

2020年度 数理論理学

講義資料(8)

青戸 等人 (知能情報システムプログラム)

目次

- 帰納的定義
- (構造) 帰納法
- 帰納法による証明

(構造) 帰納法

帰納的に定義された対象を示すときによく使われる証明手法の1つとして、(構造) 帰納法がある。帰納法は、直接/間接的に論理学と関係し、論理学においてもコンピュータサイエンス全般においても非常に重要。

この資料では、特に、帰納的定義、そして、構造帰納法について、命題論理式を題材にして学習する。

命題論理の言語

はじめに，「命題論理式が帰納的に定義されている」ことについて説明する．

用いられる個々の記号を言語とよぶ．

命題論理の言語として次の3種類の記号を用いる．

命題変数： $P_0, P_1, P_2, \dots$

命題結合子： $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \top, \perp$

その他： $(,)$

青字が対象記号，黒字はメタ記号．(開・閉)括弧も言語に含まれることに注意． P_1 では， P と $_1$ がそれぞれ独立の記号ではなく， P_1 で1つの記号と見做していることに注意．通常，命題変数は(可算)無限個あるとする．

命題論理式の定義

次の規則に従って約束される記号の列が、命題論理の言葉や言明にあたる．これを**命題論理式**とよぶ．

- (規則1.) P を命題変数とするとき、 P は命題論理式である．
- (規則2.) $\top$ および $\perp$ は命題論理式である．
- (規則3.) A が命題論理式であるとき、 $(\neg A)$ は命題論理式である．
- (規則4.) A と B がともに命題論理式であるとき、
 $(A \wedge B)$ 、 $(A \vee B)$ 、 $(A \rightarrow B)$ 、 $(A \leftrightarrow B)$
は、ともに命題論理式である．
- (規則5.) 規則1～4により構成されるものだけが命題論理式である．

帰納的定義

(規則5)の代わりに「**帰納的に定義する**」という言い方が一般的に用いられる。

定義 8.1. 命題論理式を次のように帰納的に定義する。

(規則1.) P を命題変数とするとき、 P は命題論理式である。

(規則2.) $\top$ および $\perp$ は命題論理式である。

(規則3.) A が命題論理式であるとき、 $(\neg A)$ は命題論理式である。

(規則4.) A と B がともに命題論理式であるとき、
 $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, $(A \leftrightarrow B)$
は、ともに命題論理式である。

なお、講義資料(2)で説明したように、以降、括弧は約束に従って省略する。

例. $((P_0 \rightarrow P_2) \vee (\neg \perp))$ が命題論理式であることを導く.

(1) (規則1.)から, P_0, P_2 は命題論理式.

(2) (規則4.)を使うと, (1)より, $(P_0 \rightarrow P_2)$ は命題論理式.

(3) (規則2.)から, $\perp$ は命題論理式.

(4) (規則3.)を使うと, (3)より, $(\neg \perp)$ は命題論理式.

(5) (規則4.)を使うと, (2)と(4)より, $((P_0 \rightarrow P_2) \vee (\neg \perp))$ は命題論理式.

集合を用いた定義

定義 8.2. 集合 $C_0, C_1, \dots$ を以下のように定義する.

$$\begin{aligned} C_0 &= \{P_i \mid i \in \mathbb{N}\} \cup \{\top, \perp\} \\ C_{n+1} &= C_n \cup \{(\neg A) \mid A \in C_n\} \\ &\quad \cup \{(A \wedge B) \mid A, B \in C_n\} \\ &\quad \cup \{(A \vee B) \mid A, B \in C_n\} \\ &\quad \cup \{(A \rightarrow B) \mid A, B \in C_n\} \\ &\quad \cup \{(A \leftrightarrow B) \mid A, B \in C_n\} \end{aligned}$$

