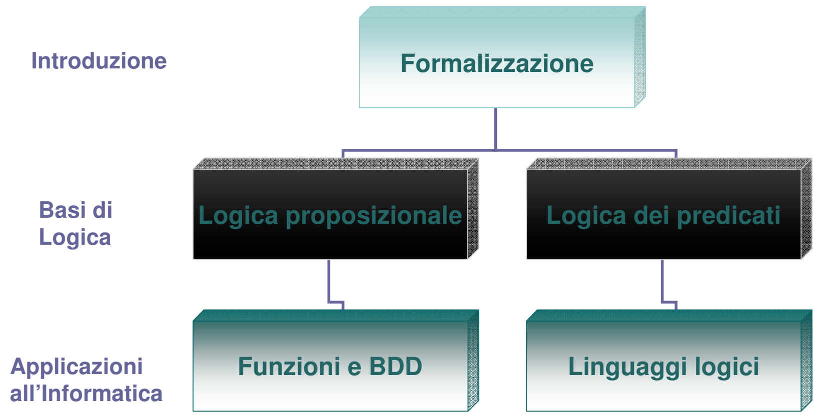
Fondamenti di Logica Matematica
presentazione del corso

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Obiettivi

- Goethe: *"I matematici sono come i francesi: ogni volta che gli si dice qualcosa, la traducono nel loro linguaggio e subito appare diversa."*
- L'obiettivo del corso è di capire il **linguaggio** e le **tecniche** della *logica matematica* per utilizzarli nella risoluzione dei problemi *informatici*.

Obiettivi del Corso



Contenuti

Logica Matematica 5 CFU

Fondamenti di Logica Matematica 6 CFU

- Formalizzazione del linguaggio naturale
- Calcolo Proposizionale
- Calcolo dei Predicati
- Applicazioni della Logica all'Informatica
  - Linguaggi logici
  - Gestione delle funzioni booleane
    - Binary Decision Diagrams (BDD)
- Applicazioni della Logica alla Sicurezza Informatica

Prerequisiti

- Concetti di base della matematica discreta
- Concetti di base delle reti combinatorie

Materiale didattico

- **Libro di testo:**
  Logica a Informatica
  Andrea Asperi, Agata Ciabattoni
  McGraw-Hill, *1997* (biblioteca)
- Lucidi del corso disponibili nel sito: *www.dti.unimi.it/~ciriani*
- Appunti forniti dal docente

Materiale didattico

- Letture consigliate:
  - *Il diavolo in cattedra*
    Piergiorgio Odifredi
    Einaudi, 2003
  - *Algoritmi, divinità e gente comune*,
    Fabrizio Luccio e Linda Pagli
    ETS, 1999

Orario delle lezioni

- Mercoledì 14:00 -16:00
- Giovedì 14:00 -16:00

- Ricevimento:
  - Su appuntamento e a fine lezione
  - ciriani@dti.unimi.it

Modalità di esame

- Per chi frequenta:
  - esercizi in itinere il cui voto sarà un bonus per l'esame scritto
  - esercizi risolvibili anche a coppia
- Esame scritto
- Orale facoltativo

Fondamenti di Logica Matematica
Lezione 1

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- La logica linguistica
- La logica filosofica
  - Paradosso di Achille e la tartaruga
  - Paradosso del mentitore
- La logica matematica

Le vie della logica

- La logica è tradizionalmente studiata in:
  - Linguistica
  - Filosofia
  - Matematica

La prima via

La linguistica

La linguistica

- La via dell'argomentazione
  - logos: parlare da solo, ragionare
  - dialogos: parlare in due, conversare
- Alcune strategie:
  - Riduzione all'assurdo delle posizioni avversarie (Zenone)
  - Argomentazione a favore e contro qualunque posizione (Socrate)
- Ogni parte della struttura del discorso si presta ad abusi retorici.

Esempi di abusi retorici

- Per esempio l'uso improprio di connettivi può portare a conclusioni sbagliate:
- **Congiunzione:**
  - La congiunzione due affermazioni singolarmente consistenti non è necessariamente un'affermazione vera
  - Es: "pena di morte per omicidio" è giustificata da due principi mutuamente contradditori:
    - negazione della violenza
    - affermazione delle giustizia

Esempi di abusi retorici

- **Disgiunzione:**
  - Può portare a falsi dilemmi
  - Ovvero un insieme di alternative non esaustive
  - Es: “siamo uomini o caporali?” dove l’eliminazione di una possibilità sembra implicare necessariamente l’accettazione dell’altra

Esempi di abusi retorici

- **Negazione:**
  - Ciò che non si sa essere vero viene asserito come falso e viceversa.
  - Es: Il massimalismo che considera permesso tutto ciò che non è espressamente proibito.

Esempi di abusi retorici

- **Implicazione:**
  - Un errore tipico è quello di collegare due eventi solo perché si presentano in successione temporale.
  - Oppure quando si dimostrano conclusioni più forti o non implicate dalle premesse.
  - Es: “Se piove allora esco con l’ombrello” non implica che se esco con l’ombrello allora sta piovendo!

Esempi di abusi retorici

- **Quantificatori:**
  - Un errore tipico è quello di affermare contemporaneamente una proposizione universale e il suo controesempio
  - Es: “vero in teoria, ma falso in pratica”
  - Oppure confondere uno o alcuni esempi con una proposizione universale
  - Es: “Se qualcuno è diventato ricco, tutti possono diventarlo”

La seconda via

La filosofia

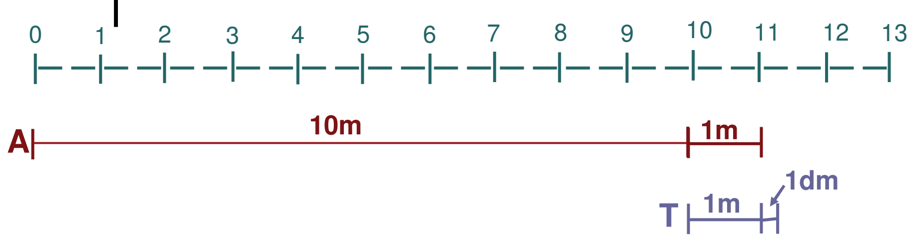
La filosofia

- La via del paradosso
- **Paradosso** ↔ “contro l’opinione corrente”
- Fra i più famosi ci sono:
  - il paradosso di Achille e la tartaruga
  - il paradosso del mentitore

Achille e la tartaruga

- Paradosso di Zenone
- Achille (A) simbolo di rapidità
- Tartaruga (T) simbolo di lentezza
- A corre 10 volte più veloce di T
- **A concede a T 10m di vantaggio**

Achille e la tartaruga



- Quando A corre 10m, T corre 1m
- Quando A corre 1m, T corre 1dm
- .... A non raggiunge mai T.

Achille e la tartaruga

- Caratteristiche di questo paradosso:
  - Infinita divisibilità dello spazio
  - Infinito regresso del ragionamento logico
  - Dicotomia = suddivisione infinitesimale

Ragionamento corretto

- Sia t è il tempo necessario ad Achille per raggiungere la tartaruga;
- Velocità di A 10m/s e di T 1m/s.
- Nell'istante in cui A raggiunge T abbiamo che:
  - In metri: $10t = 10 + 1t$ (10m di vantaggio)
  - $t=10/9$ (tempo finito!!!)

Cosa genera il paradosso?

- Il paradosso è generato dal fatto che anche la **somma di infiniti numeri** può dare per risultato un **numero finito**!!!
- $t = 10/9 = \underbrace{1}_{10m} + \underbrace{0,1}_{1m} + \underbrace{0,01}_{0,1m} + \underbrace{0,001}_{0,01m} + \ldots$

m percorsi da Achille

Il cretese Epimenide

- Epimenide è un **cretese** del VI sec. a.C.
- Disse:
  - *"**tutti** i cretesi dicono **sempre** il falso"*
- Ci chiediamo:
  - *Questa frase può essere vera?*
  - *Questa frase può essere falsa?*
- Se la frase non può essere né vera né falsa è un **paradosso**.

La frase è vera?

- **Tutti** i cretesi dicono **sempre** il falso.
- *Supponiamo che sia vera:*
  - Epimenide ha detto il vero almeno una volta (ha detto una frase vera).
  - La frase "tutti i cretesi dicono sempre il falso" è vera.
- Assurdo: la frase *non può essere vera.*

La frase è falsa?

- **Tutti** i cretesi dicono **sempre** il falso.
- *Supponiamo che sia falsa:*
  - Epimenide sta dicendo il falso.
  - La frase "**tutti** i cretesi dicono **sempre** il falso" è falsa. Ovvero la frase "**qualche** cretese (non necessariamente Epimenide) **qualche** volta **non** dice il falso" è vera.
- La frase è quindi falsa!
- **Non** è un paradosso.

Paradosso di Eubulide

- Eubulide di Mileto (IV sec. a.C.) modifica la frase e arriva ad un vero paradosso.
- Disse: “Io sto mentendo”
- *Supponiamo che sia vera:*
  - Eubulide non sta mentendo.
  - La frase “io sto mentendo” è vera, quindi Eubulide sta mentendo.
- Assurdo: *non è vera.*

Paradosso di Eubulide

- “Io sto mentendo”
- *Supponiamo che sia falsa:*
  - Eubulide sta mentendo.
  - La frase “io sto mentendo” è falsa, quindi Eubulide non sta mentendo.
- Assurdo: *non è falsa.*
- Questo è un **paradosso** perché la frase non è né vera né falsa**!**

Paradosso di Eubulide

- Caratteristica di questo paradosso:
  - Autoriferimento (o riferimento incrociato)
- Come evitare questo tipo di paradosso:
  1. Le frasi contradittorie non hanno senso.
  2. Diversi livelli di linguaggio. Le frasi che parlano di verità e falsità non possono riferirsi a se stesse (sono di un livello superiore).
  3. Terzo valore di verità: indefinito.

La terza via

La matematica!

La matematica

- La via della dimostrazione
- Antichi Greci già distinguevano tra
  - *enunciati*: affermazioni pure e semplici
  - *teoremi*: affermazioni dotate di una dimostrazione
- La necessità pratica di distinguere tra enunciati corretti ed errati ha portato alla formalizzazione delle dimostrazioni.

Letture

- Un approfondimento di questa introduzione storico-filosofica è nei primi capitoli del libro:
  - *Il diavolo in cattedra*
    di Piergiorgio Odifredi
    Einaudi, 2003

Fondamenti di Logica Matematica

Lezione 2

Valentina Ciriani

DTI - Università di Milano

www.dti.unimi.it/~ciriani

Contenuti della lezione

- Richiami di nozioni di base
  - Insiemi
  - Operazioni tra insiemi
  - Funzioni

Gli insiemi

- Un **insieme** è una collezione di oggetti detti elementi dell'insieme.
- Principio di *estensionalità*:

  Gli insiemi sono completamente caratterizzati dai loro elementi: due insiemi sono uguali se contengono gli stessi elementi.

L'universo

- Gli insiemi che consideriamo conterranno solo elementi di tale universo U.
- Esempi di universo:
  - Naturali $\{0, 1, 2, ...\}$
  - Interi $\{... -2, -1, 0, 1, 2, ...\}$
  - Reali
  - Booleani $B = \{0, 1\}$
  - $B^n = \{0, 1\}^n$ = {numeri di n cifre booleane}.

    Es. $n = 2$, $\{0, 1\}^2 = \{00, 01, 10, 11\}$

Rappresentazione di insiemi

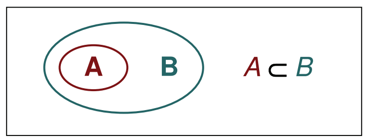
- Consideriamo l'universo dei naturali. L'insieme che contiene gli elementi *1, 7,* e *4* può essere denotato con { *1, 7, 4* }.
- L'ordine non conta.
- Non contano le ripetizioni.
- Es: { *1, 7, 4* } = { *1, 4, 7* } = { *1, 7, 4, 7* }.

Appartenenza

- I simbolo di appartenenza: $\in$
- Sia a è un elemento e A un insieme, allora $a \in A$ significa che a **appartiene** ad A.
- Scriveremo $a \notin A$ per dire che a non appartiene ad A.
- **L'insieme vuoto** $\emptyset$ = {} è un insieme che non contiene alcun elemento.

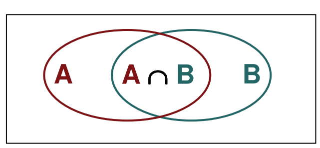
Inclusione

- Un insieme A è un **sottoinsieme** di B ($A \subseteq B$) se ogni elemento di A appartiene anche a B.
- Un insieme A è un **sottoinsieme proprio** di B ($A \subset B$) se $A \subseteq B$ ma A è diverso da B ($A \neq B$).
- Se $A \subseteq B$ e B $\subseteq A$ allora $A = B$



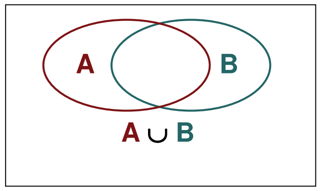
Intersezione

- L'insieme degli elementi che appartengono sia a A che a B si chiama **intersezione** di A e B ($A \cap B$).
- Due insiemi tali che $A \cap B = \varnothing$ si dicono **disgiunti**.



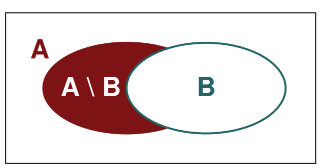
Unione

- L'**unione** di A e B (*A* ∪ *B*) è l'insieme degli elementi che appartengono ad almeno uno dei due insiemi A e B.



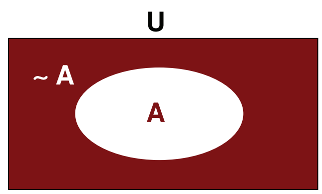
Differenza

- la **differenza** tra A e B (*A* \ *B*) è l'insieme degli elementi che appartengono ad A ma non a B.



Il complemento

- Il **complemento** di un insieme A (~A) è la differenza tra l'universo e l'insieme A
- Non esiste una notazione standard per il complemento di un insieme. Le notazioni più usate sono ~A, ¬A, A', A^c.



Leggi di associatività

- Leggi di associatività:
  - $(A \cup B) \cup C = A \cup (B \cup C)$
  - $(A \cap B) \cap C = A \cap (B \cap C)$
- Grazie a queste identità possiamo omettere alcune parentesi.
- Ad esempio possiamo scrivere:
  - $A \cup B \cup C$

Altre proprietà

- Idempotenza
  - $(A \cup A) = A$
  - $(A \cap A) = A$
- Commutatività
  - $(A \cup B) = (B \cup A)$
  - $(A \cap B) = (B \cap A)$
- Distributività
  - $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
  - $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Le coppie

- <a, b> è una **coppia ordinata** il cui primo elemento è a e il cui secondo elemento è b.
- Osservazioni:
  - <a, b> = <c, d> solo se a = c e b = d.
  - Se $a \neq b$ allora $\langle a, b\rangle \neq \langle b, a\rangle$
- Un insieme di due elementi {a, b} è anche chiamato **coppia non ordinata**.

Tuple

- La nozione di tupla generalizza quella di coppia.
- Una **n-tupla** (ordinata) di un insieme A è una sequenza $<a_1, \dots , a_n>$ di elementi di A.
- Es: una 4-tupla di numeri naturali:
  $<10, 3, 10, 5>$

Prodotti cartesiani

- Il **prodotto cartesiano** degli insiemi A e B è l'insieme di coppie il cui primo elemento appartiene ad A ed il secondo appartiene a B:
  - $A \times B = \{<a,b> \mid a \in A \text{ e } b \in B\}$
- Es $\{0,1\} \times \{1,2\} = \{<0,1>,<0,2>,<1,1>,<1,2>\}$
- In generale, il prodotto cartesiano degli insiemi $A_1,\dots , A_n$ è
  - $A_1 \times \dots \times A_n = \{<a_1, \dots , a_n> \mid a_1 \in A_1,\dots, a_n \in A_n\}$

Potenze cartesiane

- Il **quadrato cartesiano** dell'insieme A è:
  - $A^2 = A \times A = \{<a,b> \mid a \in A \text{ e } b \in A\}$
- La **potenza cartesiana** n-esima dell'insieme A è l'insieme delle n-tuple di elementi di A :
  - $A^n = A \times \ldots \times A =$

    $=\{<a_1, \ldots, a_n> \mid a_1 \in A, \ldots, a_n \in A\}$

17 Logica Matematica Valentina Ciriani

Funzioni parziali

- Siano A e B due insiemi, una **funzione parziale** $F: A \to B$ è un insieme di coppie <a,b> (con $a \in A$ e $b \in B$) in cui ogni elemento di A è in coppia con al più un elemento di B.
- Es. $A = \{0,1,2,4\}$, $B= \{0,3,6\}$
- $F: A \to B = \{<0,0>, <1,0>, <4,6>\}$
- ovvero F(0)=0, F(1)=0, F(2)=?, F(4)=6

18 Logica Matematica Valentina Ciriani

Funzioni totali

- Siano A e B due insiemi, una **funzione totale** F: A $\rightarrow$ B è una funzione parziale che associa **ad ogni** elemento di A un elemento di B.
- Es. A = {a,b,c,d} , B= {0,3,6}
- F = {<a,0>, <b,0>, <c,6>, <d,6>}
- ovvero F(a)=0, F(b)=0, F(c)=6, F(d)=6

Dominio e immagine

- L'insieme degli x dove F è definita si chiama **dominio di definizione** di F (Dom (F)).
- L'insieme degli y tali che <x, y> $\in$ F, per ogni x $\in$ Dom(F), si chiama **immagine** di F (Im(F)).
- Esempio F : {a,b,c,d} $\rightarrow$ {0,2,6}

  F = {<a,0>, <c,6>, <d,6>}

  Dom(F) = {a,c,d} e Im(F) = {0,6}.
- *Osservazione:* Se il dominio di F : A $\rightarrow$ B è A allora la funzione è totale.

Funzioni suriettive

- Sia F: A → B , se Im(F) = B allora la funzione di dice **suriettiva**.
- *Esempio* F : {a,b,c,d} → {0,2,6}
  F = {<a,0>, <c,6>, <d,6>}
  Im(F) = {0,6} ≠ {0,2,6}: **non** è suriettiva.
- *Esempio* G : {a,b,c,d} → {0,2,6}
  G = {<a,0>, <c,2>, <d,6>}
  Im(G) = {0,2,6}, è suriettiva!

Funzioni iniettive

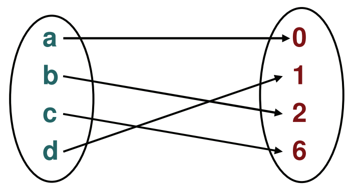
- Una funzione F si dice **iniettiva** se per ogni y ∈ Im(F) esiste al più un x tale che <x,y> ∈ F.
- *Esempio* F : {a,b,c,d} → {0,2,6}
  F = {<a,0>, <c,6>, <d,6>}
  Im(F) = {0,6}, sia <c,6> che <d,6> sono in F
  F **non** è iniettiva.
- *Esempio* G : {a,b,c,d} → {0,2,6}
  G = {<a,0>, <d,6>}
  Im(G) = {0,6}, è iniettiva!

