Aussagenlogik

Übersicht:

1. Teil 1: Syntax und Semantik
2. Teil 2: Modellierung und aussagenlogische Beweise
3. Teil 3: Elementare Beweistechniken I

Logik

Der Begriff “Logik” (griechisch $\lambda o \gamma \iota \kappa \eta$ [logiké]—denkende (Kunst, Vorgehensweise)) leitet sich vom griechischen Wort “logos” (die Sprache, das Wort) ab und bezeichnet die Lehre der **formalen Beziehungen zwischen Denkinhalten** sowie des vernünftigen, **folgerichtigen Schließens**.

In der Logik wird die Gültigkeit von Argumenten abhängig von ihrer Struktur, jedoch unabhängig vom konkreten Inhalt der Aussagen, untersucht.

Man spricht deshalb auch häufig von “formaler” Logik.

Ursprünge der Logik

- Die Ursprünge der Logik gehen auf das antike Griechenland (Aristoteles) zurück, wo man im Kontext von philosophischer Diskussion und Rhetorik an der Formalisierung der **Deduktion** interessiert war, der Ableitung gültiger **Konklusionen** aus angenommenen **Prämissen**.
- Hierzu wurden **Deduktionsregeln** entwickelt. Beispiel:

Prämisse	Alle Menschen sind sterblich
Prämisse	X ist ein Mensch
Konklusion	X ist sterblich

- Die **Syllogistik** (Schlusslehre) ist ein formales logisches System im modernen Sinn, in dem Argumente starrer Struktur, Syllogismen genannt, untersucht werden.
- Beispiele für Syllogismen sind der **Modus ponens** (gilt A impliziert B und weiterhin A, so folgt B) und der **Modus Barbara** (gilt A impliziert B sowie B impliziert C, so folgt A impliziert C).

Mathematische Logik

- Seine formale Gestalt wurde der heute bekannten Logik (symbolische Logik, formale Logik im engeren Sinn) durch Entwicklungen in der Mathematik verliehen.
- Diese begannen, nachdem die Aristotelische Logik bereits im 17. Jahrhundert durch **Leibnitz** zu einem gewissen Grad mathematisiert wurde.
- Mitte des 19. Jahrhunderts, insbesondere durch Arbeiten von **George Boole** (The Mathematical Analysis of Logic, 1847; Laws of Thought, 1854), **Augustus De Morgan** (Formal Logic, 1847), **Gottlob Frege**, Begriffsschrift, 1879).
- Der eigentliche Ausbau der symbolischen Logik erfolgte im 20. Jahrhundert, beginnend mit Arbeiten von **Bertrand Russell** (der im Jahr 1903 zusammen mit **Alfred North Whitehead** die Principia Mathematica veröffentlichte) und **David Hilbert**.

Logik in der Informatik

- Die Logik spielt heutzutage auch in der Informatik eine zentrale Rolle. Die Informatik ist “Anwender” der mathematischen Logik, hat aber auch viele Impulse zur Entwicklung neuer **logischer Systeme** geliefert.
- Das Verhältnis zwischen mathematischer Logik und Informatik ist vergleichbar zum Verhältnis von analytischer Mathematik und Physik.
- Die Anwendungen innerhalb der Informatik sind vielfältig: Logisches Design von Schaltungen und Hardwareentwurf (Aussagenlogik), Programmspezifikation und -verifikation, automatisches Beweisen und logische Programmierung, Wissensmodellierung und intelligente Systeme, etc.

Einführendes Beispiel

Falls Lisa Peter trifft, dann trifft Lisa auch Gregor.
Lisa trifft Peter oder Lisa trifft Marko.
Gregor trifft nicht Maria, aber Marko trifft Maria.

Frage: Trifft Lisa Peter?

Einführendes Beispiel

Falls Lisa Peter trifft, dann trifft Lisa auch Gregor.
$L_P \rightarrow L_G$
Lisa trifft Peter oder Lisa trifft Marko.
$L_P \vee L_M$
Gregor trifft nicht Maria, aber Marko trifft Maria.
$(\neg G_M) \wedge M_M$

Frage: Trifft Lisa Peter?
Folgt L_P?

Abkürzungen:
L_P : Lisa trifft Peter
L_G : Lisa trifft Gregor
L_M : Lisa trifft Marko
M_M : Marko trifft Maria
G_M : Gregor trifft Maria

Syntax

1. Aussagenlogische Atome (Elementaraussagen): Großbuchstaben mit oder ohne Indizes; $A, B, X, \ldots, Y_i, \ldots$

   Atome werden auch als **aussagenlogische Variablen** bezeichnet.

   Eine Menge V von Atomen wird vorgegeben.

2. Aussagenlogische Formeln: Griechische Buchstaben; $\alpha, \beta, \phi, \ldots$
3. Junktoren:
   1. $\wedge$ Konjunktion (und)
   2. $\vee$ Disjunktion (oder)
   3. $\neg$ Negation (nicht)
   4. $\rightarrow$ Implikation
   5. $\leftrightarrow$ Äquivalenz
   6. **true** konstanter Wert
   7. **false** konstanter Wert

Aussagenlogische Formeln

Definition 1

Aussagenlogische Formeln über Atomen in V werden induktiv definiert:

1. Aussagenlogische Atome (Variablen) aus V sind Formeln.
2. Ist α eine Formel, dann ist auch $(\neg\alpha)$ eine Formel.
3. Sind α und β Formeln, dann sind auch $(\alpha \wedge \beta)$ und $(\alpha \vee \beta)$ Formeln.
4. Formeln werden nur mit (1) bis (3) gebildet.

1. Die Menge der in einer Formel vorkommenden Atome wird mit $\mathit{atoms}(\alpha)$ bezeichnet.
2. α ist eine Formel über $\mathit{atoms}(\alpha)$.

Aussagenlogische Formeln

Definition 1

Aussagenlogische Formeln über Atomen in V werden induktiv definiert:

1. Aussagenlogische Atome (Variablen) aus V sind Formeln.
2. Ist α eine Formel, dann ist auch $(\neg\alpha)$ eine Formel.
3. Sind α und β Formeln, dann sind auch $(\alpha \wedge \beta)$ und $(\alpha \vee \beta)$ Formeln.
4. Formeln werden nur mit (1) bis (3) gebildet.

Beispiele:

1. $\alpha = ((\neg A) \vee (B \wedge (\neg B)))$ ist eine Formel.

2. $\beta = (((\neg(\neg A)) \vee B) \wedge (\neg X))$ ist eine Formel.

3. $\neg A$ ist (bisher) keine Formel, da Klammern fehlen.

4. $A \to B$ ist (bisher) keine Formel.

