Logique propositionnelle : Raisonnement

Dr. NSENGE MPIA HERITIER, Ph.D

Précédemment

- Qu'est-ce que la logique propositionnelle ?
- Le langage propositionnel
- Vérité propositionnelle
- Équivalence
- Calcul logique

Plan de la leçon

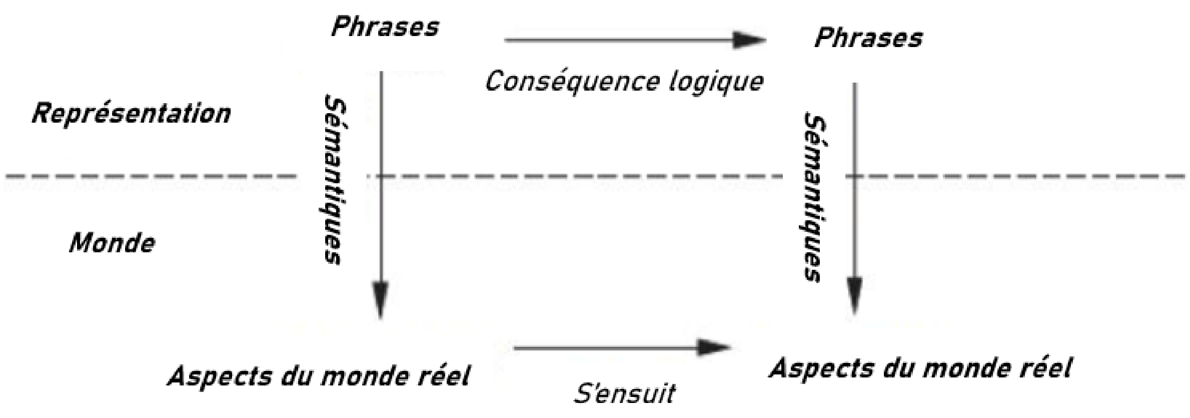
- Conséquence logique
- Arguments
- Règles d'inférence
- Raisonnement déductif à partir d'une base de connaissances

Pourquoi le raisonnement?

- Avec une compréhension de la syntaxe et de la sémantique de la logique propositionnelle pour la représentation des connaissances...
  - Nous explorons le raisonnement de base avec l'inférence logique
- Exploiter une base de connaissances "représentée" pour tirer des conclusions
- Le raisonnement et l'inférence seront des thèmes récurrents tout au long de ce cours

Représentation dans le monde réel

- La logique opère sur des propositions (c'est-à-dire des énoncés) qui sont des représentations d'objets/états du monde réel
- En IA, nous nous intéressons à l'utilisation des connaissances existantes et à l'obtention de nouvelles connaissances ou à la réponse à des questions (en fin de compte dans le monde réel).
- En logique propositionnelle, cela signifie montrer qu'à partir d'une base de connaissances (c'est-à-dire une formule de logique propositionnelle éventuellement étendue), une formule (conclusion) s'ensuit.



Logique : Modèle

- Ce n'est pas la même chose qu'un "modèle" d'apprentissage automatique !

> Un modèle est une interprétation d'une proposition qui est vraie

- Notation :
  - Lorsque la proposition (f) est vraie dans le modèle (M), nous disons : M satisfait f
  - M(f) : désigne l'ensemble des modèles dans lesquels f est vraie

- Exemple: Pour f= $A \wedge B$

  M(f) = { A =1, B =1}.

- Pour f = $A \leftrightarrow B$

  M(f) = {A=1, B=1}; {A =0; B =0}

Tautologie:
Formule telle que M(f) = Tous le modèles possibles.
("Vrai dans tous les mondes possibles")

Exemple : $A \vee \neg A$.
(**Vrai** **pour** A =0 **ou** A=1!)

Contradiction:
Formule telle que M(f) = Ensemble vide.
("Faux dans tous les mondes possibles")

Exemple : $A \wedge \neg A$.
(**Faux** **pour** A =0 **ou** A=1!)

Base de connaissances

- Base de connaissances (KB) : Ensemble de propositions {f1, f2,..., fn}
- M(KB) = Tous les modèles possibles pour $f1 \land f2 \land \dots \land fn$
- Exemple :
  - Variables : R, S, C ("pluvieux", "ensoleillé", "nuageux")
- KB :
  - $R \lor S \lor C$ ("Il est soit pluvieux, soit ensoleillé, soit nuageux.")
  - $R \rightarrow C \land \neg S$ ("S'il pleut, c'est que le ciel est nuageux et non ensoleillé.")
  - $C \leftrightarrow \neg S$ ("S’il le ciel est nuageux, il n’ y a pas de soleil, et vice versa.")

- Exemples de modèles qu’on peut avoir pour KB : {R=1, S =0, C =1} ; {R =0, C=1, S=0} ; {R=0, C=0, S=1}

Conséquence logique

- Représentée par le symbole $\boxed{\models}$
- Un énoncé qui découle logiquement d'une ou de plusieurs propositions.
  - Base de connaissances = ensemble de propositions ou une seule proposition qui affirme toutes les phrases individuelles
  - Les conséquences logiques sont la base pour faire des déductions maximalement fiables à partir d'une base de connaissances
- Exemples:

$$\{p\} \models p \vee q$$

$$\{p\} \not\models p \wedge q$$

Pas de conséquence (Il ne s’ensuit pas)

$$\{p, q\} \models p \wedge q$$

Conséquence logique (Cont.)

- Pour l'instant, par souci de simplicité, nous nous concentrerons sur les conséquences logiques portant sur une seule proposition [Dont découle une conclusion]

- Nous reviendrons sur les propositions multiples d'une base de connaissances, lorsque nous passerons aux arguments.

Équivalence et conséquence logique

Rappel :

$$P \overset{val}{=} Q \qquad \begin{cases} \text{(a) Lorsque } P \text{ est égal à 1, alors } Q \text{ aussi est égal à 1} \\ \text{(b) Lorsque } Q \text{ est égal à 1, alors } P \text{ aussi est égal à 1} \end{cases}$$

Définition :

$$\boxed{P \overset{val}{\models} Q \quad \text{signifie} \quad \{ \text{(a) Lorsque } P \text{ est égal à 1, alors } Q \text{ aussi est égal à 1}}$$

Prononcé $P \overset{val}{\models} Q$ comme "P est plus fort que Q"

- P (notre KB) implique une formule Q (c'est-à-dire que Q découle de KB), si tout modèle de P est aussi un modèle de Q

$$P \models Q$$

P	Q
1 →	1
0	1/0

$P \overset{val}{\models} Q$: **Les 1 sont reportés de P à Q**

Pour démontrer que P implique Q (conséquence logique), il faut que partout où P est 1 (vrai), Q doit l'être aussi. Mais si P est 0 (Faux), Q peut être vrai ou faux.