Funzioni biiettive

- Una funzione totale, suriettiva ed iniettiva è detta **biiettiva**.
- *Esempio* $F : \{a,b,c,d\} \rightarrow \{0,1,2,6\}$

  $F = \{<a,0>, <b,2>, <c,6>, <d,1>\}$

  F è biiettiva.



Inversa

- Sia $F: A \rightarrow B$ una funzione biiettiva, la **funzione inversa** di F (F^{-1}) è la funzione $F^{-1}: B \rightarrow A$ tale che

  $<a,b> \in F$ *se solo se* $<b,a> \in F^{-1}$
- Esempio: $F : \{a,b,c,d\} \rightarrow \{0,1,2,6\}$

  $F = \{<a,0>, <b,2>, <c,6>, <d,1>\}$

  $F^{-1} = \{<0,a>, <1,d>, <2,b>, <6,c>\}$

Fondamenti di Logica Matematica

Esercitazione 2

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Insiemi

- Quali di questi insiemi sono uguali?:
  - {*1, 7, 4* },
  - {*1, 7, 1, 4* },
  - {*1, 4, 7*},
  - {*1, 3, 4, 5*}
  - {*1, 3, 5*}
  - {*5, 3, 4, 1*}

Inclusione

- Quali di questi insiemi sono contenuti in almeno un altro?:
  - {*1, 7, 4* },
  - {*1, 7, 1, 4* },
  - {*1, 4, 7*},
  - {*1, 3, 4, 5*}
  - {*1, 3, 5*}
  - {*5, 3, 4, 1, 7*}

Intersezione e unione

- Scrivere l'intersezione e l'unione dei seguenti insiemi:
  - {*1, 7, 4* } e {*3, 7, 1* }
  - {*1, 8, 9, 4, 5* } e {*w, a, c* }
- Scrivere la differenza tra i precedenti insiemi.

Complemento

- Sia l'universo U= {*1, 2, 3, 4, 5, 6*} scrivere il complemento dei seguenti insiemi:
  - { *1, 5, 4* }
  - {*2, 3, 5, 6*}

Dimostrazioni

- Dimostrare la legge distributiva:
  - $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
- 1. dimostrare che
  - $A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$
- 2. dimostrare che
  - $(A \cup B) \cap (A \cup C) \subseteq A \cup (B \cap C)$
- 1. $a \in A \cup (B \cap C) \rightarrow a \in (A \cup B) \cap (A \cup C)$
- 2. $a \in (A \cup B) \cap (A \cup C) \rightarrow a \in A \cup (B \cap C)$

$a \in A \cup (B \cap C)$

- $a \in A \cup (B \cap C) \rightarrow a \in (A \cup B) \cap (A \cup C)$
- 1. $a \in A$
  - $\rightarrow a \in (A \cup B)$
  - $\rightarrow a \in (A \cup C)$
- 2. $a \in (B \cap C)$
  - $\rightarrow a \in B \rightarrow a \in (A \cup B)$
  - $\rightarrow a \in C \rightarrow a \in (A \cup C)$

$a \in (A \cup B) \cap (A \cup C)$

- $a \in (A \cup B) \cap (A \cup C) \rightarrow a \in A \cup (B \cap C)$
- $a \in (A \cup B)$ e $a \in (A \cup C)$
- $a \in (A \cup B) \rightarrow (a \in A$ **oppure** $a \in B)$
- $a \in (A \cup C) \rightarrow (a \in A$ **oppure** $a \in C)$
- Due casi:
  - $a \in A$ oppure
  - $a \in B$ e $a \in C$

Prodotti cartesiani

- Calcolare i seguenti prodotti cartesiani:
  - $\{4, 5, 1\} \times \{1,2\}$
  - $\{e, a\} \times \{6,9\}$

Funzioni

- Sia $A = \{0,1,2,3\}$, $B= \{0,1,3,6\}$
- I seguenti insiemi sono funzioni?
  - $F = \{<0,0>, <0,1>, <3,6>\}$
  - $F = \{<0,0>, <1,0>, <3,6>\}$
  - $F = \{<0,0>, <1,3>, <2,1>, <3,6>\}$

Funzioni

- Sia A = {0,1,2,3} , B= {0,1,3,6}
- Le seguenti funzioni sono parziali, totali, iniettive, suriettive, biiettive?
  - F = {<0,0>, <1,0>, <3,6>}
  - F = {<0,0>, <1,3>, <2,1>, <3,6>}
  - F = {<0,0>, <1,3>, <2,0>, <3,6>}
- Calcolare l’inversa delle funzioni precedenti (quando esiste).

Fondamenti di Logica Matematica
Lezione 3

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Enunciati
- Logica proposizionale
  - Proposizioni
  - Alfabeto
  - Sintassi
- Induzione strutturale

Denotazione

- Un **nome** è un'espressione linguistica che denota una qualche entità.
- Esempio:
  - Quattro
  - Il quadrato di due
  - Il predecessore di cinque
  - Tre più uno
- Sono modi diversi per denotare lo stesso oggetto: il numero quattro.

Senso

- Il **senso** di un nome è quanto effettivamente esprime ovvero la sua connotazione.
- Esempio:
  - Quattro
  - Il quadrato di due
  - Il predecessore di cinque
  - Tre più uno
- Questi nomi hanno tutti un senso diverso anche se si riferiscono allo stesso oggetto.

Sostituzioni

- La denotazione è **invariante per sostituzione**.
- Esempio:
  - Il quadrato di quattro
  - Il quadrato di (tre più uno)
- Denotano sempre lo stesso numero, ovvero sedici.

Sostituzioni

- Il senso **non è** invariante per sostituzione.
- Esempio:
  - Quattro è il quadrato di due
  - Quattro è quattro
- Le due frasi non hanno lo stesso senso.

Tipi di enunciati

- Tre tipi di enunciati (o sentenze):
  - **Interrogativi**: Piove? X=5?
  - **Imperativi**: Fate i compiti! X:=5
  - **Dichiarativi**: Oggi piove. X=5
- Gli enunciati dichiarativi sono detti anche **asserzioni**
- Noi parleremo solo di enunciati dichiarativi e li chiameremo semplicemente enunciati.

Principi

- Principio di **bivalenza**:
  - Ogni enunciato è sempre vero o falso.
- Principio di **estensionalità**:
  - Il valore di verità degli enunciati composti dipende solamente dal valore di verità degli enunciati che li compongono.

Invarianza per sostituzione

Enunciati atomici

- Un enunciato è **semplice** (o **atomico)** se non contiene nessun altro enunciato.
- Esempi di enunciati atomici:
  - Paolo corre
  - Laura ha i capelli rossi
  - X = 3

Enunciati composti

- Un enunciato è **composto** se contiene altri enunciati (è possibile scomporlo in enunciati più semplici).
- Esempi di enunciati composti:
  - Piove e c'è vento
  - Se c'è il sole allora esco
  - Carla ha gli occhi neri o Carla ha i capelli neri
  - Non piove

La logica proposizionale

- Noi ci occupiamo di particolari enunciati che formano la logica proposizionale.
- Questi enunciati sono denominati **proposizioni**

Proposizioni

- Una **proposizione** è un particolare enunciato e può essere:
  - *atomica*
  - *composta*: cioè costruita a partire da proposizioni atomiche usando i connettivi: non, e, o, se... allora.

$$\sim , \wedge , \vee , \Rightarrow$$

Connettivi

- I connettivi della logica proposizionale **sono estensionali**.
- Esempio:
  - Sara è bruna **o** Mario è biondo
  - x = 5 **e** y=7
  - **Se** oggi piove **allora** esco con l'ombrello

Connettivi non estensionali

- Esistono connettivi non estensionali (ma non sono nella logica proposizionale).
- Esempio:
  - "poiché"
  - 2 è un numero primo <u>poiché</u> è divisibile solo per 1 e per se stesso
  - 2 è un numero primo <u>poiché</u> 2 è pari.

Scelta dei connettivi

- Perché considerare solo alcuni connettivi estensionali (non, e, o se... allora)?
  - Economicità
  - Chiarezza
  - Scelta quasi del tutto arbitraria

Logica proposizionale

- Introduzione al linguaggio proposizionale: **sintassi**
- Intrerpretazione del linguaggio proposizionale, ovvero suo significato: **semantica**
- Insieme di regole che ci permetta di stabilire la verità o la falsità di una frase del linguaggio: **calcolo**

La sintassi

- Guarderemo le frasi come semplici stringhe senza dar loro alcun significato.
- Quali sono le frasi del linguaggio logico?
  - Quali frasi sono permesse?
  - Quali frasi non vanno bene?

Notazione

- Proposizioni atomiche:
  - A, B, C,.....
- Proposizioni composte:
  - P, Q
- Connettivi logici:
  - $\sim$ = non
  - $\wedge$ = e
  - $\vee$ = o
  - $\Rightarrow$ = se…allora

Alfabeto

- L'alfabeto del linguaggio proposizionale contiene:
  - Simboli atomici (o lettere enunciative): A, B,...
  - Connettivi logici: $\sim$, $\wedge$, $\vee$, $\Rightarrow$, F, T
  - Simboli ausiliari: (,).
- Come si possono combinare questi simboli?
  -)A $\sim$ (B è una formula del linguaggio?
  - ((A $\vee$ B) $\Rightarrow$ C) è una formula del linguaggio?

Formule ben formate

- Sono le formule corrette del linguaggio.
- Le **f.b.f.** sono definite in modo ricorsivo:
  - Le lettere enunciative e i simboli F e T sono f.b.f.,
  - se P è una f.b.f. anche ($\sim$P) è una f.b.f.,
  - se P e Q sono f.b.f. anche (P $\wedge$ Q), (P $\vee$ Q), (P$\Rightarrow$Q), sono f.b.f,
  - niente altro è una f.b.f.

Esempi di f.b.f.

-)A ~ (B è una formula del linguaggio?
  - No! Perché?
- $((A \vee B) \Rightarrow C)$ è una formula del linguaggio? Sì!
  - $((A \vee B) \Rightarrow C)$ è f.b.f. sse $(A \vee B)$ è f.b.f. e C è f.b.f..
  - $(A \vee B)$ è f.b.f. sse A è f.b.f. e B è f.b.f.
  - A, B, C sono f.b.f. per definizione

Quante parentesi!

- $((A \vee B) \Rightarrow C)$ è una f.b.f. ma contiene troppe parentesi!
- Per evitare l'uso di molte parentesi si fissa una **precedenza** nell'uso dei connettivi:
  - $\sim$ precede $\wedge$ che precede $\vee$ che precede $\Rightarrow$
  - connettivi uguali si intendono associati a sinistra
- $((A \vee (\sim B)) \Rightarrow C)$ è equivalente a $A \vee \sim B \Rightarrow C$
- $(A \Rightarrow B) \vee C$ **non** è equivalente a $A \Rightarrow B \vee C$
  - $A \Rightarrow B \vee C$ è equivalente a $A \Rightarrow (B \vee C)$
  - Dobbiamo lasciare le parentesi!

Sottoformule

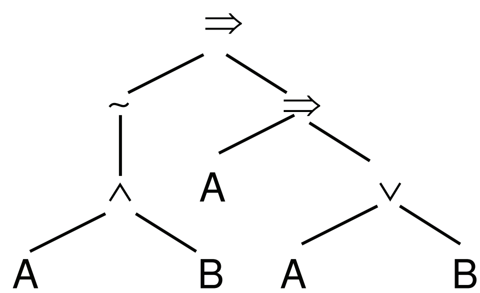
- Sia P una f.b.f. le **sottoformule di** P, **Stfm(P),** sono così definite:
- se P è una lettera enunciativa o F o T
  - $Stfm(P) = \{P\}$.
- se P è $\sim Q$,
  - $Stfm(P)=\{P\} \cup Stfm(Q)$,
- se P è $P_1 \wedge P_2$, $P_1 \vee P_2$ o $P_1 \Rightarrow P_2$,
  - $Stfm(P)=\{P\} \cup Stfm(P_1) \cup Stfm(P_2)$.

Esempio di sottoformule

- $\sim(A \wedge B) \Rightarrow (A \Rightarrow B \vee A)$.
- Sottoformule:
  - $\sim(A \wedge B) \Rightarrow (A \Rightarrow B \vee A)$
  - $\sim(A \wedge B)$
  - $(A \Rightarrow B \vee A)$
  - $A \wedge B$
  - A
  - $B \vee A$
  - B

Rappresentazione ad albero

- $\sim(A \wedge B) \Rightarrow (A \Rightarrow B \vee A)$
- Ha come **albero di struttura**:

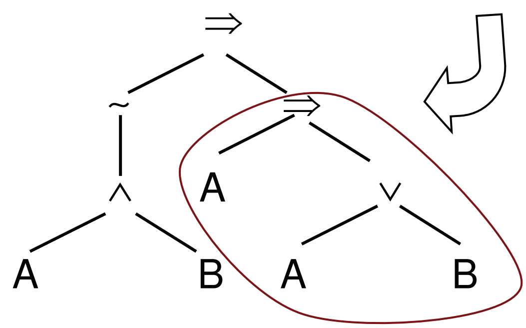


Alberi di struttura

- Un albero di struttura ha:
  - come **radice** l'ultimo connettivo usato (connettivo principale),
  - come **foglie** le lettere enunciative,
  - come **nodi interni** connettivi.
- Ogni nodo è la radice di un sottoalbero di struttura di una sottoformula
- Ogni sottoformula ha come albero di struttura il sottoalbero che ha come radice il nodo etichettato dal suo connettivo principale.

Sottoalberi e sottoformule

- $\sim(A \wedge B) \Rightarrow (A \Rightarrow B \vee A)$
- La sottoformula $A \Rightarrow B \vee A$ corrisponde:



Induzione strutturale

- Le f.b.f sono definite in modo ricorsivo.
- Per dimostrare le proprietà delle f.b.f useremo il metodo di dimostrazione per **induzione strutturale**.
- Ovvero se si dimostra che una proprietà è vera per ogni possibile “struttura” di una f.b.f
  - allora è vera per tutte le f.b.f.

Induzione strutturale

- **Teorema:** Sia $\mathbb{P}$ una proprietà, **se**
  - $\mathbb{P}$ vale per tutte le lettere enunciative e per F e T,
  - e se $\mathbb{P}$ vale per P e Q, allora si può dimostrare che $\mathbb{P}$ vale anche per ~ P, P$\wedge$Q, P$\vee$Q, P$\Rightarrow$Q

allora $\mathbb{P}$ vale per tutte le f.b.f.

Induzione strutturale

- Non dobbiamo dimostrare la validità di $\mathbb{P}$ per tutte le possibili formule!
- Basta dimostrare che:
  - Caso base:
    $\mathbb{P}$ vale per tutte le lettere enunciative e per F e T
  - I caso induttivo:
    se $\boxed{\mathbb{P} \text{ vale per P}}$ allora $\mathbb{P}$ vale per ~P
  - II caso induttivo:
    se $\boxed{\mathbb{P} \text{ vale per P e Q}}$ allora $\mathbb{P}$ vale per P$\wedge$Q
  - III caso induttivo:
    se $\boxed{\mathbb{P} \text{ vale per P e Q}}$ allora $\mathbb{P}$ vale per P$\vee$Q
  - IV caso induttivo:
    se $\boxed{\mathbb{P} \text{ vale per P e Q}}$ allora $\mathbb{P}$ vale per P$\Rightarrow$Q

Ipotesi induttiva

Esempio di ind. strutturale

- **Provare che ogni formula ben formata contiene lo stesso numero di parentesi aperte e chiuse**

- Dimostrazione per induzione strutturale
- A(P)= numero di parentesi aperte di P
- C(P)= numero di parentesi chiuse di P
- Dimostriamo che per ogni fbf P è vero che
  - A(P) = C(P) ← P

Esempio

- **Caso base**:
  - *Simboli atomici, F e T* non hanno parentesi:
    - A(atomico) = 0 = C(atomico)
    - A(T) = 0 = C(T) e A(F) = 0 = C(F)
- **I Caso induttivo (negazione)** ~(P):
  - Ipotesi induttiva: A(P) = C(P)
  - Tesi: A(~(P)) = C(~(P))

  - A(~(P)) = 1+ A(P) = 1 + C(P) = C(~(P))

Esempio

- **Il Caso induttivo (cong.)** $(P \wedge Q)$:
  - Ipotesi induttiva: $A(P) = C(P)$ e $A(Q) = C(Q)$,
  - Tesi: $A((P \wedge Q)) = C((P \wedge Q))$

  - $A((P \wedge Q)) = 1 + A(P) + A(Q) =$
    $= 1 + C(P) + C(Q) = C((P \wedge Q))$

Fondamenti di Logica Matematica
Lezione 4

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Semantica della logica proposizionale
  - Connettivi logici
  - Interpretazione
  - Soddisfacibilità e modelli
  - Tautologia
  - Conseguenza semantica

Richiami di sintassi

- L'**alfabeto**:
  - Simboli atomici (o lettere enunciative): A, B,...
  - Connettivi logici: $\sim$, $\wedge$, $\vee$, $\Rightarrow$, F, T
  - Simboli ausiliari: (,).
- Le **formule ben formate** :
  - Le lettere enunciative e i simboli F e T sono f.b.f.,
  - se P è una f.b.f. anche ($\sim$P) è una f.b.f.,
  - se P e Q sono f.b.f. anche (P $\wedge$ Q), (P $\vee$ Q), (P$\Rightarrow$Q), sono f.b.f,
  - niente altro è una f.b.f.

La semantica

- **Obiettivo:**attribuire un valore di verità ad una formula ben formata.
- Partiamo dai valori di verità delle lettere enunciative che compongono la formula.
- Otteniamo il valore di verità dell'intera formula.

La funzione semantica

- Valori di verità:
  - 1 è la denotazione del valore “vero”
  - 0 è la denotazione del valore “falso”
- Funzione semantica v:
  - v: f.b.f $\rightarrow$ {0,1}
- Le funzioni semantiche sono anche dette **interpretazioni**.