Bindungsregeln

Bindungsregeln dienen der Klammer-Ersparnis:

1. $\neg$ bindet stärker als $\wedge$
2. $\wedge$ bindet stärker als $\vee$
3. Bindungsregeln sind transitiv
4. Linksassoziativität für alle Junktoren: $A \wedge B \wedge C$ entspricht $((A \wedge B) \wedge C)$

Beispiele:

1. $\neg A \wedge B$ entspricht der Formel $((\neg A) \wedge B)$
2. $A \wedge B \vee \neg X$ entspricht $((A \wedge B) \vee (\neg X))$
3. $\neg\neg\neg A$ entspricht $(\neg(\neg(\neg A)))$
4. $A \wedge B \wedge C \vee X$ entspricht $(A \wedge B \wedge C) \vee X$

   $(A \wedge B \wedge C) \vee X$ entspricht $(((A \wedge B) \wedge C) \vee X)$

Aussagenlogische Formeln

Abkürzungen:

1. $\alpha \to \beta := \neg\alpha \vee \beta$
2. $\alpha \leftrightarrow \beta := (\alpha \to \beta) \wedge (\beta \to \alpha)$
3. $\textbf{true} := \mathsf{A} \vee \neg \mathsf{A}$
4. $\textbf{false} := \mathsf{A} \wedge \neg \mathsf{A}$

Bindungsregeln für (binäre) Abkürzungen:

1. $\vee$ bindet stärker als $\to$.
2. $\to$ bindet stärker als $\leftrightarrow$.

Binäre Operatoren gleicher Stärke werden als links geklammert angesehen. (linksassoziativ)

Semantik

Definition 2

Sei V eine Menge von Atomen (Variable). Eine Abbildung $I : V \longrightarrow \{w, f\}$ ist eine Bewertung. Eine Bewertung wird auch Interpretation genannt.

Definition 3

Bewertung von Formeln:
Sei I eine Bewertung von Atomen in V. Wir erweitern die Bewertung I auf Formeln über V wie folgt:

1. $I(\neg\alpha) = w$ genau dann, wenn $I(\alpha) = f$.
2. $I(\alpha \wedge \beta) = w$ genau dann, wenn $I(\alpha) = I(\beta) = w$.
3. $I(\alpha \vee \beta) = w$ genau dann, wenn $I(\alpha) = w$ oder $I(\beta) = w$.

Semantik

Erweiterung über Wahrheitstafeln:

$I(\alpha)$	$I(\neg\alpha)$
f	w
w	f

$I(\alpha)$	$I(\beta)$	$I(\alpha \wedge \beta)$
f	f	f
f	w	f
w	f	f
w	w	w

$I(\alpha)$	$I(\beta)$	$I(\alpha \vee \beta)$
f	f	f
f	w	w
w	f	w
w	w	w

Beispiel:

$\alpha = A \wedge (\neg A \vee B)$

Sei $I(A) = w$ und $I(B) = w$ eine Bewertung. Dann gilt $I(\alpha) = w$.

$I(\alpha) = w$ gdw. $I(A) = w$ und $I(\neg A \vee B) = w$

gdw. $I(A) = w$ und ($I(\neg A) = w$ oder $I(B) = w$).

gdw. $I(A) = w$ und ($I(A) = f$ oder $I(B) = w$).

Alternativ:

$$
\begin{aligned}
I(\alpha) &= I(I(A) \wedge I(I(\neg A) \vee I(B))) \\
&= I(w \wedge I(I(\neg A) \vee w)) \\
&= I(w \wedge I(f \vee w)) \\
&= I(w \wedge w) \\
&= w
\end{aligned}
$$

Wahrheitstafel

Wir möchten wissen ob eine Formel $\alpha = (A \vee B) \wedge (\neg A \vee B)$ erfüllbar ist. Dazu erzeugen wir eine Wahrheitstabelle, in der wir die Bestandteile von α sukzessive auswerten.

A	B	C	$(A \vee B)$	$(\neg A \vee B)$	α
f	f	f	f	w	f
f	f	w	f	w	f
f	w	f	w	w	w
f	w	w	w	w	w
w	f	f	w	f	f
w	f	w	w	f	f
w	w	f	w	w	w
w	w	w	w	w	w

Grundbegriffe

Definition 4

Sei α eine Formel.

1. α ist **erfüllbar** gdw. es eine Bewertung I mit $I(\alpha) = w$ gibt.
2. α ist eine **Tautologie** gdw. für alle Bewertungen $I(\alpha) = w$ gilt.
3. α ist **widerspruchsvoll** gdw. für alle Bewertungen $I(\alpha) = f$ gilt.
4. α ist **falsifizierbar** gdw. es eine Bewertung mit $I(\alpha) = f$ gibt.

Beispiele:

1. $A \vee \neg B$ ist erfüllbar, z.B. mit $I(A) = w, I(B) = w$
2. $A \vee \neg A$ ist eine Tautologie, d.h. wahr für alle Bewertungen.
3. $A \wedge \neg A$ ist widerspruchsvoll, d.h. falsch für alle Bewertungen.
4. $A \vee B$ ist falsifizierbar, z.B. durch $I(A) = I(B) = f$

Semantik

Satz 5

Sei α eine Formel.

α ist widerspruchsvoll
genau dann, wenn α nicht erfüllbar ist
genau dann, wenn $\neg\alpha$ eine Tautologie ist.

Beweis

Sei α widerspruchsvoll. Dann gilt per definitionem, dass $I(\alpha) = f$ für alle Bewertungen. Es gibt also keine Bewertung, die α wahr macht, sodass α nicht erfüllbar ist. Damit gilt auch für alle Bewertungen $I(\neg\alpha) = w$. Also ist $\neg\alpha$ eine Tautologie.
Sei nun $\neg\alpha$ eine Tautologie. Dann gilt per definitionem, dass $I(\neg\alpha) = w$ und damit $I(\alpha) = f$ für alle Belegungen. Die Formel α ist also nicht erfüllbar und somit widerspruchsvoll.

Semantische Folgerung

Definition 6

Sei M eine nicht-leere Menge von Formeln und β eine Formel.
β **folgt semantisch** aus M, in Zeichen $M \models \beta$, genau dann, wenn für alle Bewertungen I, für die $I(M) = w$ gilt, auch $I(\beta) = w$ gilt.
Enthält M nur eine Formel α, so schreiben wir auch $\alpha \models \beta$.
Ist M leer, dann folgt β gdw. β eine Tautologie ist. Dann schreiben wir $\models \beta$.

Schreibweise:
Seien $\alpha_1, \ldots, \alpha_n$ Formeln, dann schreiben wir für $M = \{\alpha_1, \ldots, \alpha_n\}$ auch $\alpha_1 \wedge \ldots \wedge \alpha_n$ und umgekehrt.

Semantische Folgerung

Satz 7

Seien α und β Formeln.

1. $\alpha \models \beta$ *gdw.* $\models \alpha \rightarrow \beta$ *gdw.* $\models \neg\alpha \vee \beta$
2. $\alpha \models \beta$ *gdw.* $\alpha \wedge \neg\beta$ *ist widerspruchsvoll.*
3. α *ist widerspruchsvoll gdw.* $\alpha \models \sigma$ *für alle Formeln* σ *gilt.*
4. α *ist widerspruchsvoll gdw. es ein Atom* A *mit* $\alpha \models (A \wedge \neg A)$ *gibt.*

Satz 8

Seien α, β und σ Formeln, dann gilt

1. $\alpha, \alpha \to \beta \models \beta$ *(Modus Ponens)*
2. $\alpha \to \beta, \beta \to \sigma \models \alpha \to \sigma$ *(Kettenregel)*
3. $(\alpha \vee \beta), (\neg\beta \vee \sigma) \models (\alpha \vee \sigma)$

Beweis

Ad 1: Sei I eine Bewertung mit $I(\alpha \wedge (\alpha \to \beta)) = w$. Zu zeigen: $I(\beta) = w$. Aus $I(\alpha \wedge (\alpha \to \beta)) = w$ ergibt sich $I(\alpha) = w$ und $I(\alpha \to \beta) = w$, und damit $I(\neg\alpha \vee \beta) = w$. Also gilt $I(\beta) = w$.