Détermination de la conséquence logique à partir de la table de vérité

- **Exemple 1 :**

$$\neg P \left\{ \begin{array}{c} \overset{val}{\models} \; ? \\ \overset{val}{=\!\!\mid} \; ? \end{array} \right\} P \Rightarrow Q$$

P	Q	$\neg P$	$P \Rightarrow Q$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

Extra Vrai

Ainsi $\neg P$ **est plus fort que** $P \Rightarrow Q$ (i.e., $\neg P \overset{val}{\models} P \Rightarrow Q$)

- Une proposition "plus forte" implique une proposition plus faible (c'est-à-dire qu'elle l'infère).

Détermination de la conséquence logique à partir de la table de vérité (Cont.)

- Exemple 2 :

$$P \Rightarrow Q \left\{ \begin{matrix} \overset{val}{\models} ? \\ \overset{val}{=\!\!|} ? \end{matrix} \right\} P \vee Q$$

P	Q	$P \Rightarrow Q$	$P \vee Q$
0	0	1	0
0	1	1	1
1	0	0	1
1	1	1	1

Ainsi $P \Rightarrow Q$ **et** $P \vee Q$ **sont** incomparable.

Les affaiblissements standards de la conséquence logique (implication)

- Transformations des conséquences logiques:

$\wedge$-$\vee$- Affaiblissement

$$P \wedge Q \overset{val}{\models} P$$

$$P \overset{val}{\models} P \vee Q$$

P	Q	$P \wedge Q$	P	$P \vee Q$
0	0	0	0	0
0	1	0	0	1
1	0	0	1 $\rightarrow$	1
1	1	1 $\rightarrow$	1 $\rightarrow$	1

Aussi :

$$P \wedge Q \overset{val}{\models} Q$$

et

$$Q \overset{val}{\models} P \vee Q \; .$$

Extremes:

Faux $\overset{val}{\models} P$

$P \overset{val}{\models}$ **Vrai**

Le faux est le plus fort de tous

Le vrai est le plus faible de tous

Propriétés de base de conséquence logique

$P \overset{val}{\models} P.$

Si $P \overset{val}{\models} Q$, alors $Q \overset{val}{=\!\!|} P$, et vice versa.

Si $P \overset{val}{\models} Q$ et $Q \overset{val}{\models} R$, alors $P \overset{val}{\models} R.$

$P \overset{val}{=\!=} Q$ si, et seulement si, $P \overset{val}{\models} Q$ et $P \overset{val}{=\!\!|} Q.$

Ainsi, si on veut prouver $P \overset{val}{\models} Q$ or $P \overset{val}{=\!\!|} Q$ par un calcul,

alors il suffit de prouver $P \overset{val}{=\!=} Q.$

Mais $P \overset{val}{\models} Q$ (ou $P \overset{val}{=\!\!|} Q$) ne suffit pas à conclure

$P \overset{val}{=\!=} Q!$

$P \overset{val}{\models} Q$ si, et seulement si, $P \Rightarrow Q$ est une tautologie

Théorème de déduction

Règle de substitution pour l'implication

La règle de substitution marche aussi pour $\overset{val}{\models}$ et $\overset{val}{=\!\!|}$:

$$\boxed{\textbf{LA SUBSTITUTION PRESERVE LES FAIBLESSES/FORCES}}$$

Exemple

Nous avons l'affaiblissement valide suivant :

$$P \wedge Q \overset{val}{\models} P \vee R$$

et donc, selon la règle de substitution, si nous remplaçons

$(Q \Rightarrow R)$ **pour** P **et** $(P \vee Q)$ **pour** Q **, nous obtenons un autre affaiblissement valide :**

$$(Q \Rightarrow R) \wedge (P \vee Q) \overset{val}{\models} (Q \Rightarrow R) \vee R \ .$$

La règle de Leibniz pour la conséquence logque ? Non!

Rappelons la règle de Leibniz pour établir de nouvelles équivalences :

> **A partir de l'équivalence valide**
>
> $$P \Rightarrow Q \stackrel{val}{=\!=} \neg P \vee Q$$
>
> on peut créer des nouvelles équivalences valide en remplaçant $P \Rightarrow Q$ dans certaines formules complexes par $\neg P \vee Q$, par exemple :
>
> $$(\neg P \wedge (P \Rightarrow Q)) \vee R \stackrel{val}{=\!=} (\neg P \wedge (\neg P \vee Q)) \vee R$$

Exemples

Notons que, par $\wedge$-$\vee$-**Affaiblissement,** $P \wedge Q \stackrel{val}{\models} P \vee Q$. **Considérons maintenant :**

1. $\neg(P \wedge Q) \stackrel{val}{\not\models} \neg(P \vee Q)$;
2. $R \Rightarrow (P \wedge Q) \stackrel{val}{\models} R \Rightarrow (P \vee Q)$; and
3. $(P \wedge Q) \Rightarrow R \stackrel{val}{\not\models} (P \vee Q) \Rightarrow R$.

Conclusion: En remplaçant $\stackrel{val}{=\!=}$ par $\stackrel{val}{\models}$ on n'obtient pas de règle valide

Monotonie

La plupart des logiques formelles sont monotones, ce qui signifie ***qu'ajouter un fait ou un axiome à un ensemble de faits ou d'axiomes n'enlève pas de faits à cet ensemble***. Autrement dit, cela signifie qu'ajouter une nouvelle connaissance à un système ne fera qu'augmenter les faits inférés dans ce système.

Nous disposons de la variante plus faible suivante de la règle de Leibniz

Monotonie :

(1) Si $P \overset{val}{\models} Q$, **alors** $P \wedge R \overset{val}{\models} Q \wedge R$

(2) Si $P \overset{val}{\models} Q$, **alors** $P \vee R \overset{val}{\models} Q \vee R$

- Le syllogisme :
  - "Tous les hommes sont mortels. Socrate est un homme. Donc Socrate est mortel".
- On peut l'affaiblir en ajoutant une prémisse :
  - "Tous les hommes sont mortels. Socrate est un homme. *Les vaches produisent du lait*. Donc, Socrate est mortel".
- N.B: La validité de la conclusion initiale n'est pas modifiée par l'ajout de prémisses

Arguments

Argument valable

Un argument dont la conclusion est vraie si les propositions sont vraies

- Prémisse (axiome) : une proposition tenue pour vraie
- Diverses notations (par exemple les figures 1-3 ci-contre) pour les preuves de formules de propositionnelle
  - (par exemple la figure 2) Peut faire dériver les formules (ou la formule) du dénominateur à partir des formules séparées par des virgules dans le numérateur
  - Une série d'énoncés forme un argument valide si et seulement si "la conjonction (∧) des prémisses impliquant la conclusion" est une tautologie (voir slide suivant)