このとき，以下の集合 C_* を命題論理式集合と定義する．

$$C_* = \bigcup_{i \geq 0} C_i$$

冗長だが，普遍的な数学的記述．

“帰納的”の意味

命題論理式の定義における“帰納的”の意味は，

(規則5.) 規則1～4により構成されるものだけが命題論理式であるであった．つまり，**余計なものは入っていない**ということ．

したがって，帰納的に定義された対象は定義の規則の適用を有限回適用されて構成されている．つまり，**その対象に至る，規則の有限回の適用ステップ(導出木)がある**．

$$\frac{\frac{\overline{P_0 \in \text{Prop}} \text{規則1} \quad \overline{P_2 \in \text{Prop}} \text{規則1}}{(P_0 \rightarrow P_2) \in \text{Prop}} \text{規則4} \quad \frac{\overline{\perp \in \text{Prop}} \text{規則2}}{(\neg \perp) \in \text{Prop}} \text{規則3}}{((P_0 \rightarrow P_2) \vee (\neg \perp)) \in \text{Prop}} \text{規則4}$$

Prop は命題論理式集合を表した ($\Rightarrow$ 第2回講義資料)

$((P_0 \rightarrow P_2) \vee (\neg \perp))$ の導出木

$$\begin{array}{c}
 \frac{\overline{P_0 \in \text{Prop}} \text{規則1} \quad \overline{P_2 \in \text{Prop}} \text{規則1}}{(P_0 \rightarrow P_2) \in \text{Prop}} \text{規則4} \quad \frac{\overline{\perp \in \text{Prop}} \text{規則2}}{(\neg \perp) \in \text{Prop}} \text{規則3} \\
 \hline
 ((P_0 \rightarrow P_2) \vee (\neg \perp)) \in \text{Prop} \text{規則4}
 \end{array}$$

演習 8.3. 同様にして, $((\neg P_0) \rightarrow \perp)$ の導出木を書いてみよう.

$((P_0 \rightarrow P_2) \vee (\neg \perp))$ の導出木

$$\begin{array}{c}
 \frac{}{P_0 \in \text{Prop}} \text{規則 1} \quad \frac{}{P_2 \in \text{Prop}} \text{規則 1} \quad \frac{}{\perp \in \text{Prop}} \text{規則 2} \\
 \frac{}{(P_0 \rightarrow P_2) \in \text{Prop}} \text{規則 4} \quad \frac{}{(\neg \perp) \in \text{Prop}} \text{規則 3} \\
 \hline
 ((P_0 \rightarrow P_2) \vee (\neg \perp)) \in \text{Prop} \quad \text{規則 4}
 \end{array}$$

演習 8.3. 同様にして, $((\neg P_0) \rightarrow \perp)$ の導出木を書いてみよ.

解答例 :

$$\begin{array}{c}
 \frac{}{P_0 \in \text{Prop}} \text{規則 1} \\
 \frac{}{(\neg P_0) \in \text{Prop}} \text{規則 3} \quad \frac{}{\perp \in \text{Prop}} \text{規則 2} \\
 \hline
 ((\neg P_0) \rightarrow \perp) \in \text{Prop} \quad \text{規則 4}
 \end{array}$$

BNFを用いた定義

計算機科学の分野では，帰納的定義にBNF (Backus-Naur記法)をよく用いる．

定義 8.4. 命題変数集合 Var および命題論理式集合 Prop を以下のように与える：

$$\begin{aligned} P \in \text{Var} &::= P_0 \mid P_1 \mid \dots \\ A, B \in \text{Prop} &::= P \mid \top \mid \perp \mid (\neg A) \mid (A \wedge B) \\ &\quad \mid (A \vee B) \mid (A \rightarrow B) \mid (A \leftrightarrow B) \end{aligned}$$

数学や論理学の分野ではあまり見かけないが，計算機科学の分野では標準的．

BNF と文脈自由文法

BNF のようなものをどこかで見た記憶がないだろうか？

$$\begin{aligned} P \in \text{Var} & ::= P_0 \mid P_1 \mid \dots \\ A, B \in \text{Prop} & ::= P \mid \top \mid \perp \mid (\neg A) \mid (A \wedge B) \\ & \quad \mid (A \vee B) \mid (A \rightarrow B) \mid (A \leftrightarrow B) \end{aligned}$$