Proposizioni atomiche

La semantica delle proposizioni atomiche:

- v(T) = 1
- v(F) = 0
- Sia A una lettera enunciativa
  - v(A) = 1 se A è vera
  - v(A) = 0 se A è falsa

La negazione

- ~P è vera quando P è falsa e viceversa
- Ovvero v(~P) = 1 - v(P)
- Rappresentazione con la tavola di verità:

P	**~P**
0	1
1	0

Congiunzione

- $P \wedge Q$ è vera sse P è vera e Q è vera.
- $v(P \wedge Q) = 1 \Leftrightarrow v(P) =1$ e $v(Q) =1$
- $v(P \wedge Q) = \min(v(P), v(Q))$
- Rappresentazione con la tavola di verità:

P	**Q**	**$P \wedge Q$**
0	0	0
0	1	0
1	0	0
1	1	1

Disgiunzione

- $P \vee Q$ è vera se almeno una tra P e Q è vera.
- $v(P \vee Q) = 1 \Leftrightarrow v(P) = 1$ o $v(Q) = 1$
- $v(P \vee Q) = \max(v(P), v(Q))$
- Rappresentazione con la tavola di verità:

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

Implicazione $P \Rightarrow Q$

- Se $v(P)= 1$ e $v(Q)= 1$ allora
  - $v(P \Rightarrow Q) = 1$
- Se $v(P)= 0$ e $v(Q) = 0$ allora
  - $v(P \Rightarrow Q) = 1$
- Se $v(P) = 1$ e $v(Q) = 0$ allora
  - $v(P \Rightarrow Q) = 0$

Implicazione

- Se $v(P) = 0$ e $v(Q) = 1$ allora
  - $v(P \Rightarrow Q) = 1$?
  - $v(P \Rightarrow Q) = 0$?
- $v(P \Rightarrow Q) = 1$!!!!
- Una premessa falsa implica qualunque proposizione.
- Se fosse $v(P \Rightarrow Q) = 0$ allora si avrebbe la definizione di $\Leftrightarrow$.

Implicazione

- Osservazione:
  - $v(P \Rightarrow Q) = 0$ sse $v(P) = 1$ e $v(Q) = 0$
  - $v(P \Rightarrow Q) = \mathbf{1}$ sse $v(P) = \mathbf{0}$ **<u>o</u>** $v(Q) = \mathbf{1}$
- $v(P \Rightarrow Q) = 1 \Leftrightarrow v(\mathbf{\sim P} \vee \mathbf{Q}) = 1$
- $v(P \Rightarrow Q) = 1 \Leftrightarrow v(P) \leq v(Q)$
- $v(P \Rightarrow Q) = v(\sim P \vee Q) = \max(v(\sim P), v(Q)) = \max(1 - v(P), v(Q))$

Implicazione

- Rappresentazione con la tavola di verità:

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

Interpretazione

- Fissare un'interpretazione v: f.b.f $\rightarrow$ {0,1} corrisponde a:
  - attribuire i valori 0 a F e 1 a T.
  - attribuire un valore di verità (0 o 1) a tutte le lettere enunciative.
  - attribuire il valore di verità a $\sim$A, A$\wedge$B, A$\vee$B, A$\Rightarrow$B seguendo le regole delle loro tavole di verità.

Def. di interpretazione

- Una funzione v: f.b.f $\rightarrow$ {0,1} è un'interpretazione se:
  - $v(\sim P) = 1 - v(P)$
  - $v(T) = 1$ e $v(F) = 0$
  - $v(P \wedge Q) = \min(v(P), v(Q))$
  - $v(P \vee Q) = \max(v(P), v(Q))$
  - $v(P \Rightarrow Q) = \max(1\text{-}v(P), v(Q))$

Esempio di interpretazione

- {A,B,C} è l'insieme di formule atomiche e v un'interpretazione così definita :
  - $v(A) = 0$
  - $v(B) = 1$
  - $v(C) = 0$
- $v(A \wedge B) = \min(v(A), v(B)) = \min(0, 1) = 0$
- $v(\sim A \wedge (B \vee C)) = \min(1\text{-}v(A), \max(v(B), v(C))) = \min(1 - 0, \max(1,0))= 1$
- Ovvero se A e C sono falsi e B è vero allora $\sim A \wedge (B \vee C)$ è vero!

Proprietà

- Sia P una fbf e v, v' due interpretazioni.
  - Se v(A)=v'(A) per ogni proposizione atomica che compare in P
  - allora v(P)=v'(P)
- Questa proprietà ci dice che:
  - l'interpretazione di una proposizione dipende **esclusivamente** dall'interpretazione delle proposizioni atomiche che la compongono

Dimostrazione

- Per induzione strutturale:
  - Dimostriamo che la proprietà è vera per ogni connettivo ($\sim$, $\wedge$, $\vee$, $\Rightarrow$) e per le proposizioni atomiche.
  - L'induzione strutturale ci garantisce che la proprietà è vera per **ogni proposizione.**

Dimostrazione

- **Caso base:** proposizioni atomiche, T e F
  - 1) Caso P = A proposizione atomica
  - *Dobbiamo dimostrare che* v(P)=v'(P)
  - Per *ipotesi* sappiamo che per ogni proposizione atomica A è vero che v(A)=v'(A) quindi v(P)=v'(P)
  - 2) Caso P = T o P = F. Per definizione di interpretazione:
    - v(T) = 1 = v'(T) e v(F) = 0 = v'(F).

Dimostrazione

- **Caso induttivo 1:** negazione
  - P = ~Q
  - *Dobbiamo dimostrare che* v(P)=v'(P)
  - Per *ipotesi induttiva* sappiamo che v(Q)=v'(Q)
  - quindi v(P)= 1 - v(Q) = 1 - v'(Q) =v'(P)

Dimostrazione

- **Caso induttivo 2:** congiunzione
  - P = Q $\wedge$ R
  - *Dobbiamo dimostrare che* v(P)=v'(P)
  - Per *ipotesi induttiva* sappiamo che v(Q)=v'(Q) e che v(R)=v'(R)
  - quindi v(P) = v(Q $\wedge$ R)= min(v(Q), v(R)) = = min(v'(Q), v'(R))= v'(Q $\wedge$ R) = v'(P) .

Esempio di interpretazioni

- Per la proprietà **tutte le possibili interpretazioni** della proposizione **P = ~(A$\wedge$B) $\wedge$ (A$\Rightarrow$(B$\vee$A))** sono:

int	A	B	(A$\wedge$B)	~(A$\wedge$B)	(B$\vee$A)	(A$\Rightarrow$(B$\vee$A))	P
v_1	*0*	*0*	0	1	0	1	**1**
v_2	*0*	*1*	0	1	1	1	**1**
v_3	*1*	*0*	0	1	1	1	**1**
v_4	*1*	*1*	1	0	1	1	**0**

F.b.f. soddisfacibili

- Una f.b.f. P si dice **soddisfacibile** se esiste almeno una interpretazione v tale che v(P)=1.
- Un'interpretazione v che soddisfa P si dice **modello** di P ($v \vDash P$)
- Esempio $P = \sim(A \wedge B) \wedge (A \Rightarrow (B \vee A))$ è soddisfacibile perché l'interpretazione v_1 è un modello per P ($v_1 \vDash P$).

Tautologie

- Una f.b.f P per cui ogni interpretazione è un modello si dice **tautologia** e si scrive $\vDash P$.
- Esempio: $(A \Rightarrow (A \vee B))$ è una tautologia

int	**A**	**B**	**$(A \vee B)$**	**$(A \Rightarrow (A \vee B))$**
v_1	*0*	*0*	0	1
v_2	*0*	*1*	1	1
v_3	*1*	*0*	1	1
v_4	*1*	*1*	1	1

F.b.f. insoddisfacibili

- Una f.b.f. P che non ammette modelli si dice **insoddisfacibile**. ($\nvDash$P)
- Esempio: (A $\wedge$ ~A) è insoddisfacibile.

int	A	~A	(A $\wedge$ ~A)
v_1	*0*	*1*	0
v_2	*1*	*0*	0

Numero di interpretazioni

- Se una proposizione P contiene *n* proposizioni atomiche distinte quante possibili interpretazioni può avere?
- n = 1, sono 2. Es: la negazione
- n = 2, sono 4. Es: la disgiunzione
- In generale sono 2^n.

Tautologie e f.b.f. insoddisf.

- P è una tautologia se e solo se ~ P è insoddisfacibile.
  - P è una tautologia
  - Ogni interpretazione è un modello per P
  - ~ P non ha modelli
  - ~ P è insoddisfacibile

Insiemi

- I concetti di modello, soddisfacibilità e insoddisfacibilità si possono estendere ad un insieme Γ di fbf:
  - un **modello** per Γ è una interpretazione v che sia modello per ogni fbf di Γ
  - Γ è **soddisfacibile** se ammette un modello
  - Γ è **insoddisfacibile** se nessuna interpretazione è un modello per Γ.

Conseguenza semantica

- Una f.b.f. P è **conseguenza semantica** di un insieme Γ di f.b.f. e si scrive $\Gamma \models P$, se ogni modello di Γ è un modello per P.
- In particolare P è ***conseguenza semantica*** di Q se ogni modello di Q è modello di P.

Esempio

- $\Gamma = \{\sim A, A \Rightarrow B\}$ e $P = (\sim A \vee B)$
- modelli di Γ sono v_1 e v_2 come mostra la tabella:

int	A	B	~A	(A⇒B)
v_1	**0**	**0**	**1**	**1**
v_2	**0**	**1**	**1**	**1**
v_3	1	0	0	0
v_4	1	1	0	1

Esempio

- modelli di Γ sono v_1 e v_2:
  - $v_1(A)=0$, $v_1(B)=0$
  - $v_2(A)=0$, $v_2(B)=1$
- v_1 e v_2 sono modelli di P

int	A	B	$\sim A$	$P = (\sim A \vee B)$
v_1	0	0	1	1
v_2	0	1	1	1

Esempio

- $\Gamma \models P$ vuol dire che P è vera per tutte le interpretazioni che sono modelli di Γ.
- $\Gamma = \{\sim A, A \Rightarrow B\}$ e $P = (\sim A \vee B)$
- modelli di Γ sono v_1 e v_2:
  - $v_1(A)=0$, $v_1(B)=0$
  - $v_2(A)=0$, $v_2(B)=1$
- v_1 e v_2 sono modelli di P, quindi $\mathbf{\Gamma \models P}$
- **Osservazione** P ha anche come modello v:
  - $v(A)=1$, $v(B)=1$

Fondamenti di Logica Matematica

Esercitazione 4

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Es 1

- {A,B,C} è l'insieme di formule atomiche e v un'interpretazione così definita :
  - v(A) = 1
  - v(B) = 1
  - v(C) = 0
- Calcolare $v(\sim(A \wedge B) \vee (A \Rightarrow (B \vee C)))$

Es 1 soluzione

- $v(A) = 1$, $v(B) = 1$, $v(C) = 0$

- $v(\sim(A \wedge B) \vee (A \Rightarrow (B \vee C)))=$
- $\max(v(\sim(A \wedge B)), v(A \Rightarrow (B \vee C)) =$
- $\max(1- v(A \wedge B), \max(1-v(A), v(B \vee C)))=$
- $\max(1-\min(v(A), v(B)), \max(1-v(A), \max(v(B), v(C))))=$
- $\max(0, 1) =1$

Es 2

- Mostrare tutte le possibili interpretazioni della proposizione $P = (\sim A \Rightarrow B) \wedge (B \wedge A)$:

int	A	B	~ A	~A ⇒ B	(A∧B)	P
v_1	*0*	*0*	1	0	0	**0**
v_2	*0*	*1*	1	1	0	**0**
v_3	*1*	*0*	0	1	0	**0**
v_4	*1*	*1*	0	1	1	**1**

Es 3

- Siano
  - $\Gamma = \{\sim A, A \vee B, B \vee (\sim C \wedge D)\}$
  - $P = (\sim A \vee C)$
- Dimostrare che $\Gamma \models P$?

Es 3 soluzione

- modelli di Γ sono:
  - $v_1(A)=0, v_1(B)=1, v_1(C)=0, v_1(D)=0$
  - $v_2(A)=0, v_2(B)=1, v_2(C)=0, v_2(D)=1$
  - $v_3(A)=0, v_3(B)=1, v_3(C)=1, v_3(D)=0$
  - $v_4(A)=0, v_4(B)=1, v_4(C)=1, v_4(D)=1$
- Sono anche modelli per P?

Es 3 soluzione

- Vediamo la tavola di verità per modelli di Γ:

mod	A	B	C	D	$\sim A$	$(\sim A \vee C)$
v_1	0	1	0	0	1	1
v_2	0	1	0	1	1	1
v_3	0	1	1	0	1	1
v_4	0	1	1	1	1	1

Fondamenti di Logica Matematica
Lezione 5

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Teorema di deduzione semantica
- Teorema di compattezza
- Equivalenza semantica

Conseguenza semantica

- Una f.b.f. P è **conseguenza semantica** di un insieme Γ di f.b.f. e si scrive $\Gamma \models P$, se ogni modello di Γ è un modello per P.
- In particolare P è ***conseguenza semantica*** di Q se ogni modello di Q è modello di P.

Th. deduzione semantica

- Teorema di deduzione semantica:
  - $\mathbf{\Gamma \cup \{P\} \models Q}$ **sse** $\mathbf{\Gamma \models (P \Rightarrow Q)}$.
- Dim ($\Rightarrow$):
  - Ip.: $\Gamma \cup \{P\} \models Q$ Tesi: $\Gamma \models (P \Rightarrow Q)$.
- Sia v un modello per Γ,
  - caso1: v(P)=1. Se v(P)=1 e v è un modello per Γ dall'ipotesi si ha v(Q)=1 e quindi v(P$\Rightarrow$Q)=1,
  - caso2: v(P)=0. Si ha v(P$\Rightarrow$Q)= 1
- In entrambi i casi v è un modello per P$\Rightarrow$Q.

Th. deduzione semantica

- Teorema di deduzione semantica:
  - $\mathbf{\Gamma \cup \{P\} \vDash Q}$ **sse** $\mathbf{\Gamma \vDash (P \Rightarrow Q).}$
- Dim ($\Leftarrow$):
  - Ip. $\Gamma \vDash (P \Rightarrow Q)$ Tesi: $\Gamma \cup \{P\} \vDash Q$
- Sia v un modello per $\Gamma \cup \{P\}$, allora v è un modello per Γ e per P, essendo un modello per Γ dall'ipotesi si ha $v(P \Rightarrow Q)=1$ che assieme a $v(P)=1$ implica $v(Q)=1$.
- Il teorema è vero.

Deduzione semantica

- Teorema di deduzione semantica:
  - $\mathbf{\Gamma \cup \{P\} \vDash Q}$ **sse** $\mathbf{\Gamma \vDash (P \Rightarrow Q).}$
- Corollario al teorema:
  - $\mathbf{P \vDash Q}$ **sse** $\mathbf{\vDash (P \Rightarrow Q)}$
- Quindi per dimostrare che $P \vDash Q$ basta dimostrare che $(P \Rightarrow Q)$ è una tautologia!

Conseguenza e insodd.

Teorema

$\Gamma \vDash P$ se e solo se $\Gamma \cup \{\sim P\}$ è insoddisfacibile

- Dim ($\Rightarrow$):
  - Ip.: $\Gamma \vDash P$ Tesi: $\Gamma \cup \{\sim P\}$ è insoddisfacibile
- Sia v una qualunque interpretazione,
  - Caso1: se v è un modello per Γ dall'ipotesi si ha $v(P)=1$ e quindi $v(\sim P)=0$ quindi v non è un modello per $\Gamma \cup \{\sim P\}$,
  - Caso2: se v non è un modello per Γ, non può essere sicuramente un modello per $\Gamma \cup \{\sim P\}$.

Conseguenza e insodd.

$\Gamma \vDash P$ se e solo se $\Gamma \cup \{\sim P\}$ è insoddisfacibile

- Dim ($\Leftarrow$):
  - Ip.: $\Gamma \cup \{\sim P\}$ è insoddisfacibile Tesi: $\Gamma \vDash P$
- Sia v un modello per Γ, allora non dovendo essere un modello per $\Gamma \cup \{\sim P\}$ si ha $v(\sim P)=0$ e quindi $v(P)=1$, dunque ogni modello di Γ è modello per P.

Esempio: testo

- Ipotesi:
  - Se Carlo è americano e Giovanni non è francese, allora Elena è tedesca
  - Se Elena è tedesca, allora Lucia è spagnola o Giovanni è francese
  - Se Lucia non è spagnola allora Carlo è americano
  - Giovanni non è francese.
- Tesi:
  - Lucia è spagnola

Esempio: formalizzazione

- Formalizzazione delle proposizioni:
  - A = Carlo è americano
  - B = Giovanni è francese
  - C = Elena è tedesca
  - D = Lucia è spagnola
- Ipotesi:
  - $(A \wedge \sim B) \Rightarrow C$
  - $C \Rightarrow D \vee B$
  - $\sim D \Rightarrow A$
  - $\sim B$
- Tesi:
  - D

Esempio: formalizzazione

- Formalizzazione del problema:
  - $((A \land \sim B) \Rightarrow C) \land (C \Rightarrow D \lor B) \land (\sim D \Rightarrow A) \land (\sim B) \vDash D$
- Per il teorema di deduzione semantica:
  - $\vDash ((A \land \sim B) \Rightarrow C) \land (C \Rightarrow D \lor B) \land (\sim D \Rightarrow A) \land (\sim B) \Rightarrow D$
- Basta dimostrare che è una tautologia!

Esempio: tabella di verità

(A	∧	~	B	⇒	C)	∧	(C	⇒	D	∨	B)	∧	(~	D	⇒	A)	∧	(~	B)	⇒	D
0	0	1	0	**1**	0	**1**	0	**1**	0	0	0	**0**	1	0	**0**	0	**0**	**1**	0	**1**	0
0	0	1	0	**1**	0	**1**	0	**1**	1	1	0	**1**	0	1	**1**	0	**1**	**1**	0	**1**	1
0	0	1	0	**1**	1	**0**	1	**0**	0	0	0	**0**	1	0	**0**	0	**0**	**1**	0	**1**	0
0	0	1	0	**1**	1	**1**	1	**1**	1	1	0	**1**	0	1	**1**	0	**1**	**1**	0	**1**	1
0	0	0	1	**1**	0	**1**	0	**1**	0	1	1	**0**	1	0	**0**	0	**0**	**0**	1	**1**	0
0	0	0	1	**1**	0	**1**	0	**1**	1	1	1	**1**	0	1	**1**	0	**0**	**0**	1	**1**	1
0	0	0	1	**1**	1	**1**	1	**1**	0	1	1	**0**	1	0	**0**	0	**0**	**0**	1	**1**	0
0	0	0	1	**1**	1	**1**	1	**1**	1	1	1	**1**	0	1	**1**	0	**0**	**0**	1	**1**	1
1	1	1	0	**0**	0	**0**	0	**1**	0	0	0	**0**	1	0	**1**	1	**0**	**1**	0	**1**	0
1	1	1	0	**0**	0	**0**	0	**1**	1	1	0	**0**	0	1	**1**	1	**0**	**1**	0	**1**	1
1	1	1	0	**1**	1	**0**	1	**0**	0	0	0	**0**	1	0	**1**	1	**0**	**1**	0	**1**	0
1	1	1	0	**1**	1	**1**	1	**1**	1	1	0	**1**	0	1	**1**	1	**1**	**1**	0	**1**	1
1	0	0	1	**1**	0	**1**	0	**1**	0	1	1	**1**	1	0	**1**	1	**0**	**0**	1	**1**	0
1	0	0	1	**1**	0	**1**	0	**1**	1	1	1	**1**	0	1	**1**	1	**0**	**0**	1	**1**	1
1	0	0	1	**1**	1	**1**	1	**1**	0	1	1	**1**	1	0	**1**	1	**0**	**0**	1	**1**	0
1	0	0	1	**1**	1	**1**	1	**1**	1	1	1	**1**	0	1	**1**	1	**0**	**0**	1	**1**	1

Th. di compattezza

- **Teorema:** Un insieme Γ di f.b.f. è soddisfacibile sse ogni suo sottoinsieme Δ finito è soddisfacibile.