1

$$\begin{array}{l} p \rightarrow q \\ p \\ \therefore q \end{array}$$

2

$$\frac{p \rightarrow q \, , \, p}{q}$$

3

$$\begin{array}{l} p \rightarrow q \\ p \\ \hline q \end{array}$$

"Si Titi est un oiseau, alors Titi vole".
"Titi est un oiseau"
Prémisses

"Titi vole"
Conclusion

Modèle d'argument valable

- Les lignes critiques de la table de vérité sont les lignes dont toutes les prémisses sont vraies (satisfaites).
  - Il suffit de les calculer dans la table de vérité
- Si, dans toutes les lignes critiques, la conclusion est vraie, l'argument est valide.
  - Sinon, il n'est pas valide
- Argument valide : la conjonction (∧) des prémisses impliquant la conclusion" est une tautologie
  - $(p \rightarrow q) \wedge p \rightarrow q$ dans cet exemple

$$\left.\begin{array}{l} p \rightarrow q \\ p \end{array}\right\} \text{Prémisses}$$

$$\therefore q \longrightarrow \text{Conclusion}$$

$M(KB \cup \{f\}) = \{ p = V, q = V\}$

p	q	p→q	(p→q))∧p	((p→q))∧p) → q
V	V	V	V	V
V	F	F	F	V
F	V	V	F	V
F	F	V	F	V

Pas question même de calculer

Exemple d'argument non valide

"S'il tombe et qu'il se trouve juste au-dessus de moi, je m'enfuis".

"C'est en train de tomber"

"Ce n'est pas directement au-dessus de moi"

Prémisses

"Je ne m'enfuirai pas" **Conclusion**

$S=(t \wedge a \rightarrow e)$;
t;
$\neg a$; **Prémisses**

$\therefore \neg e$ **Conclusion**

a e t	$\neg$a	t $\wedge$ a	S	$\neg$e
V V V	F	V	V	F
V V F	F	F	V	F
V F V	F	V	F	V
V F F	F	F	V	V
F V V	V	F	V	F
F V F	V	F	V	F
F F V	V	F	V	V
F F F	V	F	V	V

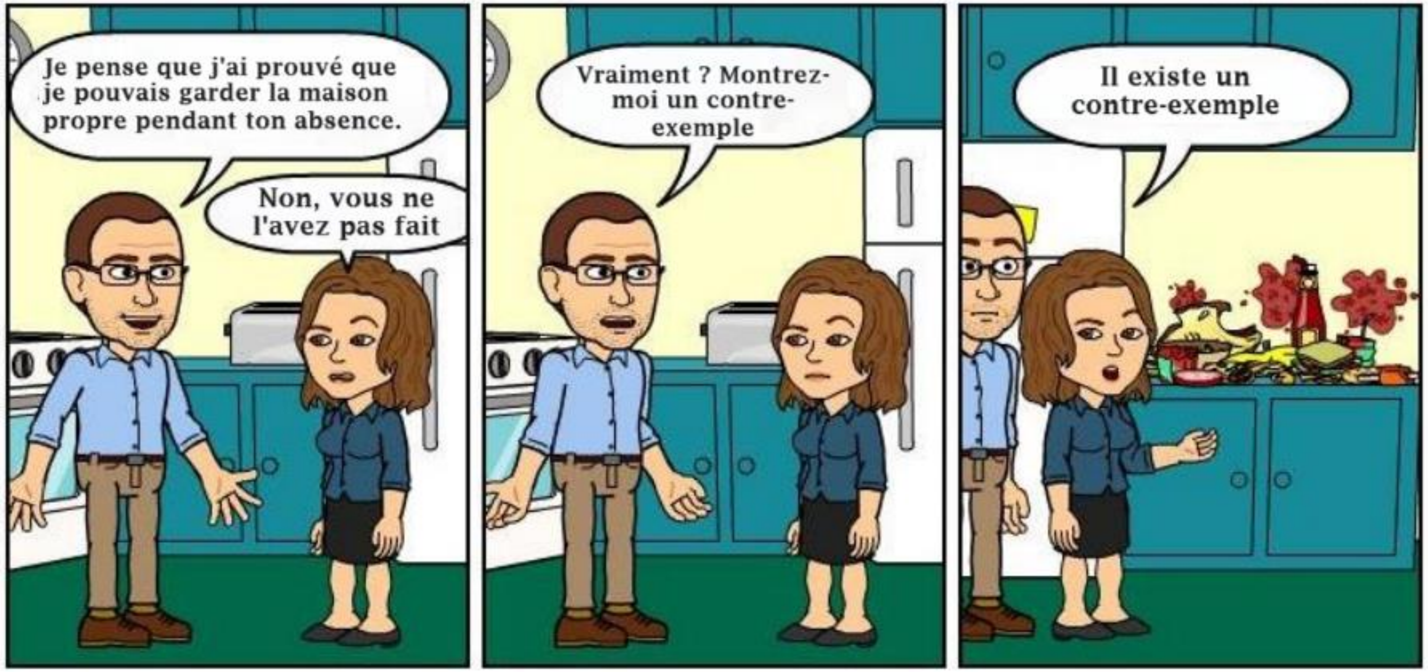
Lignes critiques

Argument invalide : La conclusion dans la 5ième ligne est fausse!

Contre-exemple

- Une ligne critique avec une fausse conclusion est un contre-exemple qui *invalide l'argument*.

"Preuve infaillible"



Rôle de l'implication (Conséquence logique) dans les arguments valides

- Une autre façon de considérer la validité d'un argument
- Un argument logique valide est un argument dans lequel la conclusion est impliquée par les prémisses.

$$\begin{array}{ll} & P \land Q \\ & P \\ \therefore & P \lor Q \end{array}$$

P	Q	$P \land Q$	P	$P \lor Q$
0	0	0	0	0
0	1	0	0	1
1	0	0	1	1
1	1	1	1	1

$P \land Q \models P \lor Q$ ✔

$P \models P \lor Q$ ✔

Argument valide!