Ad 2: Sei I eine Bewertung mit $I((\alpha \to \beta) \wedge (\beta \to \sigma)) = w$. Zu zeigen: $I(\alpha \to \sigma) = w$.
Fallunterscheidung:
Fall 1: $I(\beta) = w$, dann muss $I(\sigma) = w$ gelten wegen $I(\beta \to \sigma) = w$.
Fall 2: $I(\beta) = f$, dann muss $I(\alpha) = f$ wegen $I(\alpha \to \beta) = w$ gelten.

q.e.d.

Logische Äquivalenz

Definition 9

Zwei Formeln α und β sind **logisch äquivalent**, in Zeichen $\alpha \approx \beta$, genau dann, wenn für alle Bewertungen $I(\alpha) = I(\beta)$ gilt.

Satz 10

Seien α und β Formeln, dann gilt:
$\alpha \approx \beta$ genau dann, wenn $\alpha \models \beta$ und $\beta \models \alpha$ gilt.

Umformungsregeln - Teil 1

Seien α und β aussagenlogische Formeln. Es gelten folgende Regeln:

Negation	$\neg\neg\alpha \approx \alpha$
Idempotenz	$\alpha \vee \alpha \approx \alpha$
	$\alpha \wedge \alpha \approx \alpha$
Neutrales Element	$(\alpha \wedge \neg\alpha) \vee \beta \approx \beta$
	$(\alpha \vee \neg\alpha) \wedge \beta \approx \beta$
Kontradiktion/Tautologie	$(\alpha \wedge \neg\alpha) \wedge \beta \approx \alpha \wedge \neg\alpha$
	$(\alpha \vee \neg\alpha) \vee \beta \approx \alpha \vee \neg\alpha$

Die Gültigkeit der Regeln lässt sich leicht mit Wahrheitstafeln zeigen.

Umformungsregeln - Teil 2

Seien α, β und γ aussagenlogische Formeln. Es gelten folgende Regeln:

Kommutativität	$\alpha \vee \beta \approx \beta \vee \alpha$
	$\alpha \wedge \beta \approx \beta \wedge \alpha$
Assoziativität	$(\alpha \vee \beta) \vee \gamma \approx \alpha \vee (\beta \vee \gamma)$
	$(\alpha \wedge \beta) \wedge \gamma \approx \alpha \wedge (\beta \wedge \gamma)$
Distributivität	$(\alpha \wedge \beta) \vee \gamma \approx (\alpha \vee \gamma) \wedge (\beta \vee \gamma)$
	$(\alpha \vee \beta) \wedge \gamma \approx (\alpha \wedge \gamma) \vee (\beta \wedge \gamma)$
De Morgan	$\neg(\alpha \wedge \beta) \approx \neg\alpha \vee \neg\beta$
	$\neg(\alpha \vee \beta) \approx \neg\alpha \wedge \neg\beta$
Absorption	$\alpha \wedge (\alpha \vee \beta) \approx \alpha$
	$\alpha \vee (\alpha \wedge \beta) \approx \alpha$

Die Gültigkeit der Regeln lässt sich leicht mit Wahrheitstafeln zeigen.

Normalformen

Definition 11

Negationsnormalform (NNF)
Eine Formel α (ohne Implikations- und Äquivalenzzeichen) ist in **Negationsnormalform** (NNF) genau dann, wenn jedes Negationszeichen direkt vor einem Atom steht.

Beispiel:
1. $(\neg X \vee Y) \wedge (\neg A \vee B)$ ist in NNF
2. $\neg\neg A$ und $\neg(A \vee B)$ sind nicht in NNF

Normalformen

Definition 11

Negationsnormalform (NNF)
Eine Formel α (ohne Implikations- und Äquivalenzzeichen) ist in **Negationsnormalform** (NNF) genau dann, wenn jedes Negationszeichen direkt vor einem Atom steht.

Beispiel:
1. $(\neg X \vee Y) \wedge (\neg A \vee B)$ ist in NNF
2. $\neg\neg A$ und $\neg(A \vee B)$ sind nicht in NNF

Satz 12

Zu jeder Formel α gibt es eine äquivalente Formel β in NNF.

Transformation:

1. Ersetze Teilformeln $\neg\neg\sigma$ durch σ (Negation).
2. Ersetze Teilformeln $\neg(\sigma \vee \beta)$ durch $(\neg\sigma \wedge \neg\beta)$ (de Morgan).
3. Ersetze Teilformeln $\neg(\sigma \wedge \beta)$ durch $(\neg\sigma \vee \neg\beta)$ (de Morgan).

Normalformen

Definition 13

Literal: negiertes oder nicht-negiertes Atom
positives Literal: Atom, z.B. A
negatives Literal: negiertes Atom, z.B. $\neg A$
Klausel: Disjunktion von Literalen, z.B. $(A \vee B \vee \neg D)$
k-Klausel: Klauseln, die maximal k Literale enthalten
Unit-Klausel: Klausel, die aus einem Literal besteht

Definition 14

Konjunktive Normalform (KNF)
Seien $\alpha_1, \ldots, \alpha_n$ Klauseln. Dann ist $\alpha = \alpha_1 \wedge \ldots \wedge \alpha_n$ in konjunktiver Normalform (KNF). Eine Formel ist in KNF genau dann, wenn sie aus einer Konjunktion von Klauseln besteht.

Alternative Schreibweise: Sei $\alpha = \alpha_1 \wedge \ldots \wedge \alpha_n$ in KNF, dann schreiben wir auch $\alpha = \{\alpha_1, \ldots, \alpha_n\}$.
Beispiel: $\alpha = \{(A \vee \neg B), (\neg A \vee D)\}$

Satz 15

Zu jeder Formel α gibt es eine logisch äquivalente Formel β in KNF.

Transformation:

1. Transformiere α in eine äquivalente Formel σ in NNF.
2. Wende auf σ evtl. mehrfach das Distributivgesetz an.

Beispiel:

1. $\neg(A \wedge (\neg B \vee \neg(\neg C \wedge E) \vee A))$
2. $\neg A \vee \neg(\neg B \vee \neg(\neg C \wedge E) \vee A)$ NNF-Schritt
3. $\neg A \vee (B \wedge (\neg C \wedge E) \wedge \neg A)$ in NNF
4. $(\neg A \vee B) \wedge (\neg A \vee (\neg C \wedge E)) \wedge (\neg A \vee \neg A)$ Distributivgesetz
5. $(\neg A \vee B) \wedge (\neg A \vee \neg C) \wedge (\neg A \vee E) \wedge (\neg A \vee \neg A)$ Distributivgesetz

Länge der Formel

Definition 16

Die Länge einer Formel α, in Zeichen $|\alpha|$, ist die Anzahl der Atomvorkommen in α.