Dim. del Th. di compattezza

Teorema: Un insieme Γ di f.b.f. è soddisfacibile sse ogni suo sottoinsieme Δ finito è soddisfacibile.

Dim:

1) Γ è soddisfacibile $\Rightarrow$ ogni suo sottoinsieme Δ finito è soddisfacibile
2) ogni sottoinsieme Δ finito di Γ è soddisfacibile $\Rightarrow$ Γ è soddisfacibile

Dim. del Th. di compattezza

1)Γ è soddisfacibile ⇒ ogni suo sottoinsieme Δ finito è soddisfacibile

Dim:

è sempre vero per la definizione di soddisfacibilità, infatti ogni modello di Γ deve essere un modello di ogni formula in Γ. Quindi se Γ è soddisfacibile ogni sottoinsieme di Γ ha almeno un modello.

Dim. del Th. di compattezza

2)Ogni sottoinsieme Δ finito di Γ è soddisfacibile ⇒ Γ è soddisfacibile

Dim: Basta dim. che è possibile trovare un modello comune a tutti i sottoinsiemi finiti di Γ. Tale modello è quindi un modello per tutte le formule di Γ, in quanto ogni formula può essere vista come un sottoinsieme finito di Γ, ed è perciò un modello per Γ.

Dim. del Th. di compattezza

Dim cont.: Tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegna alla lettera enunciativa A_1 un valore fissato $v(A_1)$.

Se così non fosse ci sarebbero due sottoinsiemi finiti Δ_1 e Δ_2 di Γ tali che:

- in tutti i modelli del primo A_1 vale 0,
- in tutti i modelli del secondo ad A_1 vale 1

ma allora il sottoinsieme finito $\Delta_1 \cup \Delta_2$ di Γ è insoddisfacibile, ma questo è assurdo.

Dim. del Th. di compattezza

Dim cont.:

- Supponiamo che
  - tutti i sottoinsiemi finiti di Γ abbiano almeno un modello che assegna alle lettere enunciative A_1, A_2, ..., A_n la n-upla di valori $v(A_1)$, $v(A_2)$, ..., $v(A_n)$
- Dimostriamo che
  - è possibile assegnare un valore $v(A_{n+1})$ ad A_{n+1} in modo che tutti i sottoinsiemi finiti di Γ abbiano almeno un modello che assegni alle lettere enunciative A_1, A_2, ..., A_{n+1} i valori $v(A_1)$, $v(A_2)$, ..., $v(A_n)$, $v(A_{n+1})$.

Dim. del Th. di compattezza

Dim cont.:

- Se tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegna alle prime n+1 lettere enunciative $A_1, ..., A_{n+1}$ i valori $v(A_1), ..., v(A_n)$, **$v(A_{n+1})=0$**, abbiamo concluso.
- Altrimenti esiste almeno un sottoinsieme finito Δ_3 di Γ tale che tutti i modelli di Δ_3 che assegnano alle prime n lettere i valori $v(A_1), ..., v(A_n)$, devono assegnare ad A_{n+1} il valore 1.
  - dimostriamo per assurdo che i sottoinsiemi finiti di Γ hanno almeno un modello che assegna alle prime n+1 lettere i valori $v(A_1), ..., v(A_n)$, **v(An+1)=1**.

Dim. del Th. di compattezza

Dim cont.:

- Tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegna alle prime n+1 lettere i valori $v(A_1), ..., v(A_n)$, **v(An+1)=1**.
  - Altrimenti esisterebbe almeno un sottoinsieme finito Δ_4 di Γ tale che tutti i modelli di Δ_4 che assegnano alle prime n lettere i valori $v(A_1), ...,$ $v(A_n)$, devono assegnare ad A_{n+1} il valore 0.
  - Ma allora il sottoinsieme finito $\Delta_3 \cup \Delta_4$ di Γ, non potrebbe avere un modello che assegni alle prime n lettere enunciative $A_1, A_2, ..., A_n$ valori $v(A_1), ..., v(A_n)$, assurdo (per l'ipotesi fatta).

Dim. del Th. di compattezza

Dim cont.:

- Abbiamo dimostrato che
  - Tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegna alla lettera A_1 un valore fissato $v(A_1)$.
  - **Se** tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegna ad $A_1, A_2, ..., A_n$ la n-upla di valori $v(A_1), v(A_2), ..., v(A_n)$
  - **allora** tutti i sottoinsiemi finiti di Γ hanno almeno un modello che assegni alle lettere $A_1, A_2, ..., A_{n+1}$ i valori $v(A_1), v(A_2), ..., v(A_n), v(A_{n+1})$.

Dim. del Th. di compattezza

Dim cont.:

- per ogni P in Γ con n suff. grande avremo un insieme $\{A_1, ..., A_n\}$ che contiene tutte le lettere in P.
- {P} un sottoinsieme finito di Γ
- quindi {P} ha un modello con valori $v(A_1), ..., v(A_n)$

- Quindi v è un modello per ogni P in Γ.
- esiste un modello comune a tutti i P di Γ.
- Γ è soddisfacibile!

Th. di compattezza

- **Osservazione:** la dimostrazione non è costruttiva, non abbiamo costruito il modello comune ma abbiamo solo dimostrato che esiste.

Corollari

- **Teorema:** Un insieme Γ di f.b.f. è soddisfacibile sse ogni suo sottoinsieme Δ finito è soddisfacibile.
- **Corollario:** Γ è insoddisfacibile sse esiste un sottoinsieme finito Δ di Γ insoddisfacibile
- **Corollario:** $\Gamma \models P$ sse esiste un sottoinsieme finito Δ di Γ tale che $\Delta \models P$.

Corollari

- **Corollario:** $\Gamma \models P$ sse esiste un sottoinsieme finito Δ di Γ tale che $\Delta \models P$.
- **Dim.:**
  - ($\Rightarrow$): Se $\Gamma \models P$, allora $\Gamma \cup \{\sim P\}$ è insoddisfacibile e quindi, per il primo corollario, esiste un suo sottoinsieme finito$\Delta \cup \{\sim P\}$ insoddisfacibile e perciò $\Delta \models P$.
  - ($\Leftarrow$): L'implicazione contraria e' ovvia.

Osservazioni su $\Rightarrow$

- Se oggi è martedì (A) allora domani piove (B), oppure se domani piove allora oggi è martedì.
- **$(A \Rightarrow B) \vee (B \Rightarrow A)$**

int	A	B	$(A \Rightarrow B)$	$(B \Rightarrow A)$	$(A \Rightarrow B) \vee (B \Rightarrow A)$
v_1	*0*	*0*	1	1	1
v_2	*0*	*1*	1	0	1
v_3	*1*	*0*	0	1	1
v_4	*1*	*1*	1	1	1

Osservazioni su $\Rightarrow$

- È una tautologia anche se la frase non ha molto senso!!!
- Denotazione vs. senso
- $A \Rightarrow B$ viene letta normalmente come:
  - B si ottiene mediante un ragionamento logico da A. Per la logica prop. NO!
  - Questo concetto viene espresso con $A \vdash B$, ovvero se è vero A possiamo concludere B.

Equivalenza semantica

- Una formula P è **semanticamente equivalente a** Q (scriveremo $P \equiv Q$) se tutti e soli i modelli di P sono modelli di Q, in altre parole se P è *conseguenza semantica* di Q e se Q è *conseguenza semantica* di P.
- $P \equiv Q$ se e solo se $P \Leftrightarrow Q$ è una tautologia.
  - Oss: $P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

Conseguenza ed equivalenza

- $\sim A \wedge (A \Rightarrow B) \models \sim A \vee B$
- ma **non** è vero che $\sim A \wedge (A \Rightarrow B) \equiv \sim A \vee B$
- modelli di $\sim A \wedge (A \Rightarrow B)$:
  - $v_1(A)=0$, $v_1(B)=0$
  - $v_2(A)=0$, $v_2(B)=1$
- modelli di $\sim A \vee B$:
  - $v_1(A)=0$, $v_1(B)=0$
  - $v_2(A)=0$, $v_2(B)=1$
  - $v_3(A)=1$, $v_3(B)=1$

Equivalenze semantiche

- Idempotenza:
  - $P \vee P \equiv P$
  - $P \wedge P \equiv P$
- Commutatività:
  - $P \vee Q \equiv Q \vee P$
  - $P \wedge Q \equiv Q \wedge P$
- Associatività:
  - $(P \vee Q) \vee R \equiv P \vee (Q \vee R)$
  - $(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$

Equivalenze semantiche

- Assorbimento:
  - $P \vee (P \wedge Q) \equiv P$
  - $P \wedge (P \vee Q) \equiv P$
- Distributività:
  - $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$
  - $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$
- Leggi di De Morgan:
  - $\sim(P \vee Q) \equiv \sim P \wedge \sim Q$
  - $\sim(P \wedge Q) \equiv \sim P \vee \sim Q$

Equivalenze semantiche

- Doppia negazione:
  - $\sim (\sim P) \equiv P$
- Implicazione:
  - $P \Rightarrow Q \equiv \sim P \vee Q$
  - $P \Rightarrow Q \equiv \sim(P \wedge \sim Q)$
- Elementi neutri:
  - $P \equiv T \wedge P$
  - $P \equiv F \vee P$

Una dimostrazione

- **Proposizione:** $\sim(P \vee Q) \equiv \sim P \wedge \sim Q$
- **Dim:** sia v un'interpretazione qualsiasi
- $v(\sim(P \vee Q)) = 1 - v(P \vee Q)$
  $= 1 - \max(v(P),v(Q))$
  $= \min(1-v(P),1-v(Q))$
  $= v(\sim P \wedge \sim Q)$
- quindi tutti i modelli di $\sim(P \vee Q)$ sono modelli di $(\sim P \wedge \sim Q)$ e viceversa.
- Per definizione di equivalenza semantica abbiamo che:
  $$\sim(P \vee Q) \equiv \sim P \wedge \sim Q$$

Funzioni di verità

- Sia P una f.b.f. con *n* proposizioni atomiche distinte $A_1, ..., A_n$, la **funzione di verità** f_P è la funzione f_P: $\{0,1\}^n \rightarrow \{0,1\}$ tale che
  - $\forall(a_1, ..., a_n) \in \{0,1\}^n$, $f(a_1, ..., a_n)=v(P)$,
  - dove v è una interpretazione tale che $v(A_i)= a_i$, $\forall A_i$ in P.

Es. funzioni di verità

- La funzione di verità di $A \wedge B$ è:
  - $f_{A \wedge B}: \{0,1\}^2 \rightarrow \{0,1\}$ così definita
  - $f_{A \wedge B}(0,0) = 0 = v_1(A \wedge B)$ con $v_1(A)=0$, $v_1(B)=0$
  - $f_{A \wedge B}(0,1) = 0 = v_2(A \wedge B)$ con $v_2(A)=0$, $v_2(B)=1$
  - $f_{A \wedge B}(1,0) = 0 = v_3(A \wedge B)$ con $v_3(A)=1$, $v_3(B)=0$
  - $f_{A \wedge B}(1,1) = 1 = v_4(A \wedge B)$ con $v_4(A)=1$, $v_4(B)=1$

Fondamenti di Logica Matematica
Lezione 6

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Completezza funzionale
- Forme normali
- Sostituzioni
- Dualità

Connettivi derivabili

- Un connettivo è semanticamente derivabile se è possibile definirlo in funzione di altri connettivi
- Esempio $A \Leftrightarrow B \equiv (A \Rightarrow B) \wedge (B \Rightarrow A)$

A	B	$(A \Rightarrow B)$	$(B \Rightarrow A)$	$(A \Rightarrow B) \wedge (B \Rightarrow A)$	$A \Leftrightarrow B$
0	*0*	1	1	1	1
0	*1*	1	0	0	0
1	*0*	0	1	0	0
1	*1*	1	1	1	1

Completezza funzionale

- Un insieme di connettivi logici si dice **funzionalmente completo** se per ogni funzione $f: \{0,1\}^n \rightarrow \{0,1\}$ esiste una f.b.f. P costituita mediante questi e tale che $f_P = f$.
- Ovvero un insieme di connettivi è completo se ogni altro connettivo può essere derivato da essi.

Es. insiemi completi

- Gli insiemi:
  - $\{\sim, \wedge\}$
  - $\{\sim, \vee\}$
  - $\{\Rightarrow, F\}$
  - $\{\sim, \vee, \wedge, \Rightarrow, F, T\}$ (quello usato da noi)
- sono funzionalmente completi
- Esempio: con $\{\Rightarrow, F\}$ si deriva
  - $A \wedge B = (((A \Rightarrow F) \Rightarrow F) \Rightarrow (B \Rightarrow F)) \Rightarrow F$
  - Molto complicato!

Forme normali

- Trasformiamo una fbf in un'altra equivalente che ha una **forma canonica**.
- Si trasforma la fbf originale sostituendo una sua componente con altre equivalenti fino ad arrivare alla forma canonica
- La forma canonica è detta **normale** perché non si può ulteriormente sostituire

Disgiunzioni e congiunzioni

- Un **letterale** è una formula atomica o la sua negazione (A o ~A)
- Una **disgiunzione** di fbf P_1, P_2, ..., P_n è la formula $P_1 \vee P_2 \vee ... \vee P_n$
- Una **congiunzione** di fbf P_1, P_2, ..., P_n è la formula $P_1 \wedge P_2 \wedge ... \wedge P_n$

Forma normale congiuntiva

- Una fbf P è detta in **forma normale congiuntiva** (**FNC**) sse:
  - $P = P_1 \wedge P_2 \wedge ... \wedge P_n$ con $n \geq 1$
  - e $\forall i = 1,...,n$ P_i è una disgiunzione di letterali:
    - $P_i = L_1 \vee L_2 \vee ... \vee L_m$
- Es. $(A \vee \sim B \vee C) \wedge B \wedge \sim D \wedge (A \vee D)$

Forma normale disgiuntiva

- Una fbf P è detta in **forma normale disgiuntiva** (**FND**) sse:
  - $P = P_1 \vee P_2 \vee ... \vee P_n$ con $n \geq 1$
  - e $\forall i = 1,...,n$ P_i è una congiunzione di letterali:
    - $P_i = L_1 \wedge L_2 \wedge ... \wedge L_m$
- Es. $(A \wedge \sim B \wedge C) \vee B \vee \sim D \vee (A \wedge D)$

Trasformazione

- Per ogni fbf P esistono una forma normale congiuntiva P^C e una forma normale disgiuntiva P^D, tali che $P \equiv P^C$ e $P \equiv P^D$ (lo dimostriamo formalmente dopo).
- Per la trasformazione si usano le regole di equivalenza, le formule di De Morgan e le regole distributive.

Osservazione

- Per ogni formula fbf P possono esistere più fbf in FNC e in FND equivalenti ad essa.
- Esempio:
  - $P = (A \wedge \sim B \wedge C) \vee (\sim(A \Rightarrow B) \wedge A) \vee (A \wedge B \wedge C)$
  1. $P \equiv (A \wedge C) \vee (\sim B \wedge A)$
  2. $P \equiv (\sim B \wedge A) \vee (A \wedge B \wedge C)$
  3. $P \equiv (A \wedge \sim B \wedge C) \vee (\sim B \wedge A) \vee (A \wedge B \wedge C)$
  - sono tutte e tre FND equivalenti a P

Trasformazione in FND

- **Passo 1** eliminare il connettivo $\Rightarrow$:
  - $P \Rightarrow Q \equiv \sim P \vee Q$
- **Passo 2** eliminazione di T e F
  - $F \equiv \sim A \wedge A$ (A è una qualsiasi formula atomica)
  - $T \equiv \sim A \vee A$ (A è una qualsiasi formula atomica)
- **Passo 3** portare le negazione all'interno e eliminare le doppie negazioni:
  - $\sim(P \vee Q) \equiv \sim P \wedge \sim Q$ (De Morgan)
  - $\sim(P \wedge Q) \equiv \sim P \vee \sim Q$ (De Morgan)
  - $\sim\sim P \equiv P$

Trasformazione in FND

- **Passo 4** portare le congiunzioni all'interno delle disgiunzioni:
  - $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$ (distr.)
  - $(P \vee S) \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R) \vee (S \wedge Q) \vee (S \wedge R)$ (distr.)

Esempio

- Trasformare $\sim(A \wedge (B \Rightarrow F))$ in FND:
- Passo 1
  - $\sim(A \wedge (\sim B \vee F))$
- Passo 2
  - $\sim(A \wedge (\sim B \vee (\sim A \wedge A)))$
- Passo 3
  - $\sim A \vee \sim(\sim B \vee (\sim A \wedge A))$
  - $\sim A \vee (\sim\sim B \wedge \sim(\sim A \wedge A))$
  - $\sim A \vee (B \wedge \sim(\sim A \wedge A))$
  - $\sim A \vee (B \wedge (\sim\sim A \vee \sim A))$
  - $\sim A \vee (B \wedge (A \vee \sim A))$

Esempio

- $\sim A \vee (B \wedge (A \vee \sim A))$
- Passo 4
  - $\sim A \vee ((B \wedge A) \vee (B \wedge \sim A))$
  - $\sim A \vee (B \wedge A) \vee (B \wedge \sim A)$ in forma FND

Trasformazione in FNC

- **Passo 1** eliminare il connettivo $\Rightarrow$:
  - $P \Rightarrow Q \equiv \sim P \vee Q$
- **Passo 2** eliminazione di T e F
  - $F \equiv \sim A \wedge A$ (A è una qualsiasi formula atomica)
  - $T \equiv \sim A \vee A$ (A è una qualsiasi formula atomica)
- **Passo 3** portare le negazione all'interno e eliminare le doppie negazioni:
  - $\sim(P \vee Q) \equiv \sim P \wedge \sim Q$ (De Morgan)
  - $\sim(P \wedge Q) \equiv \sim P \vee \sim Q$ (De Morgan)
  - $\sim\sim P \equiv P$

Trasformazione in FNC

- **Passo 4** portare le disgiunzioni all'interno delle congiunzioni:
  - $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$ (distr.)
  - $(P \wedge S) \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R) \wedge (S \vee Q) \wedge (S \vee R)$ (distr.)