Fallacy (croyance erronée)

Une erreur de raisonnement qui aboutit à un argument non valide

- Premier faux raisonnement: erreur de conversion
  - Exemple
    - Si c'est Noël, alors c'est un jour férié
    - C'est un jour férié, donc c'est Noël

p	q	$p \to q$
F	V	V

$$p \to q;\quad q;\quad \therefore\ p$$

- Deuxième faux raisonnement: erreur d'inversion
  - Exemple
    - S'il pleut, alors je resterai à la maison
    - Il ne pleut pas, donc je ne resterais pas à la maison

p	q	$p \to q$	$\neg p$	$\neg q$
F	V	V	V	F

$$p \to q;\quad \neg p;\quad \therefore\ \neg q$$

Argument invalide mais conclusion correcte

- Un argument peut être ***invalide***, mais il peut néanmoins aboutir à une conclusion correcte (par exemple, par coïncidence)
- Exemple
  - Si New York est une grande ville, alors New York a de grands immeubles
  - New York a de grands immeubles
  - Par conséquent : New York est une grande ville
- Que s'est-il passé ?
  - Nous venons de produire un argument invalide
    - Erreur d'inversion !
  - Mais la conclusion est vraie (un fait vrai en soi)

Règles d'inférence

Règles d’inférence

Une règle d'inférence est une construction logique (c'est-à-dire un argument) qui prend des prémisses, analyse leur syntaxe et *renvoie une conclusion*.

- Arguments valides simples qui peuvent être utilisés comme éléments de base pour construire des arguments valides plus compliqués ou pour dériver une ***preuve***.

Une **preuve** est un enchaînement de conclusions qui conduisent à un objectif souhaité

Règles d'inférence : Modus Ponens

- En calcul des propositions, le *modus ponens* est une règle d'inférence, qui peut être résumée de la manière suivante:
  - « P implique que Q et P soient déclarés vrais, donc Q doit être vrai ».

- Le terme modus ponens est une abréviation du latin *modus ponendo ponens* qui signifie « la méthode qui, en affirmant, affirme ».

- Il vient de ce qu'en posant *P*, on pose *Q*.

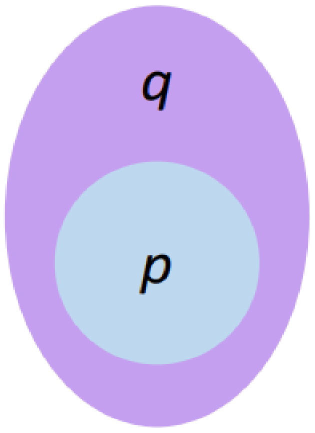
Règles d'inférence : Modus Ponens (Cont.)

- Si *P*, et que *P* implique *Q*, alors *Q*
  - La forme la plus célèbre de syllogisme
  - Nous l'avons déjà vu à de nombreuses reprises (dans ce cours)
- Exemple :
  - "S’il pleut, je vais étudier".
  - "Il pleut"
  - "Par conséquent, je vais étudier"

$$p \rightarrow q$$
$$p$$
$$\therefore q$$

Tautologie correspondante : $(p \wedge (p \rightarrow q)) \rightarrow q$

p	q	$p \rightarrow q$	$p \wedge (p \rightarrow q)$	$(p \wedge (p \rightarrow q)) \rightarrow q$
0	0	1	0	1
0	1	1	0	1
1	0	0	0	1
1	1	1	1	1



Règles d'inférence : Modus Tollens

- En calcul des propositions, le *modus tollens* ou plus précisément *modus tollendo tollens* (du latin « procédé qui *nie* en niant ») désigne un raisonnement logique déductif classique qui part de l'implication «si A alors B», puis niant le conséquent «B» (non B) en déduit la négation de l'antécédent «non A».

  - C'est la méthode de négation.

  - On parle aussi d'une règle d'inférence ou d'un syllogisme conditionnel.

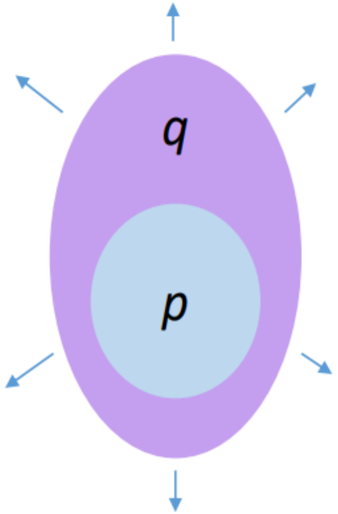
Règles d'inférence : Modus Tollens (Cont.)

- Si non q et que p implique q, alors non p
- Exemple :
  - "S'il pleut, alors je vais étudier".
  - "Je n'étudierai pas".
  - "Par conséquent, il ne pleut pas"

$$\begin{array}{l} p \rightarrow q \\ \neg q \\ \therefore \neg p \end{array}$$

Tautologie correspondante : $(\neg q \land (p \rightarrow q)) \rightarrow \neg p$

p	q	¬q	p → q	¬q ∧ (p → q)	¬p	(¬q ∧ (p → q)) → ¬p
0	0	1	1	1	1	1
0	1	0	1	0	1	1
1	0	1	0	0	0	1
1	1	0	1	0	0	1



Règles d'inférence : Résolution

- La résolution est une règle d'inférence qui permet de vérifier la validité d'un argument comme le Modus Ponens et le Modus Tollens qui affirme que:
  - Si p ou q, et non p ou r, alors q ou r
- Exemple :
  - "Il fait beau ou il fait chaud".
  - "Il ne fait pas chaud ou il fait sec".
  - "Par conséquent, il fait chaud ou sec"

$$\begin{array}{rl} & p \lor q \\ & \neg p \lor r \\ \therefore & q \lor r \end{array}$$

Tautologie correspondante :$(p \lor q) \land (\neg p \lor r) \rightarrow q \lor r$

p	q	r	¬p	¬p ∨ r	p ∨ q	(p ∨ q) ∧ (¬p ∨ r)	q ∨ r	(p ∨ q) ∧ (¬p ∨ r) → q ∨ r
0	0	0	1	1	0	0	0	1
0	0	1	1	1	0	0	1	1
0	1	0	1	1	1	1	1	1
0	1	1	1	1	1	1	1	1
1	0	0	0	0	1	0	0	1
1	0	1	0	1	1	1	1	1
1	1	0	0	0	1	0	1	1
1	1	1	0	1	1	1	1	1

Règles d'inférence : Syllogisme hypothétique

- **Aussi connue comme**: Transitivité
- Si p implique q, et q implique r, alors p implique r
- Exemple :
  - "Il fait chaud quand il y a du soleil".
  - "Il fait sec quand il fait chaud".
  - "Par conséquent, il doit faire sec lorsqu'il y a le soleil"

$$\begin{array}{l} p \rightarrow q \\ q \rightarrow r \\ \therefore\ p \rightarrow r \end{array}$$