Beispiel:
$|\neg\neg A| = 1$ und $|A \vee (A \wedge B)| = 3$

Satz 17

Es gibt aussagenlogische Formeln α_n der Länge $2n$, zu der jede logisch äquivalente Formel in KNF mindestens die Länge 2^n besitzt.

Disjunktive Normalform

Definition 18

Eine Formel $\alpha = (L_1 \wedge \ldots \wedge L_n)$ mit Literalen $L_1, \ldots, L_n$ wird als **Monom** bezeichnet.
Ein Monom mit maximal k Literalen bezeichnen wir als k-**Monom**.
Eine Formel α ist in disjunktiver Normalform (DNF) genau dann, wenn α eine Disjunktion von Monomen ist. ($\alpha = \alpha_1 \vee \ldots \vee \alpha_n$ und α_i ist ein Monom für $1 \leq i \leq n$.)

Satz 19

Zu jeder Formel gibt es eine logisch äquivalente Formel in DNF.

Aussagenlogik

Übersicht:

1. Teil 1: Syntax und Semantik
2. Teil 2: **Modellierung und aussagenlogische Beweise**
3. Teil 3: Elementare Beweistechniken I

Modellierungsaufgabe

Es gibt drei Tauben und zwei Löcher. Jede Taube soll in ein Loch. Aber es passt nur eine Taube in ein Loch.
Gibt es eine Lösung?

Ansatz 1: (Funktion)

Die Tauben seien T_1, T_2 und T_3 und die Löcher seien L_1 und L_2.
Gibt es eine totale, injektive Abbildung $f : \{T_1, T_2, T_3\} \rightarrow \{L_1, L_2\}$?

Es gibt keine Lösung, da sowohl Definitions- als auch Bildbereich endlich sind, letzterer jedoch kleinere Kardinalität besitzt als ersterer.

Modellierungsaufgabe

Es gibt drei Tauben und zwei Löcher. Jede Taube soll in ein Loch. Aber es passt nur eine Taube in ein Loch.
Gibt es eine Lösung?

Ansatz 2: (Aussagenlogik)

$X_{i,j}$ Atom für $1 \leq i \leq 3, 1 \leq j \leq 2$. Idee: Taube i ist im Loch j.

$\alpha = (X_{1,1} \vee X_{1,2})$ Taube T_1 ist im Loch L_1 oder im Loch L_2
$\quad \wedge (X_{2,1} \vee X_{2,2}) \wedge (X_{3,1} \vee X_{3,2})$
$\quad \wedge \neg(X_{1,1} \wedge X_{2,1})$ in Loch L_1 sind nicht beide Tauben T_1 und T_2
$\quad \wedge \neg(X_{1,1} \wedge X_{3,1}) \wedge \neg(X_{2,1} \wedge X_{3,1})$
$\quad \wedge \neg(X_{1,2} \wedge X_{2,2}) \wedge \neg(X_{1,2} \wedge X_{3,2}) \wedge \neg(X_{2,2} \wedge X_{3,2})$

$$\begin{aligned}
\alpha \approx\ & (X_{1,1} \vee X_{1,2}), (X_{2,1} \vee X_{2,2}), (X_{3,1} \vee X_{3,2}),\\
& (\neg X_{1,1} \vee \neg X_{2,1}), (\neg X_{1,1} \vee \neg X_{3,1}), (\neg X_{2,1} \vee \neg X_{3,1})\\
& (\neg X_{1,2} \vee \neg X_{2,2}), (\neg X_{1,2} \vee \neg X_{3,2}), (\neg X_{2,2} \vee \neg X_{3,2})
\end{aligned}$$

Modellierungsaufgabe

Allgemeiner: Kann man $n+1$ Tauben auf n Löcher verteilen?

Pigeonhole Formeln $P_n = A_n \wedge B_n$ mit

$$A_n = \bigwedge_{k=1}^{n} \bigwedge_{1 \leq i < j \leq n+1} (\neg X_{i,k} \vee \neg X_{j,k})$$

und

$$B_n = \bigwedge_{i=1}^{n+1} \left(\bigvee_{k=1}^{n} X_{i,k} \right) .$$

Formeln kontrollierbarer Komplexität, von denen man weiß, dass sie unerfüllbar sind.

Formeln sind in KNF.

Modellierungsaufgabe

Fortsetzung Ansatz 2: (Aussagenlogik)

$$\begin{aligned}\alpha \approx\ & (X_{1,1} \vee X_{1,2}), (X_{2,1} \vee X_{2,2}), (X_{3,1} \vee X_{3,2}),\\ & (\neg X_{1,1} \vee \neg X_{2,1}), (\neg X_{1,1} \vee \neg X_{3,1}), (\neg X_{2,1} \vee \neg X_{3,1})\\ & (\neg X_{1,2} \vee \neg X_{2,2}), (\neg X_{1,2} \vee \neg X_{3,2}), (\neg X_{2,2} \vee \neg X_{3,2})\end{aligned}$$

Ist α widerspruchsvoll?

Überprüfung mit Bewertungen erfordert 2^6 Tests (6 Atome).
Wie viele Tests sind allgemein notwendig für P_n?

Alternativer Ansatz für den Beweis auf Widerspruch: RESOLUTION

Aussagenlogische Resolution

Formeln sind in konjunktiver Normalform.

Definition 20 (Resolutionsregel)

Seien α eine Klausel mit Literal L und β eine Klausel mit Literal $\neg L$.
α und β sind die Elternklauseln.
$(\alpha \setminus \{L\}) \cup (\beta \setminus \{\neg L\})$ ist die **Resolvente**.

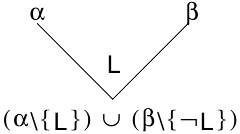
Schreibweisen:

1. $\frac{\alpha \qquad \beta}{(\alpha\setminus\{L\}) \cup (\beta\setminus\{\neg L\})}(RES)$
2. $\alpha, \beta \vdash^{1}_{RES} (\alpha \setminus \{L\}) \cup (\beta \setminus \{\neg L\})$
3. Die leere Klausel wird mit $\sqcup$ bezeichnet. Sie ist immer falsch.
   Beispiel: $\{L, \neg L\} \vdash^{1}_{RES} \sqcup$

Aussagenlogische Resolution

Für einen Resolutionsschritt $\frac{\alpha \vee L, \neg L \vee \beta}{\alpha \vee \beta}(\text{Res})$ schreiben wir auch

$$\alpha \vee L, \neg L \vee \beta \vdash^{1}_{RES} \alpha \vee \beta.$$



Baumdarstellung eines Resolutionsschrittes

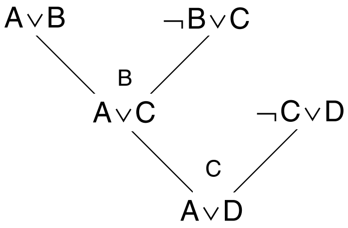
Aussagenlogische Resolution

Beispiel 1: $\alpha_1 = \{A \vee B \vee C \vee D, \neg A \vee B \vee C \vee D\}$

$$
\begin{array}{ccc}
A \vee B \vee C \vee D & & \neg A \vee B \vee C \vee D \\
& A & \\
& B \vee C \vee D &
\end{array}
$$