Esempio

- Trasformare $\sim(A \wedge (B \Rightarrow (\sim A \wedge B)))$ in FNC:
- Passo 1
  - $\sim(A \wedge (\sim B \vee (\sim A \wedge B)))$
- Passo 3
  - $\sim A \vee \sim(\sim B \vee (\sim A \wedge B))$
  - $\sim A \vee (\sim\sim B \wedge \sim(\sim A \wedge B))$
  - $\sim A \vee (B \wedge \sim(\sim A \wedge B))$
  - $\sim A \vee (B \wedge (\sim\sim A \vee \sim B))$
  - $\sim A \vee (B \wedge (A \vee \sim B))$

Esempio

- $\sim A \vee (B \wedge (A \vee \sim B))$
- Passo 4
  - $(\sim A \vee B) \wedge (\sim A \vee A \vee \sim B)$ in forma FNC

Proprietà

- **Teorema:** Per ogni fbf P esistono almeno una forma normale congiuntiva P^C e una forma normale disgiuntiva P^D, tali che $P \equiv P^C$ e $P \equiv P^D$.
- *Dim.* Per costruzione. Basta utilizzare le equivalenze semantiche e seguire lo schema degli algoritmi:
  1. Si eliminano i connetti diversi da $\sim$, $\vee$, $\wedge$ utilizzando le equivalenze semantiche: $P \Rightarrow Q \equiv \sim P \vee Q$, $F \equiv \sim A \wedge A$ e $T \equiv \sim A \vee A$
  2. Si utilizza ripetutamente la legge di De Morgan per portare le negazioni davanti alle proposizioni atomiche
  3. Si utilizza la distributività per convertire P in P^C o P^D.

Forme normali complete

- Costruzione **FND completa** di P:
  - Siano A_1, ..., A_n le formule atomiche di P
  - Si costruisce la tabella di verità di P
  - Ogni linea che ha valore di verità 1 forma una disgiunzione così definita:
    - Sia v l'interpretazione associata a tale linea
    - Per ogni formula atomica e $v(A_i) = 1$ allora viene inserito A_i altrimenti $\sim A_i$.

Esempio

- Trasformare $\sim(A \wedge (B \Rightarrow (\sim A \wedge B)))$ nella FND completa:

int	~	(A	∧	(B	⇒	(~	A	∧	B)))
v_1	**1**	0	0	0	1	1	0	0	0
v_2	**1**	0	0	1	1	1	0	1	1
v_3	**0**	1	1	0	1	0	1	0	0
v_4	**1**	1	0	1	0	0	1	0	1

- FND completa: $(\sim A \wedge \sim B) \vee (\sim A \wedge B) \vee (A \wedge B)$

Forme normali complete

- Costruzione **FNC completa** di P:
  - Siano A_1, ..., A_n le formule atomiche di P
  - Si costruisce la tabella di verità di P
  - Ogni linea che ha valore di verità 0 forma una congiunzione così definita:
    - Sia v l'interpretazione associata a tale linea
    - Per ogni formula atomica e $v(A_i) = 0$ allora viene inserito A_i altrimenti ~ A_i.

Esempio

- Trasformare ~(A ∧ (B ⇒ (~A ∧ B))) nella FNC completa:

int	~	(A	∧	(B	⇒	(~	A	∧	B)))
v_1	**1**	0	0	0	1	1	0	0	0
v_2	**1**	0	0	1	1	1	0	1	1
v_3	**0**	1	1	0	1	0	1	0	0
v_4	**1**	1	0	1	0	0	1	0	1

FNC completa: (~A ∨ B)

Proprietà

- Data una fbf P, la FND completa e la FNC completa di P sono **uniche**.
  - Vero per costruzione utilizzando la tabella di verità.
- **Corollario** L'insieme dei connettivi $\{\sim, \wedge, \vee\}$ è funzionalmente completo
- **Corollario** L'insiemi di connettivi $\{\sim, \vee\}$ e $\{\sim, \wedge\}$ sono funzionalmente completi
  - $A \wedge B \equiv \sim(\sim A \vee \sim B)$
  - $A \vee B \equiv \sim(\sim A \wedge \sim B)$

Sostituzione

- Siano P e R due fbf e A una formula atomica. La sostituzione (R[P/A]) di P al posto di A in R si definisce in modo ricorsivo nel seguente modo:
  - Se R è una formula atomica diversa da A allora R[P/A] = R
  - Se $R \equiv A$ allora R[P/A] = P
  - Se $R \equiv \sim Q$ allora $(\sim Q)[P/A] = \sim Q[P/A]$
  - Se $R \equiv Q_1 \vee Q_2$ allora $(Q_1 \vee Q_2)[P/A] = Q_1[P/A] \vee Q_2[P/A]$
  - Se $R \equiv Q_1 \wedge Q_2$ allora $(Q_1 \wedge Q_2)[P/A] = Q_1[P/A] \wedge Q_2[P/A]$
  - Se $R \equiv Q_1 \Rightarrow Q_2$ allora $(Q_1 \Rightarrow Q_2)[P/A] = Q_1[P/A] \Rightarrow Q_2[P/A]$

Esempio

- $R = \sim(A \wedge (B \Rightarrow (\sim A \wedge B)))$
- $P = (A \vee B)$

- $R[P/A] =$
- $= \sim((A \vee B) \wedge (B \Rightarrow (\sim (A \vee B) \wedge B)))$
- $P[R/A] =$
- $= ((\sim(A \wedge (B \Rightarrow (\sim A \wedge B)))) \vee B)$

Proprietà

- Sia v un'interpretazione.
  - Se $v(P) = v(Q)$ allora:
  - per ogni R, $v(R[P/A]) = v(R[Q/A])$
- Dim per induzione strutturale
- Per esercizio

Th di sostituzione

- Sia P ≡ Q
  - allora R[P/A] ≡ R[Q/A]
- Dim
  - Si v una interpretazione.
  - Per ipotesi P ≡ Q quindi v(P) = v(Q)
  - Per la proprietà di prima:
    - v(R[P/A]) = v(R[Q/A])
  - Questo vale per ogni interpretazione e quindi per def. di equivalenza si ha che
    - R[P/A] ≡ R[Q/A]

Sostituzione simultanea

- $R[P_1, \dots P_n/A_1, \dots A_n]$ si ottiene sostituendo P_i al posto di A_i per tutti gli i **simultaneamente**.
- Es. R = ~(A ∧ (B ⇒ (~A ∧ B))),
  P = (A ∨ B) e Q = (~A ∧ B)
  - R[P,Q / A,B] =
  - ~((A ∨ B) ∧ ((~A ∧ B) ⇒ (~ (A ∨ B) ∧ (~A ∧ B))))

Dualità

- $A \wedge B = 1$ sse $A = B = 1$
- $A \vee B = 0$ sse $A = B = 0$
- Sono operatori **duali** ovvero ognuno deriva dall'altro rovesciando il ruolo di 0 e 1.

P	Q	$P \wedge Q$
0	0	0
0	1	0
1	0	0
1	1	1

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

Trasformazione: $^{\perp}$

- La funzione $^{\perp}$: FBF $\rightarrow$ FBF soddisfa:
  - $P^{\perp} = {\sim}P$ se P è una formula atomica
  - $(P \wedge Q)^{\perp} = P^{\perp} \vee Q^{\perp}$
  - $(P \vee Q)^{\perp} = P^{\perp} \wedge Q^{\perp}$
  - $({\sim}P)^{\perp} = {\sim}P^{\perp}$

Esempio

- $((A \wedge \sim B) \vee C)^{\perp} =$
  - $((A \wedge \sim B)^{\perp} \wedge C^{\perp} =$
  - $(A^{\perp} \vee (\sim B)^{\perp}) \wedge \sim C =$
  - $(\sim A \vee (\sim\sim B)) \wedge \sim C =$
  - $(\sim A \vee B) \wedge \sim C$
- La trasformazione ha l'effetto di:
  - $\mathbf{P^{\perp} \equiv \sim P}$
- **Non è la funzione di dualità!**

Funzione di dualità

- La funzione di dualità ^{d} : FBF $\rightarrow$ FBF soddisfa:
  - $\mathbf{P^{d} = P}$ se P è una formula atomica
  - $(P \wedge Q)^{d} = P^{d} \vee Q^{d}$
  - $(P \vee Q)^{d} = P^{d} \wedge Q^{d}$
  - $(\sim P)^{d} = \sim P^{d}$

Esempio

- $(P \wedge Q)^d =$
  - $P^d \vee Q^d =$
  - $P \vee Q$
- È la funzione giusta!!!
- **Osservazione:** la funzione di dualità consiste nello scambiare gli operatori $\wedge$ e $\vee$.

Th di dualità

- $P \equiv Q \Leftrightarrow P^d \equiv Q^d$
- Esempio: Leggi di De Morgan
- Prima legge:
  - $\mathbf{\sim(P \vee Q) \equiv \sim P \wedge \sim Q}$
  - $(\sim(P \vee Q))^d \equiv (\sim P \wedge \sim Q)^d$
  - $\mathbf{\sim(P \wedge Q) \equiv \sim P \vee \sim Q}$
- Seconda legge!

Fondamenti di Logica Matematica
Esercitazione 6

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Es 1

- Quali delle seguenti formule sono tautologie, quali soddisfacibili e quali insoddisfacibili?
  - $(A \wedge (A \Rightarrow B)) \Rightarrow B$
  - $((A \wedge \sim B) \Rightarrow C)$
  - $A \vee \sim A$
  - $(A \wedge B) \Rightarrow A$
  - $A \Rightarrow (A \wedge B)$
  - $A \wedge \sim A$

Es 2

- I seguenti insiemi di formule sono soddisfacibili?
  - $\{(A \wedge B) \Rightarrow A \vee B\ ,\ A \wedge \sim A,\ B\}$
  - $\{A \vee \sim A,\ (A \Rightarrow B) \Rightarrow A,\ \sim A \vee B\}$

Es 3

- Provare che
  - $F \vee \sim A \equiv \sim A$
- Sia v un'interpretazione qualsiasi
- $v(F \vee \sim A) =$
  - $\max(v(F), v(\sim A)) =$
  - $\max(0, 1 - v(A)) =$
  - $1 - v(A) =$
  - $v(\sim A)$.
- Per definizione di equivalenza semantica abbiamo che $F \vee \sim A \equiv \sim A$
- Cosa ci dice il teorema di dualità?
  - $T \wedge \sim A \equiv \sim A$!

Es 4

- Trovare le forme FND e FNC equivalenti alle seguenti formule:
  - $(\sim(A \wedge \sim B) \Rightarrow C)$
  - $(\sim((A \wedge \sim B) \vee C) \wedge \sim(A \vee B))$
  - $((A \wedge C) \Rightarrow (A \vee B)) \wedge \sim(C \vee B)$

Es 5

- Data le seguente tabella di verità scrivere la fbf descritta, nelle due forme normali complete :

A	B	C	P
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Es 6

- Calcolare la forma duale e $^{\perp}$ delle seguenti formule:
  - $(\sim((A \wedge \sim B) \vee C) \wedge \sim(A \vee B))$
  - $((A \wedge B \wedge \sim C) \vee (\sim A \vee B)) \wedge \sim(C \vee \sim B)$
- Utilizzare il teorema di dualità per dimostrare:
  - $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$
- Sapendo che:
  - $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$

Es 7

Sia $P = (\sim((A \wedge \sim B) \vee C) \wedge \sim(A \Rightarrow B))$ e $R = (C \vee \sim B)$

- Calcolare P[R/B].
- Trasformare P[R/B] in forma FND.
- Trasformare P[R/B] in forma FNC.
- Calcolare la forma completa FND di P[R/B].
- Calcolare la forma completa FNC di P[R/B].
- Fra tutte le forme calcolate quale ha il minor numero di letterali?

Fondamenti di Logica Matematica

Lezione 7

Valentina Ciriani

DTI - Università di Milano

www.dti.unimi.it/~ciriani

Contenuti della lezione

- Complessità computazionale dei problemi della logica proposizionale:
  - SAT
  - TAUTOLOGY

Soddisfacibilità

- Problema **SAT**: la f.b.f. proposizionale P è soddisfacibile?
- Con una tabella di verità è possibile decidere se una f.b.f. è soddisfacibile.
- Se P ha *n* formule atomiche distinte, la tabella di verità di P ha $\boldsymbol{2^n}$ righe (una per ogni interpretazione)
- L'algoritmo per SAT basato sulla tabella di verità ha complessità al caso pessimo $O(2^n)$.

È possibile fare meglio?

- Esiste un algoritmo polinomiale
  - che stabilisca se una formula ben formata proposizionale è soddisfacibile?
  - ovvero che risolva il problema SAT?
- La risposta per ora non esiste:
  - SAT è un problema NP-compelto!
  - Tutti gli algoritmi conosciuti per SAT hanno complessità esponenziale

Tautologia

- Problema **TAUTOLOGY**: la f.b.f. proposizionale P è una tautologia?
- Sembrerebbe più difficile di SAT perché
  - non ci possiamo accontentare di trovare un modello di P (come per SAT)
  - ma dobbiamo verificare che tutte le 2^n interpretazioni siano modelli di P

Tautologia

- Una formula P è una tautologia
  - se e solo se la sua negata è insoddisfacibile
  - ovvero se e solo se ~P non è soddisfacibile
- TAUTOLOGY è il problema complementare di SAT

NP e co-NP

- SAT è in NP:
  - la **verifica** che una interpretazione v, data da un oracolo, sia un modello di P ha costo polinomiale
- Non sappiamo se TAUTOLOGY sia in NP:
  - la verifica che P sia una tautologia controllando tutte le 2^n interpretazioni non ha costo polinomiale!
  - non conosciamo una verifica polinomiale
- Dato che il suo complementare (SAT) è in NP allora si dice che TAUTOLOGY è in **co-NP**

Conclusioni

- Nonostante stiamo utilizzando un linguaggio logico molto limitato:
  - SAT è un problema **NP-completo**
  - TAUTOLOGY è un problema in **co-NP**
- Gli algoritmi di calcolo che vedremo hanno complessità esponenziale al caso pessimo.

Fondamenti di Logica Matematica

Esercitazione 7

Valentina Ciriani

DTI - Università di Milano

www.dti.unimi.it/~ciriani

Esercizio: ind. strutturale

Dimostrare per induzione strutturale la seguente proposizione.

Sia v un'interpretazione. Siano P, Q e R tre fbf.

- **Se v(P) = v(Q) allora:**
- **per ogni R, v(R[P/A]) = v(R[Q/A])**

*Per induzione strutturale sulla struttura di **R***

Soluzione

- **Caso base**: *Simboli atomici, F e T*.
  1. R è un simbolo atomico:
     - Caso R ≠ A,
       - per def. di sostituzione R[P/A] = R = R[Q/A]
     - Caso R = A,
       - per def. di sostituzione: R[P/A] = P e R[Q/A] = Q
       - per l'ipotesi della proposizione: v(P) = v(Q)
       - e quindi:

         v(R[P/A]) = v(P) = v(Q) = v(R[Q/A])

Soluzione

- **Caso base**:
  2. R = T:
     - Per def. di sostituzione (dato che A non è contenuto in T):
       - T[P/A] = T = T[Q/A]
  3. R = F
     - Per def. di sostituzione (dato che A non è contenuto in F):
       - F[P/A] = F = F[Q/A]

Soluzione

- **Caso induttivo Negazione** ~R:
  - Ipotesi induttiva: v(R[P/A]) = v(R[Q/A])
  - Tesi: v((~R)[P/A]) = v((~R)[Q/A])
  - Quindi:

  v((~R)[P/A]) = v(~(R[P/A])) = 1- v(R[P/A]) = 1- v(R[Q/A]) = v(~(R[Q/A])) = v((~R)[Q/A])

Soluzione

- **Caso induttivo Cong.** $(R_1 \wedge R_2)$:
  - Ipotesi induttiva: $v(R_1[P/A]) = v(R_1[Q/A])$ e $v(R_2[P/A]) = v(R_2[Q/A])$
  - Tesi: $v((R_1 \wedge R_2)[P/A]) = v((R_1 \wedge R_2)[Q/A])$
  - Quindi:

  $v((R_1 \wedge R_2)[P/A]) = v((R_1[P/A] \wedge R_2[P/A])) =$
  $= \min(v(R_1[P/A]), v(R_2[P/A])) =$
  $= \min(v(R_1[Q/A]), v(R_2[Q/A])) =$
  $= v((R_1[Q/A] \wedge R_2[Q/A])) = v((R_1 \wedge R_2)[Q/A])$

Es 1.4 libro pag. 32

- Il seguente insieme di formule è soddisfacibile?
  - $A_1 \vee A_2$,
  - $\sim A_2 \vee \sim A_3$,
  - $A_3 \vee A_4$,
  - $\sim A_4 \vee A_5$
- Sol: bisogna trovare un'interpretazione che sia modello di tutte le formule.

Es 1.4 soluzione

A_1	A_2	$A_1 \vee A_2$
0	0	0
0	**1**	**1**
1	0	1
1	1	1

A_2	A_3	$\sim A_2 \vee \sim A_3$
0	0	1
0	1	1
1	**0**	**1**
1	1	0

A_3	A_4	$A_3 \vee A_4$
0	0	0
0	**1**	**1**
1	0	1
1	1	1

A_4	A_5	$\sim A_4 \vee A_5$
0	0	1
0	1	1
1	0	0
1	**1**	**1**

Es 1.4 soluzione

int	A_1	A_2	A_3	A_4	A_5
v_1	**0**	**1**	**0**	**1**	**1**
v_2	1	0	0	1	1
v_3	1	0	1	0	0
v_4	1	0	1	0	1
v_5	1	0	1	1	1
v_6	1	1	0	1	1

Es 1.7 libro pag. 32

- **Stabilire se le seguenti affermazioni sono equivalenti:**
  - **P $\models$ Q**
  - **Se $\models$ P allora $\models$ Q**
- Dobbiamo dim che
  - P $\models$ Q **implica** se $\models$ Q allora $\models$ Q
  - Se $\models$ P allora $\models$ B **implica** P $\models$ Q

Es 1.7 soluzione

- P ⊨ Q **implica** se ⊨ P allora ⊨ Q
  - Sia P una tautologia dobbiamo dim che Q è una tautologia.
  - Sappiamo che P ⊨ Q, ovvero tutti i modelli di P sono anche modelli di Q.
  - Se P è una tautologia tutte le interpretazioni sono modelli di P e quindi tutte le interpretazioni sono modelli di Q, quindi Q è una tautologia.

Es 1.7 soluzione

- Se ⊨ P allora ⊨ Q **implica** P ⊨ Q
  - Dobbiamo dim che ogni modello di P è anche un modello di Q.
  - Nel caso che P sia una tautologia per ipotesi sappiamo che anche Q è una tautologia. E quindi l'affermazione è vera
  - Supponiamo che P non sia una tautologia, l'ipotesi non ci aiuta.
  - Può essere falsa?