Tautologie correspondante :$(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$

p	q	r	$p \rightarrow q$	$q \rightarrow r$	$(p \rightarrow q) \wedge (q \rightarrow r)$	$p \rightarrow r$	$(p \rightarrow q) \wedge (q \rightarrow r) \rightarrow (p \rightarrow r)$
0	0	0	1	1	1	1	1
0	0	1	1	1	1	1	1
0	1	0	1	0	0	1	1
0	1	1	1	1	1	1	1
1	0	0	0	1	0	0	1
1	0	1	0	1	0	1	1
1	1	0	1	0	0	0	1
1	1	1	1	1	1	1	1

Règles d'inférence : Syllogisme disjonctif

- **Aussi connue comme**: Elimination
- Si p ou q, et non p, alors q
- Exemple :
  - "Il fait beau ou chaud".
  - "Il n’y a pas de soleil".
  - "Par conséquent, il fait chaud"

$$p \lor q$$
$$\neg p$$
$$\therefore q$$

Tautologie correspondante : $(p \lor q) \land (\neg p) \rightarrow q$

p	q	p ∨ q	¬p	(p ∨ q) ∧ (¬p)	(p ∨ q) ∧ (¬p) → q
0	0	0	1	0	1
0	1	1	1	1	1
1	0	1	0	0	1
1	1	1	0	0	1

Règles d'inférence : Simplification et Addition

- Simplification : (particularisation)
  - Si p et q alors p
  - Tautologie correspondante : $p \land q \to p$

$$\begin{array}{l} \quad p \land q \\ \therefore p \end{array}$$

- Addition : (généralisation)
  - Si p alors p ou q
  - Tautologie correspondante : $p \to p \lor q$

$$\begin{array}{l} \quad p \\ \therefore p \lor q \end{array}$$

p	q	$p \land q$	$p \land q \to p$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	1

Simplification

p	q	$p \lor q$	$p \to p \lor q$
0	0	0	1
0	1	1	1
1	0	1	1
1	1	1	1

Addition

Règles d'inférence : Conjonction et Contradiction

- Conjonction : (addition conjonctive)
  - Si p et si q alors p et q
  - Tautologie correspondante : $p \land q \rightarrow p \land q$
  -

$$\begin{array}{l} p \\ q \\ \therefore p \land q \end{array}$$

- Contradiction :
  - Si non p implique FAUX alors p
  - Tautologie correspondante : $\neg p \rightarrow F \rightarrow p$

$$\begin{array}{l} \neg p \rightarrow F \\ \therefore p \end{array}$$

p	q	$p \rightarrow q$	$p \land q \rightarrow p \land q$
0	0	1	1
0	1	1	1
1	0	0	1
1	1	1	1

Conjonction

p	q	$\neg p$	F	$\neg p \rightarrow F$	$\neg p \rightarrow F \rightarrow p$
0	0	1	0	0	1
0	1	1	0	0	1
1	0	0	0	1	1
1	1	0	0	1	1

Contradiction

Résumé : Règles d'inférence

Modus Ponens	Modus Tollens	Résolution
$p \rightarrow q$ p $\therefore q$	$p \rightarrow q$ $\neg q$ $\therefore \neg p$	$p \vee q$ $\neg p \vee r$ $\therefore q \vee r$

Simplification	Addition
$p \wedge q$ $\therefore p$	p $\therefore p \vee q$

Syllogisme disjonctif	Syllogisme hypothétique
$p \vee q$ $\neg p$ $\therefore q$	$p \rightarrow q$ $q \rightarrow r$ $\therefore p \rightarrow r$

Conjonction	Contradiction
p q $\therefore p \wedge q$	$\neg p \rightarrow F$ $\therefore p$

Raisonnement déductif à partir d'une base de connaissances

Raisonnement déductif

- Dans la logique propositionnelle, le raisonnement déductif prend la forme de démontrer la conséquence logique entre une base de connaissances et une certaine conclusion

Base de connaissances: Satisfiabilité et Conséquence logique

- Une base de connaissances (KB) est satisfaisable si $M(KB) \neq \emptyset$
  - L'union des modèles satisfaisants sur toutes les propositions de KB est n'est pas un ensemble vide
  - Sinon, la base de connaissances contredit la conclusion
    - Supposons que la conclusion = proposition "p".

- KB implique (Conséquence logique) la conclusion "p" (c'est-à-dire KB $\models$ p) si

$$M(KB \cup \{p\}) = M(KB)$$

  - c'est-à-dire que dans tous les mondes possibles où KB est vrai, p est également vrai

Démonstration de la Conséquence logique de la Base de connaissances

- En pratique, comment procéder ?
- Quels sont les défis auxquels nous sommes confrontés lorsque nous travaillons avec une vaste base de connaissances et/ou des propositions complexes ?
- Méthodes de démonstration de la conséquence logique de la base de connaissances :
  - Vérification de la table de vérité (satisfiabilité)
  - Preuve par contradiction/réfutation
    - Vérification de la table de vérité (insatisfiabilité)
    - La résolution de théorèmes : Détermination de l'insatisfiabilité
    - Résolution (utilise la preuve par contradiction)

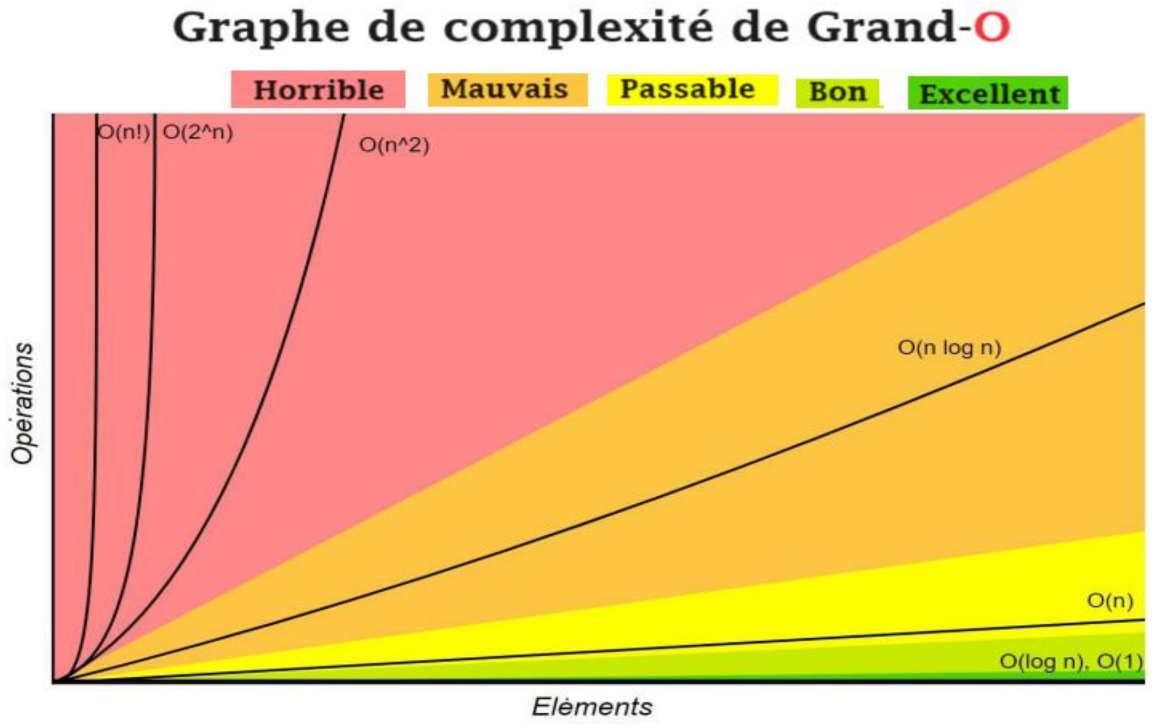
Conséquence logique de la Base de connaissances: Méthode de la table de vérité