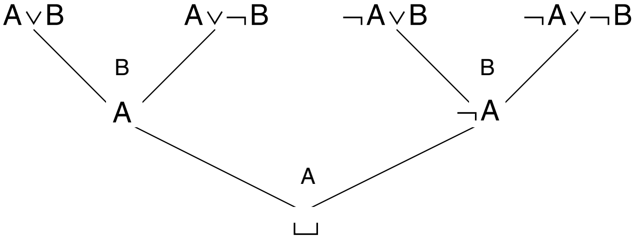
Aussagenlogische Resolution

Beispiel 2: $\alpha_2 = \{A \vee B, \neg B \vee C, \neg C \vee D\}$



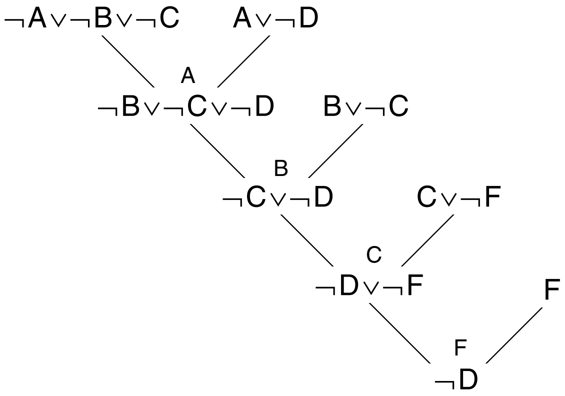
Aussagenlogische Resolution

Beispiel 3: $\alpha_3 = \{A \vee B, A \vee \neg B, \neg A \vee B, \neg A \vee \neg B\}$



Aussagenlogische Resolution

Beispiel 4: $\alpha_4 = \{\neg A \vee \neg B \vee \neg C, A \vee \neg D, B \vee \neg C, C \vee \neg F, F\}$



Aussagenlogische Resolution

Definition 21 (Herleitung)

Sei $\alpha = \{\alpha_1, \ldots, \alpha_n\}$ eine Formel in KNF und π eine Klausel. Eine Folge $\pi_1, \ldots, \pi_k$ ist eine *Herleitung* der Klausel π aus α, wenn $\pi = \pi_k$ gilt und für alle j mit $1 \leq j \leq k$ gilt $\pi_j \in \alpha$ oder es gibt j_1 und j_2 mit $1 \leq j_1, j_2 < j$ und $\pi_{j_1}, \pi_{j_2} \vdash^{1}_{RES} \pi_j$.

Wir sagen π ist *(mit der Resolution) herleitbar* aus α, in Zeichen $\alpha \vdash_{RES} \pi$.

Die Klauseln aus α bezeichnen wir als *Ausgangs*– oder *Startklauseln* oder auch als *Input–Klauseln*.

Die *Länge der Herleitung* $\pi_1, \ldots, \pi_k$ ist k.

Aussagenlogische Resolution

Wir können $\vdash_{RES}$ als den transitiven und reflexiven Abschluss von $\vdash^{1}_{RES}$ auffassen.

Wir verwenden für $\alpha = \{\alpha_1, \ldots, \alpha_n\} \in$ KNF mit Klauseln α_i und α_j und $\alpha_i, \alpha_j \vdash^{1}_{RES} \delta$
die Schreibweise $\alpha_1, \ldots, \alpha_n \vdash^{1}_{RES} \delta$ oder auch $\alpha \vdash^{1}_{RES} \delta$.

Für Resolventenmengen verwenden wir analoge Abkürzungen.

Für die Herleitungen $\alpha \vdash_{RES} L_1$ und $\alpha \vdash_{RES} L_2$ schreiben wir kurz $\alpha \vdash_{RES} L_1, L_2$.

Für $\beta = \{\beta_1, \ldots, \beta_m\}$ mit $\alpha \vdash_{RES} \beta_j$ für alle $1 \leq j \leq m$ schreiben wir $\alpha \vdash_{RES} \beta$.

Aussagenlogische Resolution

Definition 22 (Resolutionsabschluss)

Sei $\alpha \in KNF$ und $n \geq 0$. Dann sei

$$\begin{aligned} Res^0(\alpha) &:= \alpha \\ Res^1(\alpha) &:= \{\pi \mid \exists \tau_1 \tau_2 \in \alpha : \tau_1, \tau_2 \mid\!\frac{1}{RES}\ \pi\} \cup \{\alpha\} \\ Res^{n+1}(\alpha) &:= Res^1(Res^n(\alpha)) \text{ für } n > 0 \\ Res^*(\alpha) &:= \bigcup_n Res^n(\alpha) \end{aligned}$$

Wir bezeichnen $Res^*(\alpha)$ auch als *Resolutionsabschluss* von α.

Aussagenlogische Resolution

Beispiel für Stufensättigungsstrategie:
Wir geben die Resolvententabelle für die Formel α bis zur Stufe 3 an.

$$\alpha = \{\{\neg A, \neg B, \neg C\}, \{A, \neg D\}, \{B, \neg C\}, \{C, \neg F\}, \{F\}\}$$

Nach Stufe 3 können keine neuen Klauseln mehr generiert werden.

0	1		2		3	
1. $\{\neg A, \neg B, \neg C\}$	6. $\{\neg B, \neg C, \neg D\}$	$(1,2)$	11. $\{\neg C, \neg D\}$	$(6,3)$	21. $\{\neg D, \neg F\}$	$(11,4)$
2. $\{A, \neg D\}$	7. $\{\neg A, \neg C\}$	$(1,3)$	12. $\{\neg B, \neg D, \neg F\}$	$(6,4)$	22. $\{\neg D\}$	$(11,10)$
3. $\{B, \neg C\}$	8. $\{\neg A, \neg B, \neg F\}$	$(1,4)$	13. $\{\neg A, \neg F\}$	$(7,4)$		
4. $\{C, \neg F\}$	9. $\{B, \neg F\}$	$(3,4)$	14. $\{\neg A, \neg C, \neg F\}$	$(8,3)$		
5. $\{F\}$	10. $\{C\}$	$(4,5)$	15. $\{\neg A, \neg B\}$	$(8,5)$		
			16. $\{B\}$	$(9,5)$		
			17. $\{\neg C, \neg D, \neg F\}$	$(6,9)$		
			18. $\{\neg B, \neg D\}$	$(6,10)$		
			19. $\{\neg A\}$	$(7,10)$		
			20. $\{\neg A, \neg F\}$	$(8,9)$		

Resolutionslemma

Lemma 23

Sei α eine Formel in KNF (dargestellt als Klauselmenge). Ferner sei π eine Resolvente zweier Klauseln K_1 und K_2 aus α. Dann sind α und $\alpha \cup \{\pi\}$ äquivalent.