Es 1.7 soluzione

- Se $\models P$ allora $\models Q$ **implica** $P \models Q$
  - Controesempio:
  - $P = A$, $Q = A \wedge B$.
  - P non è una tautologia quindi ($\models P$) è falso e quindi (Se $\models P$ allora $\models Q$) è vero.
  - Ma $A \models A \wedge B$ è falso perché esiste un modello per A che non è modello di $A \wedge B$: $v(A) = 1$ $v(B) = 0$!!!

Es 1.8.1 libro pag. 32

- Provare che:
  - $F \vee P \equiv P$
- Dimostriamo che:
  - Tutti i modelli di $F \vee P$ sono anche modelli di P
  - Tutti i modelli di P sono anche modelli di $F \vee P$.

Es 1.8.1 soluzione

- Tutti i modelli di $F \vee P$ sono anche modelli di P, sia v un modello di $F \vee P$:
  - $1 = v(F \vee P) = \max(v(F), v(P)) = \max(0, v(P)) = v(P)$
- Tutti i modelli di P sono anche modelli di $F \vee P$, sia v un modello di P:
  - $v(F \vee P) = \max(v(F), v(P)) = \max(0, v(P)) = \max(0,1) = 1$

Es 1.10 libro pag. 33

- Sia Δ il connettivo definito dalla seguente tavola di verità.
- Esprimerlo in funzione di $\{\sim, \vee\}$.

A	B	A Δ B
0	0	1
0	1	1
1	0	0
1	1	0

Es 1.10 soluzione

A	B	A Δ B
0	0	1
0	1	1
1	0	0
1	1	0

- FND: A Δ B ≡ (~A ∧ ~B) ∨ (~A ∧ B)
- **Oss:** (P ∧ Q) ≡ ~ (~P ∨ ~Q)
- A Δ B ≡ ~(A ∨ B) ∨ ~(A ∨ ~ B)

Es 1.11 libro pag. 33

- Provare che l'insieme {⇒, F} è funzionalmente completo sapendo che lo è {∧, ~}.
- E' necessario dimostrare che ∧ e ~ possono essere scritti in funzione di:
  - {⇒, F}

Es 1.11 soluzione

- $\sim A \equiv \sim A \vee F \equiv A \Rightarrow F$
- $A \wedge B \equiv \sim(\sim A \vee \sim B) \equiv \sim(A \Rightarrow \sim B) \equiv$
  $\equiv \sim(A \Rightarrow (B \Rightarrow F)) \equiv (A \Rightarrow (B \Rightarrow F)) \Rightarrow F$
- $\mathbf{\sim A \equiv A \Rightarrow F}$
- $\mathbf{A \wedge B \equiv (A \Rightarrow (B \Rightarrow F)) \Rightarrow F}$

Es 1.12 libro pag. 33

- Sia # il connettivo ternario espresso dalla seguente funzione di verità
- $f(a_1, a_2, a_3) = 1 \Leftrightarrow a_1 + a_2 + a_3 \geq 2$
- Esprimerlo in funzione di $\{\wedge, \sim\}$.

Es 1.12 soluzione

- $f(a_1, a_2, a_3) = 1 \Leftrightarrow a_1 + a_2 + a_3 \geq 2$

A_1	A_2	A_3	$f(A_1, A_2, A_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Es 1.12 soluzione

- FNC: $(A_1 \vee A_2 \vee A_3) \wedge (A_1 \vee A_2 \vee \sim A_3) \wedge (A_1 \vee \sim A_2 \vee A_3) \wedge (\sim A_1 \vee A_2 \vee A_3)$

- $\sim(\sim A_1 \wedge \sim A_2 \wedge \sim A_3) \wedge \sim(\sim A_1 \wedge \sim A_2 \wedge A_3) \wedge \sim(\sim A_1 \wedge A_2 \wedge \sim A_3) \wedge \sim(A_1 \wedge \sim A_2 \wedge \sim A_3)$

Es 1.13 libro pag. 33

- Sia NOR il connettivo definito dalla seguente tavola di verità.
- Dimostrare che per ogni connettivo *n*-ario, esiste una formula associata ad esso contenente il solo connettivo NOR

A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

Es 1.13 soluzione

A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

- Oss: ~A = A NOR A
- Oss: A NOR B = ~(A $\vee$ B) e quindi:
  - A $\vee$ B = ~(A NOR B)
    = (A NOR B) NOR (A NOR B)

Es 1.14 libro pag. 33

- Mostrare che l'insieme di connettivi
  - $\{\wedge, \Leftrightarrow, \oplus\}$
- È funzionalmente completo
- e che nessuno dei suoi sottoinsiemi propri lo è.

Es 1.14 soluzione

- **$\{\wedge, \Leftrightarrow, \oplus\}$ è funzionalmente completo**

- Il connettivo $\wedge$ può essere banalmente scritto in funzione di $\{\wedge, \Leftrightarrow, \oplus\}$
- Basta mostrare che anche il connettivo $\sim$ può essere scritto in funzione di $\{\wedge, \Leftrightarrow, \oplus\}$
  - $\sim A \equiv A \oplus T \equiv A \oplus (A \Leftrightarrow A)$

Es 1.14 soluzione

- {∧} non è funzionalmente completo
  - Basta mostrare che il ~ non può essere scritto in funzione di {∧}
  - $A \wedge A \equiv A$

Es 1.14 soluzione

- {⇔} non è funzionalmente completo
  - Basta mostrare che il ~ non può essere scritto in funzione di {⇔}
  - $A \Leftrightarrow A \equiv T$
  - $A \Leftrightarrow T \equiv A$
  - $T \Leftrightarrow A \equiv A$
  - $T \Leftrightarrow T \equiv T$

Es 1.14 soluzione

- {⊕} non è funzionalmente completo
  - Basta mostrare che il ~ non può essere scritto in funzione di {⊕}
  - $A \oplus A \equiv F$
  - $A \oplus F \equiv A$
  - $F \oplus A \equiv A$
  - $F \oplus F \equiv F$

Es 1.14 soluzione

- {∧, ⇔} non è funzionalmente completo
  - Basta mostrare che il ~ non può essere scritto in funzione di {∧, ⇔}
  - $A \Leftrightarrow A \equiv T$
  - $A \Leftrightarrow T \equiv A$
  - $T \Leftrightarrow A \equiv A$
  - $T \Leftrightarrow T \equiv T$
  - $A \wedge A \equiv A$
  - $A \wedge T \equiv A$
  - $T \wedge A \equiv A$
  - $T \wedge T \equiv T$

Es 1.14 soluzione

- $\{\wedge, \oplus\}$ non è funzionalmente completo
  - Basta mostrare che il ~ non può essere scritto in funzione di $\{\wedge, \oplus\}$
  - $A \oplus A \equiv F$
  - $A \oplus F \equiv A$
  - $F \oplus A \equiv A$
  - $F \oplus F \equiv F$
  - $A \wedge A \equiv A$
  - $A \wedge F \equiv F$
  - $F \wedge A \equiv F$
  - $F \wedge F \equiv F$

Es 1.14 soluzione

- $\{\Leftrightarrow, \oplus\}$ non è funzionalmente completo
  - Basta mostrare che il $\wedge$ non può essere scritto in funzione di $\{\Leftrightarrow, \oplus\}$

A	B	$A \wedge B$	$A \Leftrightarrow B$	$A \oplus B$
0	0	0	1	0
0	1	0	0	1
1	0	0	0	1
1	1	1	1	0

Es 1.18 libro pag. 34

- Trovare un criterio affinché
  - una formula in FNC sia una tautologia
  - una formula in FND sia una tautologia

Es 1.18 soluzione

- **Trovare un criterio affinché una formula in FNC sia una tautologia**:
  - $P = P_1 \wedge P_2 \wedge \dots \wedge P_n$
  - P_i è una disgiunzione di letterali:
    - $P_i = L_1 \vee L_2 \vee \dots \vee L_m$
- $P = P_1 \wedge P_2 \wedge \dots \wedge P_n \equiv T$
  - sse $P_1 \equiv P_2 \equiv \dots \equiv P_n \equiv T$
  - sse ogni P_i contiene un letterale e il suo negato
  - ovvero ogni P_i contiene $A_i \vee \sim A_i$
- Esempio:

  $P = (A \vee B \vee \sim A) \wedge (\sim B \vee B \vee \sim A) \wedge (A \vee C \vee \sim A \vee \sim D)$

Es 1.18 soluzione

- **Trovare un criterio affinché una formula in FND sia una tautologia**:
  - $P = P_1 \vee P_2 \vee \ldots \vee P_n$
  - P_i è una congiunzione di letterali:
    - $P_i = L_1 \wedge L_2 \wedge \ldots \wedge L_m$
- $P = P_1 \vee P_2 \vee \ldots \vee P_n = T$
  - sse la sua forma normale completa contiene tutte le possibili congiunzione di letterali
- Non è facile capirlo come per FNC: dobbiamo costruire la forma completa che ci costa $O(2^n)$.
- Esempio:

$P = (A) \wedge (\sim A \vee B) \wedge (\sim A \vee \sim B) \equiv$

$\equiv (A \vee B) \wedge (A \vee \sim B) \wedge (\sim A \vee B) \wedge (\sim A \vee \sim B)$ (FND compl.)

Es 1.21 libro pag. 34

- Si definisca il connettivo $\odot$ duale del connettivo di implicazione $\Rightarrow$, dandone la tabella di verità
- Si esprima $\odot$ in funzione di $\{\wedge, \sim\}$
- Si estenda la funzione $^{\perp}$ a formule composte anche dai connettivi $\Rightarrow$ e $\odot$,
- Si dimostri che per ogni P
  - $v(P^{\perp}) = 1 - v(P)$

Es 1.21 soluzione

- **Si definisca il connettivo ⊙ duale del connettivo di implicazione ⇒, dandone la tabella di verità**

A	B	A ⇒ B
0	0	1
0	1	1
1	0	0
1	1	1

A	B	A ⊙ B
0	0	0
0	1	1
1	0	0
1	1	0

Es 1.21 soluzione

- **Si esprima** ⊙ in funzione di {∧, ~}
- FND**:** ~A ∧ B
- Osservazioni:
  - ⊙ è il duale di ⇒**:** (A ⇒ B) = ~A ∨ B
  - per ogni interpretazione v e per ogni f.b.f. P e Q si avrà che :
    - v(P⊙Q) = min(1-v(P), v(Q))

Es 1.21 soluzione

- **Si estenda la funzione $^\perp$ a formule composte anche dai connettivi $\Rightarrow$ e $\odot$,**
- La funzione $^\perp$: FBF $\rightarrow$ FBF soddisfa:
  - $P^\perp = \sim P$ se P è una formula atomica
  - $(P \wedge Q)^\perp = P^\perp \vee Q^\perp$
  - $(P \vee Q)^\perp = P^\perp \wedge Q^\perp$
  - $(\sim P)^\perp = \sim P^\perp$
  - $\mathbf{(P \Rightarrow Q)^\perp = P^\perp \odot Q^\perp}$
  - $\mathbf{(P \odot Q)^\perp = P^\perp \Rightarrow Q^\perp}$

Es 1.21 soluzione

- Si dimostri che per ogni P
  - $\mathbf{v(P^\perp) = 1 - v(P)}$
- Per induzione strutturale:
- *Caso base*: P proposizione atomica.
  - Oss: $P^\perp = \sim P$
  - $v(P^\perp) = v(\sim P) = 1 - v(P)$

Es 1.21 soluzione

- *Caso induttivo*: negazione:
  - Da dim: $v((\sim P)^{\perp}) = 1 - v(\sim P)$
  - Oss: $(\sim P)^{\perp} = \sim P^{\perp}$
  - Ip. Ind: $v(P^{\perp}) = 1 - v(P)$
  - $v((\sim P)^{\perp}) = v(\sim P^{\perp}) = 1 - v(P^{\perp}) =$
    $1 - 1 - v(P) = 1 - v(\sim P)$

Es 1.21 soluzione

- *Caso induttivo* congiunzione:
  - Da dim: $v((P \wedge Q)^{\perp}) = 1 - v\ (P \wedge Q)$
  - Oss: $(P \wedge Q)^{\perp} = P^{\perp} \vee Q^{\perp}$
  - Ip. Ind: $v(P^{\perp}) = 1 - v(P)$
    e $v(Q^{\perp}) = 1 - v(Q)$
  - $v((P \wedge Q)^{\perp}) = v(P^{\perp} \vee Q^{\perp}) =$
    $= \max(v(P^{\perp}), v(Q^{\perp}))$
    $= \max(1 - v(P), 1 - v(Q)) =$
    $= 1 - \min(v(P), v(Q)) = 1 - v\ (P \wedge Q)$

Es 1.21 soluzione

- *Caso induttivo* implicazione:
  - Da dim: $v((P \Rightarrow Q)^{\perp}) = 1 - v\,(P \Rightarrow Q)$
  - Oss: $(P \Rightarrow Q)^{\perp} = P^{\perp} \odot Q^{\perp}$
  - Ip. Ind: $v(P^{\perp}) = 1 - v(P)$
    e $v(Q^{\perp}) = 1 - v(Q)$
  - $v\,((P \Rightarrow Q)^{\perp}) = v(P^{\perp} \odot Q^{\perp}) =$
    $=\min(1 - v(P^{\perp}), v(Q^{\perp}))$
    $=\min(1 - 1 - v(P), 1 - v(Q))=$
    $=1 - \max(1 - v(P), v(Q))=$
    $1 - v\,(P \Rightarrow Q)$

Fondamenti di Logica Matematica
Lezione 8

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Sistemi deduttivi
  - Introduzione
  - Proprietà
- Deduzione naturale
  - Regole di inferenza
- Sistemi assiomatici (cenni)
- Calcolo dei sequenti
  - Assiomi e regole di inferenza

Dimostrazioni

- Se $A \Rightarrow B$ e $A \Rightarrow (B \Rightarrow C)$ allora $A \Rightarrow C$
- Possiamo dimostrare che è una tautologia costruendo la tabella di verità di:
  - $((A \Rightarrow B) \wedge (A \Rightarrow (B \Rightarrow C))) \Rightarrow (A \Rightarrow C)$
- Possiamo anche darne una dimostrazione diretta.

Dimostrazioni

- Ipotesi:
  - $A \Rightarrow B$
  - $A \Rightarrow (B \Rightarrow C)$
- Tesi:
  - $A \Rightarrow C$ (se vale A deve valere anche C)
- Dimostrazione:
  - A e $(A \Rightarrow B)$: se vale A vale anche **B**
  - A e $(A \Rightarrow (B \Rightarrow C))$: se vale A allora vale anche **$(B \Rightarrow C)$**
  - B e $(B \Rightarrow C)$: se vale B allora vale anche **C**

Sistemi deduttivi

- **Scopo:** definire dei sistemi di calcolo (*sistemi deduttivi*) per lo sviluppo di dimostrazioni.
- **Dimostrazione in un sistema deduttivo**: è una sequenza di passi elementari che partendo dalle *premesse* consenta di ottenere la *conclusione*.

Proprietà dei sistemi deduttivi

- **Correttezza:** il sistema deduttivo non inferisce proposizioni non valide.
- **Completezza:** ogni formula valida è dimostrabile con il sistema deduttivo.
- Un sistema deduttivo è *corretto* e *completo* se inferisce tutte e sole le proposizioni valide.

Notazione

- $P_1, P_2, \ldots, P_n \vdash Q$
- Nel sistema di calcolo siamo in grado di fornire una dimostrazione della formula B partendo dalle premesse $P_1, P_2, \ldots, P_n$.
- Useremo anche: $\Gamma \vdash Q$.
- Oss: Γ può essere vuoto.

Proprietà

- **Congiunzione**:
  - $P \wedge Q \vdash P$ ($\wedge$e.1)
  - $P \wedge Q \vdash Q$ ($\wedge$e.2)
  - $P, Q \vdash P \wedge Q$ ($\wedge$i)
- **Disgiunzione**:
  - $P \vdash P \vee Q$ ($\vee$i.1)
  - $Q \vdash P \vee Q$ ($\vee$i.2)
  - Se $P \vdash R$ e $Q \vdash R$ allora $P \vee Q \vdash R$ ($\vee$e)

Proprietà

- **Implicazione**:
  - P, P $\Rightarrow$ Q $\vdash$ Q (⇒e)
  - Se P $\vdash$ Q allora $\vdash$ P $\Rightarrow$ Q (⇒i)
- **Negazione:**
  - Se P $\vdash$ **F** allora $\vdash$ ~P (~i)
  - P, ~P $\vdash$ **F** (~e)
- **Falsità F:**
  - **F** $\vdash$ P (⊥e)

Proprietà

- **Reductio Ad Absurdum**:
  - Se ~P $\vdash$ **F** allora $\vdash$ P (RAA)
- **Taglio:**
  - Se $\Gamma \vdash$ P e Γ,P$\vdash$Q allora $\Gamma \vdash$Q (taglio)

Sistemi deduttivi

- *Deduzione naturale*
- Sistemi assiomatici
- *Calcolo dei sequenti*
- Tableaux

Deduzione naturale

- E' stata introdotta da Gentzen
- Notazione:

$$\frac{\text{Premesse}}{\text{Conclusioni}}$$

- Due tipi di regole:
  - Regole elementari
  - Regole condizionali (premesse sussidiarie)

Regole: congiunzione

- $\dfrac{P \wedge Q}{P}$ ($\wedge$e.1)
- $\dfrac{P \wedge Q}{Q}$ ($\wedge$e.2)
- $\dfrac{P \quad Q}{P \wedge Q}$ ($\wedge$i)

Regole: disgiunzione

- $\dfrac{P}{P \vee Q}$ ($\vee$i.1)
- $\dfrac{Q}{P \vee Q}$ ($\vee$i.2)
- $\dfrac{P \vee Q \quad \begin{array}{c}[P]\\ R\end{array} \quad \begin{array}{c}[Q]\\ R\end{array}}{R}$ ($\vee$e)

Regole: Implicazione

- $\dfrac{P \quad P \Rightarrow Q}{Q}$ (⇒e)
- $\dfrac{\begin{matrix}[P] \\ Q\end{matrix}}{P \Rightarrow Q}$ (⇒i)

Regole: Negazione e ⊥

- $\dfrac{P \quad \sim P}{\mathbf{F}}$ (~e)
- $\dfrac{\begin{matrix}[P] \\ \mathbf{F}\end{matrix}}{\sim P}$ (~i)
- $\dfrac{\mathbf{F}}{P}$ (⊥e)

Regole: RAA

- $\dfrac{\begin{matrix}[\sim P] \\ \mathbf{F}\end{matrix}}{P}$ (RAA)

Esempio di deduzione

- $A \vdash_{DN} A \wedge (B \vee A)$

Si arriva alle ipotesi

$$\dfrac{A \qquad \dfrac{A}{B \vee A}\ (\vee i.2)}{A \wedge (B \vee A)}\ (\wedge i)$$

connettivo più esterno

La deduzione si calcola dal basso verso l'alto

Si parte dalla formula da dedurre

Esempio di deduzione

- $\vdash_{DN} \sim\sim A \Rightarrow A$

Ipotesi derivanti dalle regole

$$\dfrac{\dfrac{\dfrac{[\sim A] \qquad [\sim\sim A]}{F}\ (\sim e)}{A}\ (RAA)\ \text{canc. } [\sim A]}{\sim\sim A \Rightarrow A}\ (\Rightarrow i)\ \text{canc. } [\sim\sim A]$$

connettivo più esterno

Sistemi assiomatici

- Hanno un'unica regola di inferenza:
  - Modus ponens: $P, P \Rightarrow Q \vdash Q$
- Si introducono alcuni assiomi al posto delle altre regole.