Vérification du modèle: énumérer des modèles et montrer que la phrase doit être valable dans tous les modèles.

- Méthode pour calculer si un ensemble de prémisses entraîne logiquement une conclusion
  1. Former une table de vérité pour les constantes propositionnelles apparaissant dans les prémisses et la conclusion; ajouter une colonne pour les prémisses et une colonne pour la conclusion
  2. Evaluez les prémisses pour chaque ligne du tableau.
  3. Évaluer la conclusion pour chaque ligne du tableau
  4. Si *chaque ligne qui satisfait aux prémisses satisfait également à la conclusion*, alors les prémisses entraînent (implique) logiquement la conclusion.

Inconvénient de la méthode de la table de vérité

- Étant donné n constantes propositionnelles, il peut prendre 2^n temps pour effectuer l'inférence avec une table de vérité
  - La complexité temporelle de cet algorithme serait O(2^n)
  - Pourrait être infaisable même pour n = 100



Conséquence logique de la Base de connaissances: Preuve par contradiction

- **Aussi connue comme**: Preuve par réfutation

$KB \models q$ **si et seulement si** $KB \land \neg q$ **est insatisfaisant**

- **D'où cela vient-il ?**
  - **Théorème de déduction** $P \overset{val}{\models} Q$ si, et seulement si, $P \Rightarrow Q$ est une tautologie

  - **Ainsi, on sait que** $\neg(KB \rightarrow q)$ **est insatisfaisant :**
    - Négation d'une tautologie

  - **Dériver par équivalence :**
    - $\neg(KB \rightarrow q) \equiv \neg(\neg KB \lor q) \equiv KB \land \neg q$

Implication:
$P \Rightarrow Q \overset{val}{=\!=} \neg P \lor Q$

De Morgan:
$\neg(P \land Q) \overset{val}{=\!=} \neg P \lor \neg Q,$
$\neg(P \lor Q) \overset{val}{=\!=} \neg P \land \neg Q$

Démonstration de l'insatisfaisabilité

$KB \models q$ si et seulement si $KB \wedge \neg q$ est insatisfaisant

- Ajouter ¬q à KB et en déduire une contradiction (insatisfaisable)
- Méthode pour calculer si un ensemble de prémisses entraîne logiquement une conclusion
  1. Former une table de vérité pour les constantes propositionnelles
  2. Pour chaque phrase de l'ensemble et chaque ligne de la table de vérité, vérifier si la ligne satisfait la phrase. Si ce n'est pas le cas, barrez la ligne.
  3. Si toutes les lignes sont barrées, alors $KB \models q$ ✔

Déterminer la satisfiabilité des phrases en logique propositionnelle a été le premier problème dont il a été prouvé qu'il était NP-complet.

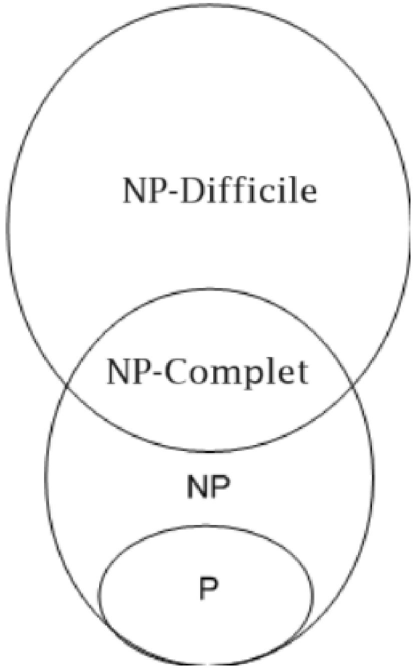
Démonstration de l'insatisfaisabilité (Cont.)

- En théorie de la complexité, un problème *NP-complet* ou problème NPC (c'est-à-dire un problème complet pour la classe NP) est un problème de décision vérifiant les propriétés suivantes :
  - il est possible de vérifier une solution efficacement (en temps polynomial) ; la classe des problèmes vérifiant cette propriété est notée NP
  - tous les problèmes de la classe NP se ramènent à celui-ci via une réduction polynomiale ; cela signifie que le problème est au moins aussi difficile que tous les autres problèmes de la classe NP.

Classes de complexité dans la résolution de problèmes

- Faire allusion aux défis dans la recherche de décisions, de conclusions, de solutions

- **P** : peut être résolu sur une machine de Turing déterministe en temps polynomial $T(n) = O(n^k)$
- **NP** : peut être résolu sur une machine de Turing non déterministe en temps polynomial
- **NP-Complet** : peut être résolu par une classe restreinte d'algorithmes de force brute en temps polynomial.
- **NP-Difficile** : difficulté en temps polynomial non déterministe
  - Au moins aussi difficile que les problèmes les plus difficiles de NP



Plus d'informations sur la démonstration de l'insatisfaisance

- Peut-on éviter la table de vérité ? Oui !

- **Preuve du théorème** : appliquer des règles d'inférence directement aux phrases de notre base de connaissances pour construire une preuve de la phrase souhaitée sans consulter de modèles

- Cela peut être fait à la main (slide qui suit), ou nous pouvons appliquer un algorithme de recherche afin de trouver une séquence d'étapes qui ne soit pas trop complexe et qui constitue la preuve.