BNF による記述は(ほぼ) **文脈自由文法** に対応する。

$$\begin{aligned} A & \longrightarrow P \mid \top \mid \perp \mid (\neg A) \mid (A \wedge A) \\ & \quad \mid (A \vee A) \mid (A \rightarrow A) \mid (A \leftrightarrow A) \\ P & \longrightarrow P_0 \mid P_1 \mid \dots \end{aligned}$$

実際，BNFで記述された対象のプログラム処理では，字句解析によって命題変数を抽象化して1つの終端記号をして扱えるようにした後，文脈自由言語の技術を用いた構文解析によって，対象の構造を解析する。

目次

- 帰納的定義
- (構造) 帰納法
- 帰納法による証明

(構造) 帰納法

帰納法

帰納的に定義された対象についての性質を示すための証明法

$\mathcal{P}$ を命題論理式に関する性質とする．帰納法は以下を示すことで，すべての命題論理式が性質 $\mathcal{P}$ をもつことを導く．

- (1) 任意の命題変数は性質 $\mathcal{P}$ をもつ．
- (2) $\top$ および $\perp$ は性質 $\mathcal{P}$ をもつ．
- (3) 任意の命題論理式 A について， A が性質 $\mathcal{P}$ をもつならば， $\neg A$ も性質 $\mathcal{P}$ をもつ．
- (4) 任意の命題論理式 A, B について， A も B も性質 $\mathcal{P}$ をもつならば， $A \wedge B, A \vee B, A \rightarrow B, A \leftrightarrow B$ も性質 $\mathcal{P}$ をもつ．

帰納法の仮定

帰納法の仮定

より構成ステップ数が少ない対象に関して性質 $\mathcal{P}$ が成立する，という仮定を**帰納法の仮定**とよぶ．

(3) 任意の命題論理式 A について， A が性質 $\mathcal{P}$ をもつならば，
 $\neg A$ も性質 $\mathcal{P}$ をもつ．

(4) 任意の命題論理式 A, B について， A も B も性質 $\mathcal{P}$ をもつならば，
 $A \wedge B, A \vee B, A \rightarrow B, A \leftrightarrow B$ も性質 $\mathcal{P}$ をもつ．

基本ステップ (Base Step) と帰納ステップ (Induction Step)
帰納法の仮定を用いる場合分けを**帰納ステップ**，帰納法の仮定を用いない場合分けを**基本ステップ**とよぶことがある．

帰納法による推論の妥当性

帰納的に定義された対象は定義の規則の適用を有限回適用されて構成されている．つまり，その対象に至る，規則の有限回の適用ステップがある．

(規則1.) より P_0, P_2 は命題論理式．

従って，(規則4.) より $(P_0 \rightarrow P_2)$ は命題論理式．

(規則2.) より $\perp$ は命題論理式．

従って，(規則3.) より $(\neg \perp)$ は命題論理式．

従って，(規則4.) より $((P_0 \rightarrow P_2) \vee (\neg \perp))$ は命題論理式．

帰納法による証明の(1)～(4)が示されていれば，これらを繰り返し使うことによって，その対象が性質 $\mathcal{P}$ を持つことが導かれる．次のページで具体的に見てみよう．

(規則1.) より P_0, P_2 は命題論理式.

1. 帰納法による証明(1) より, P_0, P_2 は性質 $\mathcal{P}$ をもつ.

従って, (規則4.) より $(P_0 \rightarrow P_2)$ は命題論理式.

2. 従って, 帰納法による証明(4) より, $(P_0 \rightarrow P_2)$ は性質 $\mathcal{P}$ をもつ.

(規則2.) より $\perp$ は命題論理式.

3. 帰納法による証明(2) より, $\perp$ は性質 $\mathcal{P}$ をもつ.