Calcolo dei sequenti

- Introdotto da Gentzen.
- Differente dai due sistemi precedenti:
  - Il calcolo dei sequenti lavora non su fbf ma su asserzioni di derivabilità del tipo $\Gamma \vdash \Delta$.
  - I connettivi logici possono essere solo introdotti e mai eliminati.

Notazione

- $\Gamma \vdash \Delta$ è un **sequente**.
- Γ e Δ insiemi di fbf.
- Le formule di Γ vanno pensate come unite dal connettivo $\wedge$.
- Le formule di Δ vanno pensate unite dal connettivo $\vee$.
- Es: $P_1, P_2, \ldots, P_n \vdash Q_1, Q_2, \ldots, Q_m$ deve essere pesato come:
  - $P_1 \wedge P_2 \wedge \ldots \wedge P_n \vdash Q_1 \vee Q_2 \vee \ldots \vee Q_m$

Notazione: regole

- Una **regola di inferenza** ha la seguente forma:

- $$\frac{\Gamma_1 \vdash \Delta_1}{\Gamma_2 \vdash \Delta_2}$$

- $$\frac{\Gamma_1 \vdash \Delta_1 \quad \Gamma_2 \vdash \Delta_2}{\Gamma_3 \vdash \Delta_3}$$

Assiomi e regole

- **Assiomi:** i sequenti della forma
  - $A \vdash A$ (Ax)
- Regole di tre tipi:
  - Strutturali.
  - Taglio.
  - Logiche.

Regole strutturali

- $\dfrac{\Gamma, P, Q, \Gamma' \vdash \Delta}{\Gamma, Q, P, \Gamma' \vdash \Delta}$ (perm-l)
- $\dfrac{\Gamma \vdash \Delta, P, Q, \Delta'}{\Gamma \vdash \Delta, Q, P, \Delta'}$ (perm-r)

Regole strutturali

- $\dfrac{\Gamma \vdash \Delta}{\Gamma, P \vdash \Delta}$ (indeb-l)
- $\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, P}$ (indeb-r)

Taglio

- $$\frac{\Gamma \vdash P, \Delta \qquad \Gamma', P \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'} \qquad \text{(taglio)}$$

Regole logiche: ∧

- $$\frac{\Gamma, P, Q \vdash \Delta}{\Gamma, P \wedge Q \vdash \Delta} \qquad (\wedge\text{-l})$$

- $$\frac{\Gamma \vdash P, \Delta \qquad \Gamma \vdash Q, \Delta}{\Gamma \vdash P \wedge Q, \Delta} \qquad (\wedge\text{-r})$$

Regole logiche: $\vee$

- $\dfrac{\Gamma, P \vdash \Delta \quad \Gamma, Q \vdash \Delta}{\Gamma, P \vee Q \vdash \Delta}$ $(\vee\text{-l})$

- $\dfrac{\Gamma \vdash P, Q, \Delta}{\Gamma \vdash P \vee Q, \Delta}$ $(\vee\text{-r})$

Regole logiche: $\sim$

- $\dfrac{\Gamma \vdash P, \Delta}{\Gamma, \sim P \vdash \Delta}$ $(\sim\text{-l})$

- $\dfrac{\Gamma, P \vdash \Delta}{\Gamma \vdash \sim P, \Delta}$ $(\sim\text{-r})$

Regole logiche: $\Rightarrow$

- $\dfrac{\Gamma \vdash P\,,\Delta \quad \Gamma, Q \vdash \Delta}{\Gamma, P \Rightarrow Q \vdash \Delta}$ ($\Rightarrow$-l)

- $\dfrac{\Gamma, P \vdash Q, \Delta}{\Gamma \vdash P \Rightarrow Q, \Delta}$ ($\Rightarrow$-r)

Calcolo

1. Scrivere l'**intero** sequente in basso
2. Identificare i connettivi più esterni
   - scegliere (a caso) uno di essi
   - utilizzare la regola corrispondente al connettivo:
     - se il connettivo è a destra di $\vdash$ si usa la regola r
     - se il connettivo è a sinistra di $\vdash$ si usa la regola l
     - tracciare una riga sopra al sequente e implementare la regola
3. Continuare con i sequenti generati usando la regola 2. finché non si arriva ad un assioma (es. $A \vdash A$).

Esempio

- Dimostrare che $C \vdash \sim(A \vee B) \Rightarrow \sim(B \vee A)$

Si arriva agli assiomi

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \vdash B}{C, B \vdash A, B}\,(\text{indeb-r}) \qquad \dfrac{A \vdash A}{C, A \vdash A, B}\,(\text{indeb-r})}{C, (B \vee A) \vdash A, B}\,(\vee\text{-l})}{C, (B \vee A) \vdash (A \vee B)}\,(\vee\text{-r})}{C \vdash (A \vee B), \sim(B \vee A)}\,(\sim\text{-r})}{C, \sim(A \vee B) \vdash \sim(B \vee A)}\,(\sim\text{-l})}{C \vdash \sim(A \vee B) \Rightarrow \sim(B \vee A)}\,(\Rightarrow\text{-r})
$$

connettivi esterni: ~ a destra e ~ a sinistra: ne scelgo uno

La deduzione si calcola dal basso verso l'alto

connettivo più esterno

Si parte dall'intero sequente da dimostrare

Fondamenti di Logica Matematica
Lezione 9

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Calcolo dei sequenti
  - Proprietà
  - Esempi
  - Invertibilità
  - Correttezza
  - Completezza
  - Ricerca di contromodelli

Regola del taglio

- La regola del taglio è ridondante
- $$\frac{\Gamma \vdash P, \Delta \qquad \Gamma', P \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'} \quad \text{(taglio)}$$
- Teorema Gentzen's Hauptsatz:
  - Ogni dimostrazione che usa la regola del taglio può essere riscritta in una che non la usa.
- Possiamo non usare la regola

Proprietà

- Proprietà della sottoformula:
  - Nel calcolo dei sequenti senza regola del taglio.
  - Una dimostrazione di $\Gamma \vdash \Delta$ contiene solo sequenti le cui sottoformule sono sottoformule di Γ, Δ.
  - Oss: le regole sono solo di introduzione e mai di eliminazione!

Esempio

- Dimostrare che $\vdash (A \wedge B) \Rightarrow (B \wedge A)$

$$\dfrac{\dfrac{\dfrac{\dfrac{B \vdash B}{A, B \vdash B} \qquad \dfrac{A \vdash A}{A, B \vdash A}}{A, B \vdash (B \wedge A)}}{(A \wedge B) \vdash (B \wedge A)}}{\vdash (A \wedge B) \Rightarrow (B \wedge A)}$$

Esempio

- Dimostrare che $\sim(A \vee B) \vdash \sim A \wedge \sim B$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{\vdash A, \sim A}}{\vdash A, B, \sim A} \qquad \dfrac{\dfrac{B \vdash B}{\vdash B, \sim B}}{\vdash A, B, \sim B}}{\vdash A, B, \sim A \wedge \sim B}}{\vdash A \vee B, \sim A \wedge \sim B}}{\sim(A \vee B) \vdash \sim A \wedge \sim B}$$

Invertibilità

- Le formule sono invertibili:
  - se esiste una dimostrazione, questa si può trovare partendo da una qualsiasi scelta della formula principale su cui lavorare.
- Sistema di prova costruttivo.
- Dimostrazione **automatica**.

Sintassi e semantica

Sintassi (calcolo)	Semantica
$\Gamma \vdash_S P$ derivabilità	$\Gamma \vDash P$ cons. semantica
$\vdash_S P$ teorema	$\vDash P$ tautologia
$\Gamma \nvdash_S \bot$ consistenza	$\Gamma \nvDash \bot$ soddisfacibilità
$\Gamma \vdash_S \bot$ inconsistenza	$\Gamma \vDash \bot$ insoddisfacibilità

Notazione

- Una f.b.f. P è **conseguenza semantica** di un insieme Γ di f.b.f. e si scrive $\Gamma \models P$, se ogni modello di Γ è un modello per P.
- Siano Γ e Δ degli insiemi di f.b.f., Δ **è soddisfatto** in Γ ($\Gamma \models \Delta$) sse
  - per ogni v se v(P)=1 per ogni P in Γ
  - allora esiste un Q in Δ t.c. v(Q)=1

Correttezza e completezza

- **Teorema di correttezza**:
  - se $\Gamma \vdash_S \Delta$ allora $\Gamma \models \Delta$

  (se Δ è derivabile da Γ allora Δ è soddisfatto in Γ)
- **Teorema di completezza**:
  - se $\Gamma \models \Delta$ allora $\Gamma \vdash_S \Delta$

  (se Δ è soddisfatto in Γ allora Δ è derivabile da Γ)

Correttezza

- **Teorema di correttezza**:
  - se $\Gamma \vdash_S \Delta$ allora $\Gamma \models \Delta$.
- Per induzione "sulle regole":
  - si ipotizza che proprietà sia vera sulle premesse della regola
  - si dimostra che è vera anche sulla conclusione
- Proprietà da dimostrare:
  - se v(P)=1 per ogni P in Γ allora esiste almeno un Q in Δ tale che v(Q)=1.
- **Caso base**: $A \vdash A$, la tesi è ovvia poiché $A \models A$ è sempre vera.

Correttezza

- **Caso induttivo: negazione** (~-l)
  - Regola $\dfrac{\Gamma \vdash P, \Delta}{\Gamma, \sim P \vdash \Delta}$ (~-l)
  - Ipotesi induttiva: per $\Gamma \vdash P, \Delta$ vale il th.
  - Tesi: vale anche per $\Gamma, \sim P \vdash \Delta$
  - Sia v t.c. rende vere tutte le prop. di Γ,~P quindi v(P) =0. Per l'ipotesi induttiva allora deve esistere almeno un Q in Δ tale che v(Q) =1

Correttezza

- **Caso induttivo: negazione** (~-r)
  - Regola $\dfrac{\Gamma, P \vdash \Delta}{\Gamma \vdash \sim P, \Delta}$ (~-r)
  - Ipotesi induttiva: per $\Gamma, P \vdash \Delta$ vale il th.
  - Tesi: vale anche per $\Gamma \vdash \sim P, \Delta$
  - Sia v t.c. rende vere tutte le prop di Γ.
    - Caso 1: v(~P) =1 ovvio v rende vera almeno una proposizione in ~P, Δ
    - Caso 2: v(~P) =0 ovvero v(P) =1 e quindi per ipotesi induttiva v rende vera almeno una proposizione in Δ

Correttezza

- **Caso induttivo: congiunzione** ($\wedge$-l)
  - Regola $\dfrac{\Gamma, P, Q \vdash \Delta}{\Gamma, P \wedge Q \vdash \Delta}$ ($\wedge$-l)
  - Ipotesi induttiva: per $\Gamma, P, Q \vdash \Delta$ vale il th.
  - Tesi: vale anche per $\Gamma, P \wedge Q \vdash \Delta$
  - Sia v t.c. rende vere tutte le prop di $\Gamma, P \wedge Q$, quindi (per def di $\wedge$) rende vere tutte le prop di Γ, P, Q e quindi per ipotesi induttiva rende vera almeno una proposizione in Δ.

Correttezza

- **Caso induttivo: congiunzione** ($\wedge$-r)
  - Regola $\dfrac{\Gamma \vdash P, \Delta \quad \Gamma \vdash Q, \Delta}{\Gamma \vdash P \wedge Q, \Delta}$ ($\wedge$-r)
  - Ipotesi induttive: per $\Gamma \vdash P, \Delta$ e $\Gamma \vdash Q, \Delta$ vale il th.
  - Tesi: vale anche per $\Gamma \vdash P \wedge Q, \Delta$
  - Sia v t.c. rende vere tutte le prop di Γ.
    - Caso 1: $v(P \wedge Q)=1$, allora la tesi è vera
    - Caso 2: $v(P \wedge Q)=0$ che vuol dire $v(P)=0$ o $v(Q)=0$.
      - Se $v(P)=0$ per l'ipotesi $\Gamma \vdash P, \Delta$ abbiamo che v deve rendere vera almeno una prop in Δ
      - Se $v(Q)=0$ per l'ipotesi $\Gamma \vdash Q, \Delta$ abbiamo che v deve rendere vera almeno una prop in Δ

Completezza finita

- **Teorema di completezza finita**:
  - Sia Γ un insieme finito di fbf, se $\Gamma \models \Delta$ allora $\Gamma \vdash_S \Delta$.
- Dimostrazione:
  - Supponiamo che $\Gamma \models \Delta$
  - Costruiamo una dimostrazione con il calcolo dei sequenti per $\Gamma \vdash_S \Delta$:
  - Poiché ogni premessa contiene meno connettivi della conclusione: il calcolo termina
  - Quindi con un numero finito di passi arriviamo a delle foglie del tipo
    - $\Gamma' \vdash_S \Delta'$
    - contenenti solo var. atomiche.

Completezza finita

- Dimostrazione:
  - Le regole preservano la validità.
  - Dato che per ipotesi $\Gamma \vDash \Delta$ allora tutte le foglie devono essere valide
    - Γ' e Δ' sono insiemi di formule atomiche
    - un sequente composto di formule atomiche per essere valido deve contenere una formula atomica A sia in Γ' che in Δ'
    - quindi con la regola di indebolimento otteniamo un assioma $A \vdash_S A$
    - il seguente è dimostrato.

Completezza forte

- **Teorema di completezza forte**:
  - Se $\Gamma \vDash \Delta$ allora $\Gamma \vdash_S \Delta$.
- Dimostrazione:
  - Per il secondo Corollario al Teorema di compattezza: se $\Gamma \vDash \Delta$ allora esiste un sottoinsieme finito Γ' di Γ t.c. $\Gamma' \vDash \Delta$.
  - Per il teorema di completezza finita: se $\Gamma' \vDash \Delta$ allora $\Gamma' \vdash_S \Delta$.
  - Poiché Γ' è un sottoinsieme di Γ: se $\Gamma' \vdash_S \Delta$ a maggior ragione $\Gamma \vdash_S \Delta$.

Ricerca di contromodelli

- Con il calcolo dei sequenti possiamo anche dimostrare che una conseguenza semantica è falsa.
- Per dimostrare che $\Gamma \models \Delta$ è falso basta dimostrare che esiste un modello di Γ che non sia modello per nessuna proposizione in Δ.
- Tale interpretazione è detta **contromodello** di $\Gamma \models \Delta$

Ricerca di contromodelli

- Se $\Gamma \models \Delta$ è falso, possiamo utilizzare il calcolo dei sequenti per costruire un contromodello:
  1. sia $A_1A_2...A_n \vdash_S B_1B_2...B_m$ il sequente che non può essere ulteriormente trasformato in un assioma ($A_i \neq B_j\ \forall i,j$)
  2. per rendere falso il sequente basta prendere un'interpretazione che:
     - assegna il valore vero a tutte le lettere A_i
     - assegna il valore falso a tutte le lettere B_i

Esempio

- Dimostrare che $(A \vee B) \vDash (B \wedge A)$ è falso.

Non sono assiomi!

$$\dfrac{\dfrac{A \vdash B \qquad B \vdash B}{(A \vee B) \vdash B} \qquad \dfrac{B \vdash A \qquad A \vdash A}{(A \vee B) \vdash A}}{(A \vee B) \vdash (B \wedge A)}$$

Esempio

- Per dimostrare che $(A \vee B) \vDash (B \wedge A)$ è falso basta costruire un contromodello.
- Lo costruisco da $A \vdash B$:
  - v(A)=1
  - v(B)=0
- Verifico:
  - $v(A \vee B) = 1$ v è modello per $A \vee B$
  - $v(B \wedge A) = 0$ v non è modello per $A \wedge B$

Fondamenti di Logica Matematica

Esercitazione 10

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Es 2.2.1 libro pag. 75

- **Dimostrare, con la deduzione naturale e con il calcolo dei sequenti, che la seguente proposizione è derivabile:**
- $((A \wedge B) \wedge C) \Rightarrow (A \wedge (B \wedge C))$

Es 2.5.1 e 2.5.3 pag. 75

- **Dimostrare, con la deduzione naturale e con il calcolo dei sequenti, che:**

- $\vdash (A \Rightarrow \sim A) \Rightarrow \sim A$
- $\vdash (((A \Rightarrow B) \Rightarrow A) \Rightarrow A)$

Es 2.9 libro pag. 76

- **Si considerino le seguenti proposizioni:**
  - **(a) Se Franco è un elettricista, allora Giorgio è un idraulico o Elena è un'insegnante.**
  - **(b) Se Giorgio è un idraulico , allora Lucia non fa la commessa o Elena è un'insegnante.**
  - **(c) Se Lucia fa la commessa, allora Franco è un elettricista.**
  - **(d) Lucia fa la commessa.**
  - **(e) Elena è un'insegnante.**
- **Dimostrare, utilizzando la deduzione naturale e il calcolo dei sequenti, che:**
  - **a,b,c,d $\vdash$e**

Esercizio

- **Decidere, con il calcolo dei sequenti, se** $A \Rightarrow B, \sim A \vDash \sim B$ **è vero.**
- **Nel caso il sequente** $A \Rightarrow B, \sim A \vdash \sim B$ **non sia derivabile, fornire un contromodello (ovvero una interpretazione che lo rende falso).**

Es 2.11 libro pag. 77

- **Definire le regole della deduzione naturale per il connettivo** $\Leftrightarrow$**.**
- **Provare con la DN che:**
  - $\vdash (A \vee A) \Leftrightarrow A$
  - $\vdash (A \wedge A) \Leftrightarrow A$

Fondamenti di Logica Matematica
Lezione 11

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Logica dei predicati o del primo ordine:
  - Alfabeto
  - Sintassi
  - Semantica

Calcolo proposizionale

- È poco espressivo.
- Non è possibile gestire la nozione di *generalità*.
- Esempi:
  - P è vera per tutti gli oggetti di un dominio
  - Esiste almeno un oggetto che gode della proprietà P.