- $p \Leftrightarrow (q \vee r), \neg p \quad \models \quad \neg q \quad ?$

Conséquence logique de la Base de connaissances: Preuve du théorème

$p \Leftrightarrow (q \vee r), \neg p \models \neg q$

En utilisant un mélange d'équivalences et de règles d'inférence

- $p \Leftrightarrow (q \vee r)$
- $(p \rightarrow (q \vee r)) \wedge ((q \vee r) \rightarrow p)$
- $(q \vee r) \rightarrow p$
- $\neg p \rightarrow \neg(q \vee r)$
- $\neg(q \vee r)$
- $\neg q \wedge \neg r$

Preuve de conséquence logique ✔

Bi-implication:

$P \Leftrightarrow Q \overset{val}{=\!=} (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

Simplification

$p \wedge q$

$\therefore p$

Contraposition:

$P \Rightarrow Q \overset{val}{=\!=} \neg Q \Rightarrow \neg P$

$\neg p$

Modus Ponens

$p \rightarrow q$

p

$\therefore q$

De Morgan:

$\neg(P \wedge Q) \overset{val}{=\!=} \neg P \vee \neg Q,$

$\neg(P \vee Q) \overset{val}{=\!=} \neg P \wedge \neg Q$

Conséquence logique de la base de connaissance: Résolution

- Les algorithmes d'inférence qui recherchent des preuves doivent se préoccuper de la l'exhaustivité (complétude)
  - c'est-à-dire que les règles d'inférence disponibles sont-elles suffisantes pour dériver la preuve ?
- Alternativement, il existe une seule règle d'inférence (règle de résolution) :
  - Elle produit un algorithme d'inférence complet lorsqu'elle est associée à un algorithme de recherche complet.
- Règle de résolution généralisée :
  - Autorise les clauses comportant un nombre arbitraire de littéraux.

$$\begin{array}{rl} & a \vee b \\ & \neg b \vee c \\ \therefore & a \vee c \end{array} \quad \overset{val}{\equiv} \quad \begin{array}{rl} & a \vee b \\ & b \rightarrow c \\ \therefore & a \vee c \end{array} \qquad \begin{array}{rl} & (a_1 \vee \ldots \vee a_m \vee b) \\ & (\neg b \vee c_1 \vee \ldots \vee c_n) \\ \therefore & (a_1 \vee \ldots \vee a_m \vee c_1 \vee \ldots \vee c_n) \end{array}$$

Règle de résolution — Généralisation de Modus Ponens — Règle de résolution généraliée

Forme normale conjonctive (FNC)

- Une forme normale conjonctive est l'écriture d'une proposition comme *conjonction de disjonctions de littéraux*, c'est-à-dire que la proposition est écrite comme la conjonction (avec le symbole ET $\wedge$) d'une disjonction (avec le symbole OU $\vee$) de littéraux (c'est-à-dire des variables ou la négation $\neg$ de ces variables)

Forme normale conjonctive (FNC) (Cont.)

- Pour que les systèmes de preuve automatique restent aussi simples que possible, ils sont généralement conçus pour fonctionner sur des formules en forme normale conjonctive.
  - Une formule est en FNC si et seulement si elle consiste en une conjonction de clauses $(K_1 \wedge K_2 \wedge \ldots \wedge K_m)$
  - Une clause K_i consiste en une disjonction de littéraux $(L_1 \vee L_2 \vee \ldots \vee L_m)$.
  - Un littéral est une proposition positive ou négative
- Exemple: $(a \vee b \vee \neg c) \wedge (a \vee b) \wedge (\neg b \vee \neg c)$.
- Toute formule de logique propositionnelle peut être transformée en une formule FNC équivalente.

Convertir une proposition en FNC

- Convertir la proposition en forme normale conjonctive

Exemple : $a \vee b \rightarrow c \wedge d$

$\neg(a \vee b) \vee (c \wedge d)$

$(\neg a \wedge \neg b) \vee (c \wedge d)$

$(\neg a \vee (c \wedge d)) \wedge (\neg b \vee (c \wedge d))$

$((\neg a \vee c) \wedge (\neg a \vee d)) \wedge ((\neg b \vee c) \wedge (\neg b \vee d))$

$(\neg a \vee c) \wedge (\neg a \vee d) \wedge (\neg b \vee c) \wedge (\neg b \vee d)$

Implication:

$P \Rightarrow Q \overset{val}{=\!=} \neg P \vee Q$

Règle de Morgan :

$\neg(P \wedge Q) \overset{val}{=\!=} \neg P \vee \neg Q,$

$\neg(P \vee Q) \overset{val}{=\!=} \neg P \wedge \neg Q$

Distributivité :

$P \wedge (Q \vee R) \overset{val}{=\!=} (P \wedge Q) \vee (P \wedge R),$

$P \vee (Q \wedge R) \overset{val}{=\!=} (P \vee Q) \wedge (P \vee R)$

Associativité :

$(P \wedge Q) \wedge R \overset{val}{=\!=} P \wedge (Q \wedge R),$

$(P \vee Q) \vee R \overset{val}{=\!=} P \vee (Q \vee R),$

Simplification de la FNC en FNC pure

- Une clause vide est fausse (F)
  - Aucune option à satisfaire
  - La phrase est donc fausse, car il s'agit d'une exigence impossible à satisfaire.

$$() \wedge (\neg a \vee d) \wedge (\neg b \vee c) \equiv F$$

- Une phrase sans clause est Vraie
  - Aucune exigence

$$() \equiv V$$

- Les clauses valides peuvent être supprimées dès leur apparition

$$(\neg a \vee d \vee a) \wedge (\neg b \vee c) \equiv (\neg b \vee c)$$

- Les littéraux répétés dans une même clause peuvent être simplifiés
  - Factorisation

$$(a \vee d \vee a) \wedge (\neg b \vee c) \equiv (a \vee d) \wedge (\neg b \vee c)$$

- Si une clause *c* est incluse dans une autre clause *c'* (*c subsume c'*), la clause *c'* peut être supprimée.

$$(a \vee d) \wedge (a \vee d \vee c) \equiv (a \vee d \vee c)$$

Conséquence logique de la base de connaissance: Algorithme de Résolution

- **Rappelons-le** : Pour démontrer que KB implique q, nous pouvons montrer que (KB ∧ ¬q) est insatisfaisable (preuve par contradiction).
- **Algorithme de résolution** :
  - Toutes les phrases de KB sont converties en FNC et sont connectées conjonctivement avec la conjecture (¬q).
  - Règle de résolution appliquée à chaque paire unique de clauses
    - De nouvelles clauses sont ajoutées à l'ensemble des clauses.
  - Les clauses qui se résolvent en Vrai peuvent être écartées de tout examen ultérieur considération
  - Le processus se poursuit jusqu'à ce que
    1. Deux clauses se résolvent en une clause vide, c'est-à-dire FAUX (KB implique q)
    2. Aucune nouvelle clause ne peut être ajoutée (KB n'implique pas q)

Exemple d'algorithme de résolution

- KB: (¬c ∨ d) , (¬d ∨ ¬q ∨ c) , (q ∨ d) , (¬d)

- **FNC** $\boxed{(\neg c \lor d) \land (\neg d \lor \neg q \lor c) \land (q \lor d) \land (\neg d) \land \neg q}$

~~(d ∨ ¬q ∨ ¬d)~~	~~(¬q ∨ c ∨ ¬c)~~	~~(¬d ∨ c ∨ d)~~	~~(¬q ∨ c ∨ q)~~	(¬c)	(q)
VRAI	**VRAI**	**VRAI**	**VRAI**		

Dans l'ensemble des clauses résultantes y a-t-il des clauses qui se résolvent en un ensemble vide ?