従って, (規則3.) より $(\neg \perp)$ は命題論理式.

4. 従って, 帰納法による証明(3) より, $(\neg \perp)$ は性質 $\mathcal{P}$ をもつ.

従って, (規則4.) より $((P_0 \rightarrow P_2) \vee (\neg \perp))$ は命題論理式.

5. 従って, 帰納法による証明(4) より, $((P_0 \rightarrow P_2) \vee (\neg \perp))$ は性質 $\mathcal{P}$ をもつ.

同様にして, どの命題論理式についても, 性質 $\mathcal{P}$ をもつことが導かれる.

(構造)帰納法 vs. 数学的帰納法

自然数に関する帰納法を特に**数学的帰納法**という．命題論理式の構造に関する帰納法は，命題論理式のサイズ(出現する記号の数)を考えれば，サイズに関する数学的帰納法で示していると考えられることもできる．

逆に，自然数を $0, S(0), \dots$ と考えれば，数学的帰納法は構造帰納法的一种と考えることも出来る．

数学的帰納法を使って、(命題論理式の) 構造帰納法による推論の妥当性を示してみよう。

命題論理式のうち、性質 $\mathcal{P}$ をもつものを D とおく。すると、すべての命題論理式が性質 $\mathcal{P}$ をもつことは、 $C_* \subseteq D$ が成立することに等しい。

帰納法による証明(1)～(4)が示せていると仮定する。

帰納法による証明(1),(2)により、 $C_0 \subseteq D$ が成立する。
帰納法による証明(3),(4)により、 $C_n \subseteq D$ ならば $C_{n+1} \subseteq D$ が成立する。

自然数に関する帰納法により、任意の $i \in \mathbb{N}$ について、 $C_i \subseteq D$ 。よって、和集合の性質から、 $C_* = \bigcup_{i \in \mathbb{N}} C_i \subseteq D$ 。

□

目次

- 帰納的定義
- (構造) 帰納法
- 帰納法による証明

再帰的な定義

帰納法による証明は再帰的な定義とともに用いられることが多い。対象 A をパラメータとする $f(A)$ を定義する際などに、定義のなかに f を用いる場合、その定義を**再帰的**であるという。

例. 自然数 n の階乗 $n!$ を求める関数 f は以下のように与えることが出来る。

$$f(x) = \begin{cases} 1 & (x = 0 \text{ の場合}) \\ x * f(x - 1) & (x > 0 \text{ の場合}) \end{cases}$$

$f(3) = 3 * f(2) = 3 * 2 * f(1) = 3 * 2 * 1 * f(0) = 3 * 2 * 1 * 1 = 6$
再帰的に定義された関数の計算結果を求めるためには、定義を繰り返し展開する必要がある。

再帰的な定義を用いる場合の注意

再帰的な定義を行う際は、きちんと定義になっているか、注意が必要である。例えば、 f の呼び出しが無限に繰り返してしまうなどすると、定義として成立しない。定義が、定義として成立している (意味のある定義になっている) とき、**well-defined** であるという。

例えば、以下の再帰的な定義を考えてみる。

$$f(x) = (\text{if } x = 0 \text{ then } 1 \text{ else } x * f(x - 1))$$

$n \geq 0$ の場合には $f(n) = n!$ となるが、 n が負の数の場合には、 f の呼び出しが無限に繰り返され、定義されない。従って、 f は整数上で **well-defined** でない。

命題論理式のサイズ

定義 8.5. 命題論理式 A の深さ $d(A)$ を次のように定義する。

$$d(A) = \begin{cases} 0 & (A \text{ が命題変数のとき}) \\ 0 & (A = \top \text{ または } A = \perp \text{ のとき}) \\ d(B) + 1 & (A = \neg B \text{ のとき}) \\ \max\{d(B), d(C)\} + 1 & (A = B \wedge C \text{ または } A = B \vee C, \\ & A = B \rightarrow C, A = B \leftrightarrow C \text{ のとき}) \end{cases}$$

$$\begin{aligned} & d((P \rightarrow (Q \rightarrow (\neg R)))) \\ &= \max\{d(P), d((Q \rightarrow (\neg R)))\} + 1 \\ &= \max\{0, d((Q \rightarrow (\neg R)))\} + 1 \\ &= \max\{0, \max\{d(Q), d((\neg R))\} + 1\} + 1 \\ &= \max\{0, \max\{0, d(R) + 1\} + 1\} + 1 \\ &= 3 \end{aligned}$$

