(v_i)$  へ送られるメッセージ  $m_{v_i \rightarrow c_j}$  は以下ようになる。

$$m_{v_i \rightarrow c_j} = n_{v_i} \quad (8)$$

チェックノード処理 チェックノード  $c_j \in C$  から隣接する変数ノード  $v_i \in \mathcal{N}(c_j)$  へ送られるメッセージ  $m_{c_j \rightarrow v_i}$  は以下ようになる。

$$m_{c_j \rightarrow v_i} = \prod_{v_k \in \mathcal{N}(c_j) \setminus \{v_i\}} m_{v_k \rightarrow c_j} \quad (9)$$

変数ノード値の更新 変数ノード値  $n_{v_i}$  は以下のように更新される。

- 隣接するチェックノード  $c_j \in \mathcal{N}(v_i)$  から送られるメッセージ  $m_{c_j \rightarrow v_i}$  と変数ノード値  $n_{v_i}$  が全て 0 のとき、変数ノード値は  $n_{v_i} = 0$  とする。
- 隣接するチェックノード  $c_j \in \mathcal{N}(v_i)$  から送られるメッセージ  $m_{c_j \rightarrow v_i}$  と変数ノード値  $n_{v_i}$  に 1 つでも 0 でない値が含まれる場合、その値を変数ノード値  $n_{v_i}$  とする。

$\hat{x}$  の計算 変数ノード値  $n_{v_i}$  をもとに、復号語  $\hat{x} \in \{0, 1\}^n$  の各成分  $x_i$  を以下で決定する。

$$x_i = \begin{cases} 0, & n_{v_i} = 1 \\ 1, & n_{v_i} = -1 \end{cases} \quad (10)$$

アルゴリズムを疑似コードで記述すると、次のページのようにまとめられる。また、具体例を用いて各処理を説明すると以下の図 1 のようになる。

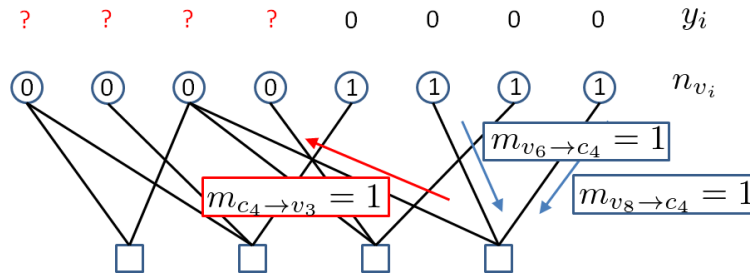


図 1 消失訂正 sum-product 復号法

## 4 L<sup>A</sup>T<sub>E</sub>X について

この資料は、L<sup>A</sup>T<sub>E</sub>X を用いて作成した。これまでに L<sup>A</sup>T<sub>E</sub>X を使ったことがない学生は、Overleaf を用いて使うとよい。Overleaf の設定については、Slack にアップロードした資料を参照のこと。

理工学の論文は L<sup>A</sup>T<sub>E</sub>X を用いて書くことが多いので、徐々に使えるようにしていくこと。また、この資料の L<sup>A</sup>T<sub>E</sub>X ソースコードを Slack にアップロードしておくので、自分でレジュメを作成する際のテンプレートファイルとして使うとよい。

---

**Algorithm 1** 消失訂正 sum-product 復号法

---

**Input:** 受信語  $y \in \{0, 1, ?\}$ , タナーグラフ  $G(V \cup C, E)$

**Output:** 復号語  $\hat{x} \in \{0, 1\}^n$  または error

```
for all  $v_i \in V, c_j \in \mathcal{N}(v_i)$  do { 初期化 }
     $n_{v_i}$  を初期化
end for
while 変数ノード値が更新され続ける限り do
    for all  $v_i \in V, c_j \in \mathcal{N}(v_i)$  do { 変数ノード処理 }
         $m_{v_i \rightarrow c_j}$  を更新
    end for
    for all  $c_j \in C, v_i \in \mathcal{N}(c_j)$  do { チェックノード処理 }
         $m_{c_j \rightarrow v_i}$  を更新
    end for
    for all  $v_i \in V$  do { 変数ノード値の更新 }
         $n_{v_i}$  を更新
    end for
end while
for all  $v_i \in V$  do
    if  $n_{v_i} = 0$  then
        return error
    end if
     $\hat{x}_i$  を計算
end for
return  $\hat{x}$ 
```

---

## 5 まとめ

プレゼンテーションの基本的な考え方と、それに基づくレジュメの書き方を説明した。

## 参考文献

- [1] 和田山正, 誤り訂正技術の基礎, 森北出版, 2010.

## 付録 A 形式的な注意

- ページ番号を振る
- 式番号を振る
- 図、表番号を振る
- 代表的な構成は
  1. タイトル
  2. 発表の目的
  3. 目次
  4. その発表の内容
  5. まとめと（もしあるなら）今後の課題
  6. 参考文献
  7. 付録

## 付録 B 参考文献の書き方

発表の際に参考にした文献があれば参考文献として書いておく。形式は以下の通り。

### 書籍

著者，タイトル，出版社，出版年．

### 論文

著者，“タイトル，” 論文誌名，巻数，号数，該当ページ，発行年．

細かい注意点は以下の通り。

- 日本語の文献では「，」や「．」は全角。英語の文献では半角 + 半角スペース。
- 該当ページの入力は pp. 00–00 とする。間のハイフンは 2 回続けて入力する
- 和書の場合は巻末、洋書の場合はタイトルページの裏側にある奥付を読めば出版社や出版年がわかる。