Beweis

Mit $K_1 = \beta_1 \vee L$ und $K_2 = \beta_2 \vee \neg L$ folgt die Behauptung direkt aus

$$\underbrace{(\beta_1 \vee L)}_{K_1} \wedge \underbrace{(\beta_2 \vee \neg L)}_{K_2} \approx \underbrace{(\beta_1 \vee L)}_{K_1} \wedge \underbrace{(\beta_2 \vee \neg L)}_{K_2} \wedge \underbrace{(\beta_1 \vee \beta_2)}_{\pi}$$

Diese Äquivalenz weist man leicht durch eine Fallunterscheidung nach: Für $I(L) = w$ gilt für die Formel auf der linken Seite $K_1 \wedge K_2 \approx \beta_2$. Gleiches gilt aber offenbar auch für die Formel auf der rechten Seite, denn $(\beta_1 \vee w) \wedge (\beta_2 \vee f) \wedge (\beta_1 \vee \beta_2) \approx \beta_2 \wedge (\beta_1 \vee \beta_2) \approx \beta_2$. Ein analoges Argument zeigt, dass beide Formeln äquivalent zu β_1 sind, falls $I(L) = f$. q.e.d

Aussagenlogische Resolution

Satz 24

1. *Der Resolutionskalkül ist korrekt.*
   Sei $\alpha \in KNF$*, dann gilt für alle Klauseln* π*:*
   $$\alpha \vdash_{RES} \pi \Longrightarrow \alpha \models \pi.$$
2. *Der Resolutionskalkül ist nicht vollständig.*
   Es gibt Formeln $\alpha \in KNF$ *und Klauseln* π*, so dass gilt:*
   $$\alpha \models \pi \text{ und nicht } \alpha \vdash_{RES} \pi.$$
3. *Der Resolutionskalkül ist widerlegungsvollständig: Sei* $\alpha \in KNF$*, dann gilt:*
   $$\alpha \text{ ist widerspruchsvoll} \Longrightarrow \alpha \vdash_{RES} \sqcup$$

Satz 25

Sei $\alpha \in KNF$*, dann gilt:* α *ist widerspruchsvoll gdw.* $\alpha \vdash_{RES} \sqcup$.

Aussagenlogisches Schlussfolgern

Beispiel: Modus Ponens

$\alpha = A \wedge (A \rightarrow B) \approx A \wedge (\neg A \vee B)$

Offensichtlich leitet man B in einem Resolutionsschritt ab, also

$$\alpha \mid\!\frac{1}{\mathit{RES}}\; B$$

B ist daher eine logische Folgerung aus α.

$\mathit{Res}^1(\alpha) = \mathit{Res}^*(\alpha) = \{A, (\neg A \vee B), B\}$

Um zu beweisen, dass $\alpha \models B$, zeigt man die Unerfüllbarkeit von $\alpha \wedge \neg B$, also

$$\alpha \wedge \neg B \mid\!\frac{}{\mathit{RES}}\; \sqcup \;.$$

Aussagenlogisches Schlussfolgern

Beispiel: Sudoku

- Atomare Formeln $S_{i,j,k}$ mit $i, j, k \in \{1, \ldots, 9\}$.
- $I(S_{i,j,k}) = w$ g.d.w. sich die Zahl k auf dem Feld (i, j) befindet.

5					4			
		1		6				7
	4					1		
				4				6
3			5			8		
		2						
	1			2				2
							9	
9				3				

Aussagenlogisches Schlussfolgern

- Auf jedem Feld befindet sich eine Zahl:

$$\bigwedge_{i,j\in\{1,\ldots,9\}} \bigvee_{k\in\{1,\ldots,9\}} S_{i,j,k}$$

- Auf keinem Feld befinden sich zwei Zahlen:

$$\bigwedge_{i,j\in\{1,\ldots,9\}} \bigvee_{k,l\in\{1,\ldots,9\},k\neq l} (\neg S_{i,j,k} \vee \neg S_{i,j,l})$$

- Forderung, dass in keiner Zeile, Spalte oder Region eine Zahl doppelt vorkommt, wird separat formuliert, z.B. für die erste Zeile:

$$\bigwedge_{j_1,j_2\in\{1,\ldots,9\},\,j_1\neq j_2} \bigvee_{k\in\{1,\ldots,9\}} (\neg S_{1,j_1,k} \vee \neg S_{1,j_2,k})$$

- Schließlich muss noch die Startbedingung des Rätsels beschrieben werden, z.B.

$$S_{1,1,5} \wedge S_{1,6,4} \wedge S_{2,3,1} \wedge \ldots$$

Aussagenlogisches Schlussfolgern

Allgemeines Schema:
Gegebenen: Formeln α und β.
Frage: $\alpha \models \beta$?

1. $\alpha \models \beta$?
2. $\alpha \wedge \neg\beta$ widerspruchsvoll?
3. Transformiere $\alpha \wedge \neg\beta$ in eine äquivalente Formel in KNF, z.B. σ ($\alpha \wedge \neg\beta$ widerspruchsvoll g.d.w. σ widerspruchsvoll)
4. $\sigma \vdash_{RES} \sqcup$?

Satz 26

Sei $\sigma \in KNF$ äquivalent zu $\alpha \wedge \neg\beta$.
$\alpha \models \beta$ genau dann, wenn $\sigma \vdash_{RES} \sqcup$

Aussagenlogisches Schlussfolgern

Zeige die Allgemeingültigkeit von

$$\alpha = (P \rightarrow Q) \rightarrow ((Q \rightarrow R) \rightarrow (P \rightarrow R))$$

Hierzu zeigen wir die Unerfüllbarkeit von $\neg\alpha$.

Die KNF von α ist gegeben durch

$$\{(\neg P \vee Q), (\neg Q \vee R), P, \neg R\}$$

Herleitung der leeren Klausel:

$$\neg P \vee Q, P \vdash_{RES} Q$$
$$\neg Q \vee R, Q \vdash_{RES} R$$
$$\neg R, R \vdash_{RES} \sqcup$$

Modellierung mit Implikationen

Häufig verwendete Form zur Modellbildung:

Wenn **Aussage mit *oder* und *und*, aber ohne Negation**,
dann **Elementaraussage**.

Wenn es regnet oder schneit und Winter ist, dann ist es glatt.

R: Es regnet
S: Es schneit
W: Es ist Winter
G: Es ist glatt

$(R \vee S \wedge W) \rightarrow G$ (richtiges Modell, $\wedge$ bindet stärker als $\vee$)?
$\approx (R \vee (S \wedge W)) \rightarrow G$.

Horn Formeln

Definition 27 (Horn Klauseln und Horn Formeln)

Eine Klausel π ist eine Horn Klausel genau dann, wenn π maximal ein positives Literal enthält. Eine Horn Formel ist eine Konjunktion von Horn Klauseln.

Beispiele:
$((A \wedge B \wedge C) \rightarrow X) \approx (\neg A \vee \neg B \vee \neg C \vee X)$,

(A) und $(\neg B \vee \neg C)$ sind Horn Klauseln

$(A \vee \neg B \vee X)$ ist keine Horn Klausel.