演習 8.6. $d(((\neg P) \wedge (Q \rightarrow R)))$ を計算せよ.

演習 8.6. $d(((\neg P) \wedge (Q \rightarrow R)))$ を計算せよ.

$$\begin{aligned} & d(((\neg P) \wedge (Q \rightarrow R))) \\ &= \max\{d((\neg P)), d((Q \rightarrow R))\} + 1 \\ &= \max\{d(P) + 1, \max\{d(Q), d(R)\} + 1\} + 1 \\ &= \max\{1, \max\{0, 0\} + 1\} + 1 \\ &= \max\{1, 1\} + 1 \\ &= 2 \end{aligned}$$

定義 8.7. 命題結合子(命題定数を含む)の個数 $c(A)$ は以下のように定義される.

$$c(A) = \begin{cases} 0 & (A \text{ が命題変数のとき}) \\ 1 & (A = \top \text{ または } A = \perp \text{ のとき}) \\ c(B) + 1 & (A = \neg B \text{ のとき}) \\ c(B) + c(C) + 1 & (A = B \wedge C \text{ または } A = B \vee C, \\ & A = B \rightarrow C, A = B \leftrightarrow C \text{ のとき}) \end{cases}$$

演習 8.8. $c(((\neg P) \wedge (Q \rightarrow R)))$ の値を計算せよ.

$$\begin{aligned}
& c(((\neg P) \wedge (Q \rightarrow R))) \\
&= c((\neg P)) + c((Q \rightarrow R)) + 1 \\
&= (c(P) + 1) + (c(Q) + c(R) + 1) + 1 \\
&= (0 + 1) + (0 + 0 + 1) + 1 \\
&= 3
\end{aligned}$$

演習 8.9. 任意の命題論理式 A について, $d(A) \leq c(A)$ を示せ.

帰納法による証明例 (1)

命題 8.10. 任意の命題論理式 A について, $d(A) \leq c(A)$.

証明. 命題論理式 A に関する帰納法により証明を行う.

1. A が命題変数のとき.

$d(A), c(A)$ の定義より, $d(A) = 0, c(A) = 0$. ゆえに,
 $d(A) \leq c(A)$.

2. $A = \top$ または $A = \perp$ のとき.

$d(A), c(A)$ の定義より, $d(A) = 0, c(A) = 1$. ゆえに,
 $d(A) \leq c(A)$.

3. $A = \neg B$ のとき.

帰納法の仮定から, $d(B) \leq c(B)$.

$d(A), c(A)$ の定義より, $d(A) = d(B) + 1, c(A) = c(B) + 1$.
よって, $d(A) = d(B) + 1 \leq c(B) + 1 = c(A)$.

4. $A = B \wedge C$, または, $A = B \vee C$, $A = B \rightarrow C$,
 $A = B \leftrightarrow C$ のとき.

帰納法の仮定から, $d(B) \leq c(B)$, $d(C) \leq c(C)$.

$d(A)$ の定義より, $d(A) = \max\{d(B), d(C)\} + 1$.

$c(A)$ の定義より, $c(A) = c(B) + c(C) + 1$.

2つの場合に分けて示す.

(1) $d(B) \leq d(C)$ のとき.

このとき, $\max\{d(B), d(C)\} = d(C)$.

よって, $d(A) = \max\{d(B), d(C)\} + 1 = d(C) + 1 \leq$
 $d(B) + d(C) + 1 \leq c(B) + c(C) + 1 = c(A)$.