$\boxed{(q) \land \neg q} \xrightarrow{\text{Règle de résolution}} \boxed{(\ )}$

(KB implique q) ✔

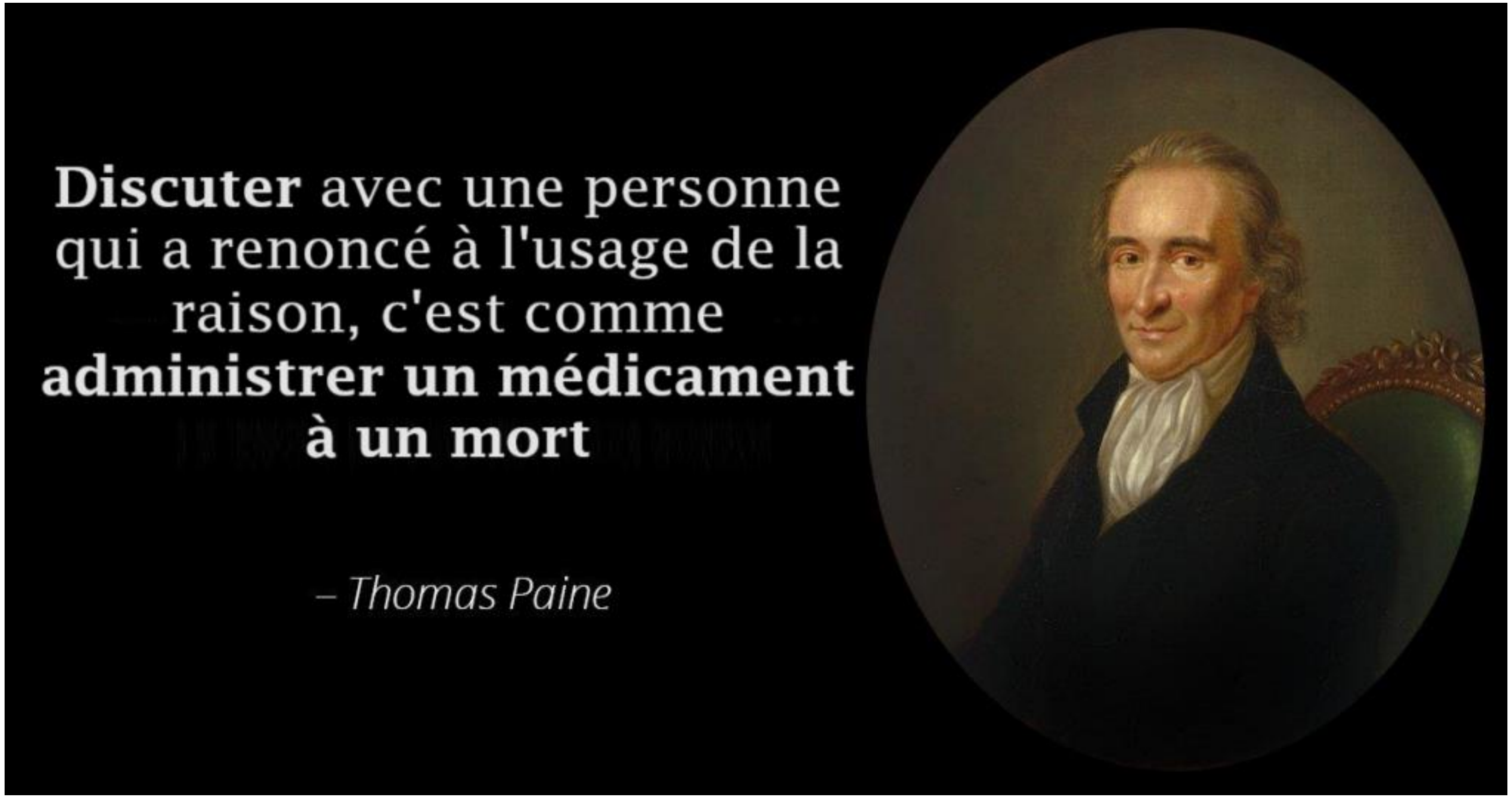
Exemple d'algorithme de résolution (Cont.)

$$S = \{p \overset{1}{\vee} q, p \overset{2}{\vee} r, \neg q \overset{3}{\vee} \neg r, \overset{4}{\neg p}\}$$

#	clause	de
5	$p \vee \neg r$	(1, 3)
6	q	(1, 4)
7	$p \vee \neg q$	(2, 3)
8	r	(2, 4)
9	p	(2, 5)
10	$\neg r$	(3, 6)
11	$\neg q$	(3, 8)
12	$\neg r$	(4, 5)
13	$\neg q$	(4, 7)
14	F	(4, 9)

Résumé de la logique propositionnelle

1. Une représentation formelle des connaissances définie avec précision
    - Propositions et connecteurs
2. Vérité propositionnelle
3. Règles de calcul avec des propositions abstraites :
    - Équivalences
    - Règles d'inférence (propositions multiples)
4. Raisonnement :
- Conséquence logique (une forme précise d'inférence)
- Arguments (combinaison de plusieurs propositions pour tirer une conclusion)
- Raisonnement déductif : preuves à partir d'une base de connaissances


Discuter avec une personne
qui a renoncé à l'usage de la
raison, c'est comme
administrer un médicament
à un mort
– *Thomas Paine*

Travail pratique 2: Rappel

- **Objectifs :**

  - Logique propositionnelle et logique du premier ordre
    - S'entraîner à travailler avec la syntaxe et la sémantique de la logique
    - S'entraîner à manipuler des expressions logiques de manière "algébrique".
    - Appliquer la logique pour faire des déductions et déterminer la "vérité".
    - Comprendre deux des approches d'inférences les plus simples sur lesquelles nous nous concentrerons principalement dans ce cours :
      - Le chaînage avant
      - Le chaînage arrière