Horn Klausel $(\neg A_1 \vee \ldots \neg A_n \vee B)$ als Implikation $(A_1 \wedge \ldots \wedge A_n) \rightarrow B$, auch geschrieben als $(A_1, \ldots, A_n) \rightarrow B$

negative Klausel: $(\neg A_1 \vee \ldots \vee \neg A_n)$
positive Unit Klausel: A (Faktum)

Horn Formeln

Folgerung für Horn Formeln:

$A, B, (A \to X), ((X \land B) \to Y) \models Y$?

gdw.

$A, B, (A \to X), ((X \land B) \to Y), \neg Y$ widerspruchsvoll?

gdw.

$A, B, (A \to X), ((X \land B) \to Y), \neg Y \vdash_{RES} \sqcup$?

Horn Formeln

Folgerung für Horn Formeln:

$A, B, (A \rightarrow X), ((X \wedge B) \rightarrow Y) \models Y$?

gdw.

$A, B, (A \rightarrow X), ((X \wedge B) \rightarrow Y), \neg Y$ widerspruchsvoll?

gdw.

$A, B, (A \rightarrow X), ((X \wedge B) \rightarrow Y), \neg Y \vdash_{RES} \sqcup$?

Definition 28 (Unit Resolution)

Die Unit-Resolution ist die gewöhnliche Resolution mit der Einschränkung, dass eine der Elternklauseln ein Literal ist. Als Bezeichnung verwenden wir $\vdash^{1}_{\textit{Unit-RES}}$ bzw. $\vdash_{\textit{Unit-RES}}$.

Horn Formeln

Satz 29

Eine Horn Formel α ist widerspruchsvoll genau dann, wenn es eine Unit-Resolutionswiderlegung ($\alpha \vdash_{\textit{Unit-RES}} \sqcup$) gibt.

(ohne Beweis)

1. Ob $\alpha \vdash_{\textit{Unit-Res}} \sqcup$ gilt, kann schnell (in linearer Zeit) entschieden werden.
2. Die Unit-Resolution ist im Allgemeinen nicht widerlegungsvollständig. $\alpha = (A \vee B) \wedge (\neg A \vee B) \wedge (A \vee \neg B) \wedge (\neg A \vee \neg B)$ ist widerspruchsvoll. Es gilt aber nicht $\alpha \vdash_{\textit{Unit-RES}} \sqcup$.

Aussagenlogik

Übersicht:

1. Teil 1: Syntax und Semantik
2. Teil 2: Modellierung und aussagenlogische Beweise
3. Teil 3: **Elementare Beweistechniken I**

Allgemeine Satz- und Beweisform

Beispiel: Satzform

Satz (Pythagoras)
Voraussetzung: Gegeben sei ein beliebiges rechtwinkeliges Dreieck, die Länge der Hypotenuse sei c und die Längen der anderen Seiten seien a und b.
Behauptung: Dann gilt $c^2 = a^2 + b^2$.

Allgemeine Satz- und Beweisform

Satz (Name des Satzes)

Voraussetzung: P
Behauptung: Q

Beweis (kurzer Hinweis auf die Idee, Methode, Struktur des Beweises)

Ausgangspunkt: P ist gegeben, P liegt vor.
Nachweis von Q, eigentlicher Beweis.

q.e.d

Indirekter Beweis

Beispiel:

Zu zeigen: $P \models Q$.

$P \models Q$ gilt gdw. $P \wedge \neg Q$ widerspruchsvoll ist.

Schema für einen indirekten Beweis

Satz (Name des Satzes)

Voraussetzung: P
Behauptung: Q

Beweis (indirekter Beweis, Beweis durch Widerspruch)

Ausgangspunkt: P ist gegeben.

Annahme: Nimm das Gegenteil von Q an, also $\neg Q$.

Führe $\neg Q$ zusammen mit der Voraussetzung P zu einem **Widerspruch**.

q.e.d

Indirekter Beweis

Satz (Satz von Euklid)
Es gibt unendlich viele Primzahlen.

Beweis (durch Widerspruch)
Angenommen, es gäbe nur endliche viele Primzahlen. Wir können diese dann mit $p_1, \ldots, p_n$ bezeichnen und das Produkt

$$m = \prod_{i=1}^{n} p_i = p_1 \cdot p_2 \cdot \ldots \cdot p_n$$

betrachten. Im Hinblick auf die Eigenschaft, prim zu sein, gibt es für den Nachfolger $m' = m + 1$ nur zwei Möglichkeiten:

- m' ist eine Primzahl. Sie ist dann gemäß Konstruktion größer als $p_1, \ldots, p_n$, was im Widerspruch zu unserer Annahme steht.
- m' ist keine Primzahl. Dann muss m' einen Primteiler q besitzen (jede natürliche Zahl lässt sich als Produkt von Primzahlen schreiben), und dieser Teiler muss eine der Zahlen $p_1, \ldots, p_n$ sein. Da q dann offensichtlich gleichzeitig Teiler von m und $m' = m + 1$ ist, folgt $q = 1$. Allerdings ist 1 keine Primzahl, sodass sich wiederum ein Widerspruch ergibt.

q.e.d

Bemerkungen

- Der Beweis benutzt eine weitere wichtige Methode, die der **Fallunterscheidung** (siehe auch Beweis zu Lemma 23).
- Wie in den meisten anderen Beweisen werden gewisse **Definitionen** (wie die der Primzahlen) vorausgesetzt und verwendet, um formale Sachverhalte einfacher darzustellen zu können.
- Viele **Schlussregeln** werden implizit verwendet, wie z.B. der *Modus Ponens*: Aus "ist m keine Primzahl, dann ist m das Produkt von Primzahlen" und "m ist keine Primzahl" folgt, dass m das Produkt von Primzahlen ist.
- Allgemein werden in einem Beweis Definitionen und früher bewiesene Sätze als **Voraussetzung** verwendet, ohne diese explizit zu nennen. Allerdings sollte an den entsprechenden Stellen auf diese Sätze und deren Quellen hingewiesen werden.
- Komplexe Sätze/Beweise werden oft weiter strukturiert, z.B. mithilfe von Lemmata, die dann im eigentlichen Beweis als Voraussetzung verwendet werden.

Satzform Konjunktion

Eine Konjunktion $Q \wedge R$ gilt, falls sowohl Q als auch R gelten. Also müssen beide Aussagen Q und R gezeigt werden. Dies geschieht nacheinander.

Schema für Konjunktion

Satz
Voraussetzung: P
Behauptung: $Q \wedge R$

Beweis
Ausgangspunkt: P ist gegeben.

Teil 1: Zeige Q.
Teil 2: Zeige R.

q.e.d

Satzform Disjunktion

Eine Disjunktion $Q \vee R$ gilt, falls Aussage Q oder Aussage R gilt.
Mindestens eine Aussage, Q oder R muss gelten.

Schema für Disjunktion

Satz
Voraussetzung: P
Behauptung: $Q \vee R$

Beweis
Ausgangspunkt: P ist gegeben.