(2) $d(B) > d(C)$ のとき.

このとき, $\max\{d(B), d(C)\} = d(B)$.

よって, $d(A) = \max\{d(B), d(C)\} + 1 = d(B) + 1 \leq$
 $d(B) + d(C) + 1 \leq c(B) + c(C) + 1 = c(A)$.

よって, いずれの場合も $d(A) \leq c(A)$.

□

帰納法による証明例 (2)

命題論理式の変換 t を以下のように定義する.

$$t(A) = \begin{cases} \neg A & (A \text{ が命題変数のとき}) \\ \top & (A = \perp \text{ のとき}) \\ \perp & (A = \top \text{ のとき}) \\ \neg t(B) & (A = \neg B \text{ のとき}) \\ t(B) \vee t(C) & (A = B \wedge C \text{ のとき}) \\ t(B) \wedge t(C) & (A = B \vee C \text{ のとき}) \\ \neg(t(C) \rightarrow t(B)) & (A = B \rightarrow C \text{ のとき}) \\ \neg(t(B) \leftrightarrow t(C)) & (A = B \leftrightarrow C \text{ のとき}) \end{cases}$$

演習 8.11. $t(P \wedge Q)$ を求めよ.

(解答)

$$\begin{aligned} t(P \wedge Q) &= t(P) \vee t(Q) \\ &= (\neg P) \vee (\neg Q) \end{aligned}$$

演習 8.12. 任意の命題論理式 A について $t(A) \cong \neg A$ となること証明せよ.

証明. 命題論理式 A に関する帰納法により証明を行う.

1. A が命題変数のとき.

$t(A) = \neg A \cong \neg A$ より成立.

2. $A = \perp$ のとき.

$t(A) = \top \cong \neg \perp = \neg A$ より成立.

3. $A = \top$ のとき.

$t(A) = \perp \cong \neg \top = \neg A$ より成立.

4. $A = \neg B$ のとき.

帰納法の仮定から, $t(B) \cong \neg B$.

よって, $t(A) = \neg t(B) \cong \neg \neg B = \neg A$ より成立.

5. $A = B \wedge C$ のとき.

帰納法の仮定から, $t(B) \cong \neg B$, $t(C) \cong \neg C$.

よって, $t(A) = t(B) \vee t(C) \cong \neg B \vee \neg C \cong \neg(B \wedge C) = \neg A$ より成立.

6. $A = B \vee C$ のとき.

帰納法の仮定から, $t(B) \cong \neg B$, $t(C) \cong \neg C$.

よって, $t(A) = t(B) \wedge t(C) \cong \neg B \wedge \neg C \cong \neg(B \vee C) = \neg A$ より成立.

7. $A = B \rightarrow C$ のとき.

帰納法の仮定から, $t(B) \cong \neg B$, $t(C) \cong \neg C$. よって,
 $t(A) = \neg(t(C) \rightarrow t(B)) \cong \neg(\neg C \rightarrow \neg B) \cong \neg(\neg\neg C \vee \neg B) \cong \neg(C \vee \neg B) \cong \neg(\neg B \vee C) \cong \neg(B \rightarrow C) = \neg A$ より成立.

8. $A = B \leftrightarrow C$ のとき.

帰納法の仮定から, $t(B) \cong \neg B$, $t(C) \cong \neg C$. よって,
 $t(A) = \neg(t(B) \leftrightarrow t(C)) \cong \neg(\neg B \leftrightarrow \neg C) \cong \neg((\neg B \rightarrow \neg C) \wedge (\neg C \rightarrow \neg B)) \cong \neg((C \rightarrow B) \wedge (B \rightarrow C)) \cong \neg((B \rightarrow C) \wedge (C \rightarrow B)) \cong \neg(B \leftrightarrow C) = \neg A$ より成立. $\square$

帰納法による証明例 (3)

定義 8.13. 命題論理式 A に含まれる命題変数の集合 $V(A)$ は次のように定義される.