Alternative 1: Nimm an, Q gilt nicht. Zeige R.
Alternative 2: Nimm an, R gilt nicht. Zeige Q.
Alternative 3: (indirekter Beweis)
Annahme: $\neg(Q \vee R)$ gilt, also $\neg P$ und $\neg Q$ gelten.
Zeige: Ausgangspunkt und Annahme führen zu einem Widerspruch.

q.e.d

Satzform Implikation

Eine Implikation $(Q \to R)$ ist äquivalent zu $\neg Q \vee R$. Der Beweis wird wie in Alternative 1 für Disjunktionen geführt. Wir nehmen Q an und zeigen R.

Beispiel: Ketten von Implikationen

Satz
Voraussetzung: $P, (P \to T), (T \to R)$.
Behauptung: R.

Beweis
Alternative 1:
Schritt 1: Zeige: Aus P und $P \to T$ folgt T.
Schritt 2: Zeige: Aus T und $T \to R$ folgt R.

Alternative 2:
Schritt 1: Zeige: Aus $P \to T$ und $T \to R$ folgt $P \to R$.
Schritt 2: Zeige: Aus P und $P \to R$ folgt R.

q.e.d

Beweisstrukturen

Satzform: Komplexe Implikation $(P_1 \vee \ldots \vee P_n) \rightarrow Q$

Es gilt:

$$\begin{aligned}(P_1 \vee \ldots \vee P_n) \rightarrow Q &\approx \neg(P_1 \vee \ldots \vee P_n) \vee Q \\ &\approx (\neg P_1 \wedge \ldots \wedge \neg P_n) \vee Q \\ &\approx (\neg P_1 \vee Q) \wedge \ldots \wedge (\neg P_n \vee Q) \\ &\approx (P_1 \rightarrow Q) \wedge \ldots \wedge (P_n \rightarrow Q)\end{aligned}$$

Schema für komplexe Implikationen

Satz
Behauptung: $(P_1 \vee \ldots \vee P_n) \rightarrow Q$

Beweis (durch Fallunterscheidung, alle Fälle müssen gelten)

Fall 1: Zeige $P_1 \rightarrow Q$
...
Fall n: Zeige $P_n \rightarrow Q$

q.e.d

Beweisstrukturen

Beispiel: Komplexe Strukturen

Satz:
Voraussetzung: Sei R eine zweistellige Relation über einer Menge M und seien $a, b \in M$ mit $a \neq b$ und $R(a, b)$ und $R(b, a)$.
Behauptung: R ist weder eine Halbordnung, noch eine strenge Halbordnung, noch eine totale Ordnung.

Beweis
Z stehe für die Voraussetzung; HO stehe für "R ist Halbordnung", und sHO für "R ist strenge Halbordnung" und tO für "R ist totale Ordnung".
Die Struktur des Satzes ist: $Z \rightarrow (\neg HO \wedge \neg sHO \wedge \neg tO)$

Nimm Z an und zeige die Konklusion $\neg HO \wedge \neg sHO \wedge \neg tO$

Teil 1: Zeige $\neg HO$
Teil 2: Zeige $\neg sHO$
Teil 3: Zeige $\neg tO$

q.e.d

Strukturelle Induktion

Das Prinzip der induktiven Beweisführung wurde bereits vorgestellt. Hier soll noch ein Beispiel für die strukturelle Induktion über aussagenlogische Formeln gegeben werden.

Allgemeines Gerüst:

- **Induktionsbehauptung:** Für jede wohlgeformte aussagenlogische Formel α gilt die Aussage $P(\alpha)$.
- **Induktionsanfang:** Zeige, dass P für alle aussagenlogischen Atome gilt (inklusive **true** und **false**).
- **Induktionsvoraussetzung:** Sei α eine zusammengesetzte aussagenlogische Formel (also kein Atom, sondern konstruiert gemäß Regel 2 oder 3 in Definition 1). Wir nehmen an, dass $P(\beta)$ für alle echten Teilformeln β von α gilt.
- **Induktionsschluss:** Zeige unter Verwendung der Induktionsvoraussetzung, dass $P(\alpha)$ gilt. Hierzu muss $P(\alpha)$ für jede Konstruktionsvorschrift/Regelanwendung garantiert werden (Fallunterscheidung).

Strukturelle Induktion

Satz

Die Junktormenge $\{\neg, \wedge\}$ ist funktional vollständig, d.h. zu jeder aussagenlogischen Formel α existiert eine äquivalente Formel α', in der ausschließlich die Junktoren $\neg$ und $\wedge$ verwendet werden.

Beweis (durch strukturelle Induktion über den Formelaufbau)

Induktionsanfang: Die Gültigkeit der Aussage für Atome folgt direkt aus der Definition.

Induktionsvoraussetzung: Sei α eine zusammengesetzte aussagenlogische Formel, und gelte die Behauptung für jede echte Teilformel β von α.

Induktionsschluss: Wir zeigen die Eigenschaft für α.

- Fall 1: Wenn $\alpha = \neg\beta$, dann setze $\alpha' = \neg\beta$.
- Fall 2: Wenn $\alpha = (\beta_1 \wedge \beta_2)$, dann setze $\alpha' = (\beta_1 \wedge \beta_2)$.
- Fall 3: Wenn $\alpha = (\beta_1 \vee \beta_2)$, dann setze $\alpha' = \neg(\neg\beta_1 \wedge \neg\beta_2)$.

Da β_1 und β_2 lediglich die Junktoren $\neg$ und $\wedge$ entahlten, gilt dies auch für α'. Ferner gilt offensichtlich $\alpha \approx \alpha'$ in jedem der drei Fälle. **q.e.d.**

Strukturelle Induktion

Satz
Für den Junktor $\downarrow$ sei $I(\alpha \downarrow \beta) = w$ g.d.w. $I(\alpha) = f$ und $I(\beta) = f$. Dann ist der Junktor $\downarrow$ funktional vollständig.

Beweis (durch strukturelle Induktion über den Formelaufbau)
Induktionsanfang: Die Gültigkeit der Aussage für Atome folgt direkt aus der Definition.
Induktionsvoraussetzung: Sei α eine zusammengesetzte aussagenlogische Formel, und gelte die Behauptung für jede echte Teilformel β von α.
Induktionsschluss: Wir zeigen die Eigenschaft für α.

- Fall 1: Wenn $\alpha = \neg\beta$, dann setze $\alpha' = \beta \downarrow \beta$.
- Fall 2: Wenn $\alpha = (\beta_1 \wedge \beta_2)$, dann setze $\alpha' = ((\beta_1 \downarrow \beta_1) \downarrow (\beta_1 \downarrow \beta_1))$.
- Fall 3: Wenn $\alpha = (\beta_1 \vee \beta_2)$, dann setze $\alpha' = ((\beta_1 \downarrow \beta_2) \downarrow (\beta_1 \downarrow \beta_2))$.

Da β_1 und β_2 lediglich den Junktoren $\downarrow$ entahlten, gilt dies auch für α'. Ferner gilt offensichtlich $\alpha \approx \alpha'$ in jedem der drei Fälle. **q.e.d.**