$$V(A) = \begin{cases} \{A\} & (A \text{ が命題変数のとき}) \\ \emptyset & (A = \top \text{ または } A = \perp \text{ のとき}) \\ V(B) & (A = \neg B \text{ のとき}) \\ V(B) \cup V(C) & (A = B \wedge C \text{ または } A = B \vee C, \\ & A = B \rightarrow C, A = B \leftrightarrow C \text{ のとき}) \end{cases}$$

$$\begin{aligned} V((\neg(P \wedge P))) &= V((P \wedge P)) \\ &= V(P) \cup V(P) \\ &= \{P\} \end{aligned}$$

演習 8.14. $V(((\neg R) \vee S))$ を計算せよ.

$$\begin{aligned} V(((\neg R) \vee S)) &= V((\neg R)) \cup V(S) \\ &= V(R) \cup \{S\} \\ &= \{R\} \cup \{S\} \\ &= \{R, S\} \end{aligned}$$

演習 8.15. $|V(A)| \leq 2^{d(A)}$, つまり, 命題論理式 A に含まれる変数の個数はたかだか $2^{d(A)}$ 個となることを示せ.

証明． 命題論理式 A に関する帰納法により証明を行う．

1. A が命題変数のとき．

このとき， V の定義より， $V(A) = \{A\}$ ． よって， $|V(A)| = 1$ ． 一方，命題論理式の深さ d の定義より， $d(A) = 0$ ． よって， $2^{d(A)} = 2^0 = 1$ ．

ゆえに， $|V(A)| = 1 \leq 1 = 2^{d(A)}$ ．

2. $A = \top$ または $A = \perp$ のとき．

このとき， V の定義より， $V(A) = \emptyset$ ． よって， $|V(A)| = 0$ ． 一方，命題論理式の深さ d の定義より， $d(A) = 0$ ． よって， $2^{d(A)} = 2^0 = 1$ ．

ゆえに， $|V(A)| = 0 \leq 1 = 2^{d(A)}$ ．

3. $A = \neg B$ のとき．

このとき， V の定義より， $V(A) = V(B)$ ． よって， $|V(A)| = |V(B)|$ ．

一方、命題論理式の深さ d の定義より、 $d(A) = d(B) + 1$.

また、帰納法の仮定により、 $|V(B)| \leq 2^{d(B)}$.

ゆえに、($d(B) \geq 0$ であることから)、 $|V(A)| = |V(B)| \leq 2^{d(B)} \leq 2^{d(B)} \cdot 2 = 2^{d(B)+1} = 2^{d(A)}$.

4. $A = B \wedge C$, または, $A = B \vee C$, $A = B \rightarrow C$,
 $A = B \leftrightarrow C$ のとき .

このとき、 V の定義より、 $V(A) = V(B) \cup V(C)$. よって、
 $|V(A)| \leq |V(B)| + |V(C)|$. (集合に重なりがあるときは、
 $|V(A)| < |V(B)| + |V(C)|$ となることに注意 .) 一方、命題
論理式の深さ d の定義より、 $d(A) = \max\{d(B), d(C)\} + 1$.

また、帰納法の仮定により、 $|V(B)| \leq 2^{d(B)}$, $|V(C)| \leq 2^{d(C)}$. ゆえに、
 $|V(A)| \leq |V(B)| + |V(C)| \leq 2^{d(B)} + 2^{d(C)} \leq 2^{\max\{d(B), d(C)\}} + 2^{\max\{d(B), d(C)\}} = 2^{\max\{d(B), d(C)\}+1} = 2^{d(A)}$. □

帰納法による証明例 (4)

定義 8.16. 命題論理式 A の部分論理式の集合 $\text{Sub}(A)$ を次のように定義する.

$$\text{Sub}(A) = \begin{cases} \{A\} & (A \text{ が命題変数または} \\ & A = \top \text{ または } A = \perp \text{ のとき}) \\ \text{Sub}(B) \cup \{A\} & (A = \neg B \text{ のとき}) \\ \text{Sub}(B) \cup \text{Sub}(C) \cup \{A\} & (A = B \wedge C \text{ または} \\ & A = B \vee C, A = B \rightarrow C, \\ & A = B \leftrightarrow C \text{ のとき}) \end{cases}$$

例. $\text{Sub}((\neg(P \wedge Q)))$
 $= \text{Sub}((P \wedge Q)) \cup \{(\neg(P \wedge Q))\}$
 $= \text{Sub}(P) \cup \text{Sub}(Q) \cup \{(P \wedge Q)\} \cup \{(\neg(P \wedge Q))\}$
 $= \{P\} \cup \{Q\} \cup \{(P \wedge Q)\} \cup \{(\neg(P \wedge Q))\}$
 $= \{P, Q, (P \wedge Q), (\neg(P \wedge Q))\}$

演習 8.17. $\text{Sub}(((\neg R) \vee R))$ を計算せよ.

$$\begin{aligned}
& \text{Sub}(((\neg R) \vee R)) \\
&= \text{Sub}((\neg R)) \cup \text{Sub}(R) \cup \{((\neg R) \vee R)\} \\
&= \text{Sub}(R) \cup \{(\neg R)\} \cup \{R\} \cup \{((\neg R) \vee R)\} \\
&= \{R\} \cup \{(\neg R)\} \cup \{R\} \cup \{((\neg R) \vee R)\} \\
&= \{R, (\neg R), ((\neg R) \vee R)\}
\end{aligned}$$

演習 8.18. A, B を命題論理式としたとき, $A \in \text{Sub}(B)$ ならば $V(A) \subseteq V(B)$ となることを示せ.

A と B のどちらの命題論理式に関する帰納法を使うと良いのだろうか? この場合, 実は, B に関する帰納法を用いると証明に成功し, A に関する帰納法を用いると証明に失敗する.

証明. 命題論理式 B に関する帰納法により証明を行う.

1. B が命題変数または $B = \top$, $B = \perp$ のとき.

このとき $\text{Sub}(B) = \{B\}$. 従って, $A \in \text{Sub}(B)$ とすると, $A = B$. よって, $V(A) \subseteq V(B)$.

2. $B = \neg C$ のとき.

このとき $\text{Sub}(B) = \text{Sub}(C) \cup \{B\}$. よって, $A \in \text{Sub}(B)$ とすると $A = B$ または $A \in \text{Sub}(C)$. $A = B$ のとき. $V(A) \subseteq V(B)$ は明らか. $A \in \text{Sub}(C)$ のとき. このときは, 帰納法の仮定より $V(A) \subseteq V(C)$. また, 定義より $V(B) = V(C)$. よって $V(A) \subseteq V(B)$ が成立する.

3. $B = C \wedge D$ または $B = C \vee D$, $B = C \rightarrow D$, $B = C \leftrightarrow D$ のとき.

このとき $\text{Sub}(B) = \text{Sub}(C) \cup \text{Sub}(D) \cup \{B\}$. よって, $A \in \text{Sub}(B)$ とすると $A \in \text{Sub}(C)$, $A \in \text{Sub}(D)$, $A = B$ のい

づれかが成立する． $A = B$ のとき． $V(A) \subseteq V(B)$ は明らか． $A \in \text{Sub}(C)$ のとき． 帰納法の仮定より $V(A) \subseteq V(C)$ ．
また，定義より $V(C) \subseteq V(C) \cup V(D) = V(B)$ ． よって，
 $V(A) \subseteq V(B)$ ． $A \in \text{Sub}(D)$ のとき． $A \in \text{Sub}(C)$ と同様．

□

まとめ

- **帰納的定義**
- **(構造) 帰納法による証明**
 - 帰納的に定義された対象に関する性質を証明
 - 帰納法の仮定とは
- **再帰的な定義**
 - それ自身を定義のなかで用いた定義
 - 再帰的に定義された値の計算
 - 定義が well-defined であるとは