Quantificatori

- Quantificatore *universale* $\forall$:
  - Tutti i numeri interi sono razionali
- Quantificatore *esistenziale* $\exists$:
  - Esiste un razionale che non è un intero

Logica dei predicati

- **Quantificatori**: esistenziale e universale.
- **Predicati**: esprimono proprietà e relazioni su insiemi di oggetti del dominio.
  - Es: 5 è un numero intero
  - Es: x è un numero intero

Logica dei predicati

- **Termini**: oggetti del dominio. Possono essere:
  - Costanti. Es: 5.
  - Variabili. Es: x.
- **Funzioni**: permettono di definire nuovi oggetti in termini di quelli presenti.
  - Es: x + 1

Notazione

- x,y,z: variabili
- a,b,c: costanti
- f,g,h: funzioni
- A,B,C: predicati
- P,Q: formule della logica predicativa
- $\forall$, $\exists$: quantificatori

Formalizzazione

- "x +1 è un numero pari e 2 è un intero"
- $A(f(x,1)) \wedge B(2)$
- Con:
  - $f(x,y) = x + y$
  - $A(x)$ = x è un numero pari
  - $B(x)$ = x è un intero

Alfabeto

- Un insieme di simboli di costante.
- Un insieme (VAR) di simboli di variabile.
- Un insieme di simboli di funzione.
- Un insieme di simboli di predicato.
- Connettivi: $\sim$, $\wedge$, $\vee$, $\Rightarrow$, F, T.
- Quantificatori: $\forall$, $\exists$.
- Simboli ausiliari: "(" , ")".

Insieme dei termini TER

- TER è l'insieme così definito:
  - Ogni costante appartiene a TER.
  - Ogni variabile appartiene a TER.
  - Se $t_1, t_2, \ldots, t_n, \in$ TER e f è un simbolo di funzione n-ario, allora:
    - $f(t_1, t_2, \ldots, t_n) \in$ TER.
  - Niente altro appartiene a TER.

Formule ben formate

- L'insieme **FBF** contenente le fbf è così definito:
  - F e T sono f.b.f.,
  - Se $t_1, t_2, \ldots, t_n, \in$ TER e *A* è un simbolo di predicato *n*-ario, allora:
    - $A(t_1, t_2, \ldots, t_n) \in$ FBF.
  - se P è una f.b.f. anche (~P) è una f.b.f.,
  - se P e Q sono f.b.f. anche $(P \wedge Q)$, $(P \vee Q)$, $(P \Rightarrow Q)$, sono f.b.f,
  - se P è una f.b.f. anche $((\forall x)P)$ e $((\exists x)P)$ sono f.b.f,
  - niente altro è una f.b.f.

Esempi

- Ogni numero naturale è un intero
- $((\forall x)\ (A(x) \Rightarrow B(x)))$
- Esiste un numero che non è naturale
- $((\exists x)\ ({\sim}A(x)))$
- Per ogni intero x esiste un numero y tale che x<y
- $((\forall x)((\exists y)\ C(x,y)))$

Parentesi

- Precedenza tra i connettivi:
  - $\forall = \exists = \sim > \wedge > \vee > \Rightarrow$
- Possiamo eliminare alcune parentesi.
- Es: $((\forall x)\ (A(x) \Rightarrow B(x)))$
  - $\forall x\ (A(x) \Rightarrow B(x))$
- Es: $((\exists x)\ (\sim A(x)))$
  - $\exists x \sim A(x)$

Sottoformule

- Sia P una f.b.f. le **sottoformule di** P, **Stfm(P),** sono così definite:
- se P è F o T o $A(t_1, t_2, \ldots, t_n)$
  - Stfm(P) = {P}.
- se P è $\sim Q$,
  - Stfm(P)={P} $\cup$ Stfm(Q),
- se P è $P_1 \wedge P_2$, $P_1 \vee P_2$ o $P_1 \Rightarrow P_2$,
  - Stfm(P)={P} $\cup$ Stfm(P_1) $\cup$ Stfm(P_2).
- se P è $(\forall x Q)$ o $(\exists x Q)$
  - Stfm(P)={P} $\cup$ Stfm(Q)

Esempio

- $P = \forall x\ (A(x) \Rightarrow B(x)) \wedge \exists x \sim A(f(x,y))$
- Sottoformule:
  - P
  - $\forall x\ (A(x) \Rightarrow B(x))$
  - $\exists x \sim A(f(x,y))$
  - $A(x) \Rightarrow B(x)$
  - $A(x)$
  - $B(x)$
  - $\sim A(f(x,y))$
  - $A(f(x,y))$

Induzione strutturale TER

- **Induzione strutturale sui termini:**
- Sia $\mathbb{P}$ una proprietà allora $\mathbb{P}(t)$ è verificata per ogni termine $t \in$ TER se:
  - $\mathbb{P}$ è verificata per tutti i simboli di variabile e costante.
  - $\forall\ t_1, t_2, \ldots, t_n, \in$ TER allora per ogni f: $\mathbb{P}(t_1), \mathbb{P}(t_2), \ldots, \mathbb{P}(t_n) \Rightarrow \mathbb{P}(f(t_1, t_2, \ldots, t_n))$.

Induzione strutturale

- Sia $\mathbb{P}$ una proprietà, allora $\mathbb{P}(P)$ vale per tutte le f.b.f P se:
  - $\mathbb{P}$ vale per tutti i predicati, per F e T.
  - Se valgono $\mathbb{P}(P)$ e $\mathbb{P}(Q)$, allora si può dimostrare che vale anche $\mathbb{P}(\sim P)$, $\mathbb{P}(P \wedge Q)$, $\mathbb{P}(P \vee Q)$, $\mathbb{P}(P \Rightarrow Q)$.
  - Se vale $\mathbb{P}(P)$, allora si può dimostrare che vale anche $\mathbb{P}(\forall x P)$ e $\mathbb{P}(\exists x P)$.

Campo d'azione (scope)

- Il **campo d'azione** di un quantificatore:
  - è la fbf immediatamente alla sua destra,
  - ovvero è l'espressione su cui il quantificatore ha effetto.
- Una variabile che è nel campo di azione di un quantificatore è detta **legata**, altrimenti **libera**.
  - $\forall x\ (A(x) \Rightarrow B(x)) \wedge \exists x \sim A(f(x,y))$
  - x è legata e y è libera.

Variabili libere in TER

- Sia $t \in$ TER: l'insieme FV(t) delle variabili libere in t è così definito:
- $FV(x) = \{x\}$, x variabile
- $FV(a) = \{\}$, a costante
- $FV(f(t_1, \ldots, t_n)) = FV(t_1) \cup \ldots \cup FV(t_n)$ con f funzione n-aria.

Variabili libere

- Per ogni fbf P l'insieme FV(P) delle variabili libere di P è così definito:
- $FV(F) = FV(T) = \{\}$,
- $FV(A(t_1, \ldots, t_n)) = FV(t_1) \cup \ldots \cup FV(t_n)$
- $FV(\sim P) = FV(P)$
- $FV(P_1 \wedge P_2) = FV(P_1 \vee P_2) = FV(P_1 \Rightarrow P_2) =$ $= FV(P_1) \cup FV(P_2)$
- $FV(\forall x P) = FV(\exists x P) = FV(P) - \{x\}$

Formule chiuse

- Una fbf P è **chiusa** se FV(P)={}, **aperta** altrimenti.
- BV(P) sono le variabili legate di P.
- Esempio P = $\forall x(Q(x,y) \Rightarrow R(x)) \wedge \forall y(\sim Q(f(x,y)) \Rightarrow \exists z R(z))$
- FV(P) = {x, y}, BV(P) = {x, y, z}
- x è sia libera che legata, ma ogni occorrenza di x deve essere esclusivamente libera o legata

Osservazioni

- $\forall x\ \exists y\ A(x,y)$ è diverso da $\exists x\ \forall y\ A(x,y)$
- Esempio: $A(x,y) = x<y$
  - x e y numeri naturali
  - $\forall x\ \exists y\ A(x,y)$ vera
  - $\exists y\ \forall x\ A(x,y)$ falsa
- È importante posizionare correttamente i quantificatori nella formula.

Osservazioni

- Per ogni naturale x esiste un numero y t.c. **se** x non è il successore di y allora x = 0
  - P = $\forall x\ \exists y(\sim E(x,s(y)) \Rightarrow E(x,0))$ è vera
    - se x = 0 allora P è vera perché lo è E(x,0)
    - se x ≠ 0 allora prendo y = (predecessore di x) e quindi $\sim E(x,s(y))$ è falsa
- Per ogni naturale x **se** esiste un numero y t.c. x non è il successore di y allora x=0
  - Q = $\forall x\ (\exists y \sim E(x,s(y)) \Rightarrow E(x,0))$ è falsa
    - per esempio se x=1: $\exists y \sim E(x,s(y))$ è vera perché esiste un y che non è successore di x (y=0) mentre E(x,0) è falsa

Ridenominazione

- I nomi di variabili quantificate ($\forall x$, $\exists x$) hanno il ruolo di **parametri formali**.
- In $\forall x$ P e $\exists x$ P possiamo **cambiare x in y** senza modificarne il significato.
- Il nuovo nome y **non** deve comparire nella formula P.
- Es: $\forall x(Q(x,z) \Rightarrow S(x) \wedge \exists x\ R(x))$
- Cambio $\forall x$ con $\forall y$ e le x legate al $\forall x$ con delle y
  - $\forall y(Q(y,z) \Rightarrow S(y) \wedge \exists x\ R(x))$
  - Le due formule sono equivalenti

Sostituzione TER

- Siano s,t $\in$ TER allora il termine s[t/x] è così definito:
- Se s è la variabile x allora s[t/x] = x[t/x] = t
- Se s è una variabile $y \neq x$ allora y[t/x] = y
- Se s è una costante c allora c[t/x] = c
- $f(t_1, \ldots, t_n)[t/x] = f(t_1[t/x], \ldots, t_n[t/x])$

Sostituzione

- Siano P una fbf e t $\in$ TER allora P[t/x] è così definito:
- F[t/x] = F e T[t/x] = T,
- $A(t_1, \ldots, t_n)[t/x] = A(t_1[t/x], \ldots, t_n[t/x])$
- $(\sim P)[t/x] = \sim P[t/x]$
- $(P_1 \wedge P_2)[t/x] = P_1[t/x] \wedge P_2[t/x]$
- $(P_1 \vee P)[t/x] = P_1[t/x] \vee P_2[t/x]$
- $(P_1 \Rightarrow P_2)[t/x] = P_1[t/x] \Rightarrow P_2[t/x]$

Sostituzione cont.

- Se $x \neq y$ e $y \notin FV(t)$ allora
  - $(\forall yP)[t/x] = \forall y(P[t/x])$
- Se $x \neq y$ e $y \in FV(t)$ e z non occorre in P e t allora:
  - $(\forall yP)[t/x] = \forall z(P[z/y][t/x])$
- Se $x = y$ allora
  - $(\forall yP)[t/x] = \forall y\ P$
- Se $x \neq y$ e $y \notin FV(t)$ allora
  - $(\exists yP)[t/x] = \exists y(P[t/x])$
- Se $x \neq y$ e $y \in FV(t)$ e z non occorre in P e t allora:
  - $(\exists yP)[t/x] = \exists z(P[z/y][t/x])$
- Se $x = y$ allora
  - $(\exists yP)[t/x] = \exists y\ P$

Osservazione

- La sostituzione si applica solo alle variabili libere!
- Esempio:
- $P = \forall y(Q(x,y) \Rightarrow S(x) \wedge \exists x\ R(x))$
- $P[f(y)/x] = \forall z(Q(f(y),z) \Rightarrow S(f(y)) \wedge \exists x\ R(x))$

Struttura

- Una **struttura** è una coppia S=(D, I) dove D è un insieme non vuoto detto dominio e I è un assegnamento che associa:
  - Ad ogni costante un elemento $I(c) \in D$
  - Ad ogni funzione f di arità k>0 una funzione:
    - $I(f): D^k \rightarrow D$
  - Ad ogni predicato A di arità k>0 una funzione:
    - $I(A): D^k \rightarrow \{0, 1\}$

Notazione

- $c^S = I(c)$
- $f^S = I(f)$
- $B^S = I(B)$
- Osservazione:
  - Non si può interpretare una fbf contenente variabili libere.
  - Si assegnano ad esse degli elementi di D.
  - Il valore di verità di una fbf dipende dall'assegnamento delle variabili libere.

Ambiente

- Un **ambiente** per S è la funzione
  - ξ^S: VAR $\rightarrow$D
- L'insieme di tutti i possibili ambienti è
  - $ENV_s=\{\xi^S \mid \xi^S: VAR \rightarrow D\}$
- Siano d$\in$D e x $\in$VAR, $\xi^S[d/x]$ è l'ambiente così definito:
  - se y$\neq$x allora $\xi^S[d/x](y)= \xi^S(y)$
  - se y=x allora $\xi^S[d/x](y)=$ d
- Esempio: $\xi^S(x)=0$, $\xi^S(y)=2$,$\xi^S(z)=0$ $\xi^S[2/z](x)=0$, $\xi^S[2/z](y)=2$, $\boldsymbol{\xi^S[2/z](z)=2}$

Semantica

- Un'**interpretazione** di un linguaggio proposizionale $\mathcal{L}$ è una coppia (S, ξ^S), con S struttura e ξ^S ambiente per S.
- L'interpretazione ci consente di dare un significato alle formule ben formate.

Esempio

- $P = \forall x\ A(x,f(x)) \vee B(g(c,y))$
- $y \in FV(P)$ $x \in BV(P)$
- Un'interpretazione (S, ξ^S) ristretta a P è:
  - $D = \{0,1,....\}$ (naturali)
  - $A^s=\{(m,n) \mid n, m \in D \text{ e } m>n\}$
  - $B^s=\{n \in D \mid n \text{ è primo}\}$
  - $f^s(n) = n+1$
  - $g^s(m,n) = m+n$
  - $c^s = 2$
  - $\xi^S(y) = 1$
- In questa interpretazione il significato di P è:
  - "per ogni naturale x, x>x+1 oppure 2+1 è primo"

Esempio cont.

- $P = \forall x\ A(x,f(x)) \vee B(g(c,y))$
- $y \in FV(P)$ $x \in BV(P)$
- Un'interpretazione (S, ξ^S) ristretta a P è:
  - $D = \{0,1,....\}$ (naturali)
  - $A^s=\{(m,n) \mid n, m \in D \text{ e } m>n\}$
  - **$B^s=\{n \in D \mid n = 0\}$**
  - $f^s(n) = n+1$
  - $g^s(m,n) = m+n$
  - $c^s = 2$
  - $\xi^S(y) = 1$
- In questa interpretazione il significato di P è:
  - "per ogni naturale x, x>x+1 oppure 2+1= 0"

Interpretazione: valore TER

- Sia P una fbf e (S, ξ) un'interpretazione. Per ogni termine t in P il suo valore in (S, ξ^S) è denotato dalla funzione $[\![\ ]\!]_{S,\xi}$: TER $\rightarrow$ D:
- se t è una variabile x allora $[\![t]\!]_{S,\xi} = \xi(x)$
- se t è una costante c allora $[\![t]\!]_{S,\xi} = c^S$
- se t è la funzione $f(t_1, \ldots, t_n)$ allora $[\![t]\!]_{S,\xi} = f^S([\![t_1]\!]_{S,\xi}, \ldots, [\![t_n]\!]_{S,\xi})$

35 Logica Matematica Valentina Ciriani

Esempio

- Sia l'interpretazione (S, ξ^S):
  - D = {0,1,....} (naturali)
  - $f^s(n) = n+1$
  - $g^s(m,n) = m+n$
  - $c^s = 2$
  - $\xi^S(y) = 1$
- In questa interpretazione:
  - $[\![g(c,f(y))]\!]_{S,\xi} = (2 + (1+1))$

36 Logica Matematica Valentina Ciriani

Funzione di valutazione

- Sia P una fbf e (S, ξ) un'interpretazione. La funzione di valutazione $v^{(S,\xi)}$: FBF $\rightarrow$ {0,1}:
- $v^{(S,\xi)}(F) = 0$ e $v^{(S,\xi)}(T) = 1$
- $v^{(S,\xi)}(A(t_1, \ldots, t_n)) = A^S([\![t_1]\!]_{S,\xi}, \ldots, [\![t_n]\!]_{S,\xi})$
- $v^{(S,\xi)}(\sim P)=1- v^{(S,\xi)}(P)$
- $v^{(S,\xi)}(P_1 \wedge P_2)=\min(v^{(S,\xi)}(P_1), v^{(S,\xi)}(P_2))$
- $v^{(S,\xi)}(P_1 \vee P_2)=\max(v^{(S,\xi)}(P_1), v^{(S,\xi)}(P_2))$
- $v^{(S,\xi)}(P_1 \Rightarrow P_2)=\max(1- v^{(S,\xi)}(P_1), v^{(S,\xi)}(P_2))$
- $v^{(S,\xi)}(\forall x P_1)=\min \{v^{(S,\xi[a/x])}(P_1) \mid a \in D\}$
- $v^{(S,\xi)}(\exists x P_1)=\max \{v^{(S,\xi[a/x])}(P_1) \mid a \in D\}$

Osservazioni

- Il quantificatore universale può essere visto come una congiunzione iterata
- Il quantificatore esistenziale può essere visto come una disgiunzione iterata
- Non è in genere possibile determinare il valore di verità di una fbf in un'interpretazione (S, ξ^S), in quanto D è in genere infinito.

Proprietà

- Il valore di verità di una fbf P data un'interpretazione (S, ξ^S) dipende
  - dalla restrizione di ξ^S all'insieme delle variabili libere in P.
- Ovvero: non c'è bisogno di considerare i valori delle altre variabili nell'ambiente.

Fondamenti di Logica Matematica

Lezione 12

Valentina Ciriani

DTI - Università di Milano

www.dti.unimi.it/~ciriani

Contenuti della lezione

- Logica dei predicati:
  - Soddisfacibilità
  - Modelli
  - Equivalenza semantica
  - Forma normale prenessa
  - Forma di Skolem

Soddisfacibilità

- P (fbf) è **soddisfatta** in una struttura S rispetto all'ambiente ξ, se $v^{(s,\xi)}(P) = 1$ (scriveremo $(S,\xi) \models P$).
- P (fbf) è **soddisfacibile in S** se esiste un ambiente ξ tale che $(S,\xi) \models P$.
- P (fbf) è **vera in S** se per ogni ambiente ξ si ha $(S,\xi) \models P$. S è un **modello** per P.
- P (fbf) è **soddisfacibile** se esiste una struttura S tale che P è soddisfacibile in S.
- P (fbf) è **valida** se è vera in ogni struttura (scriveremo $\models P$).

Osservazione

- Formula valida predicativa corrisponde alla tautologia proposizionale.
- Logica proposizionale:
  - per verificare una tautologia possiamo utilizzare la definizione di interpretazione (o la tavola di verità).
- Logica predicativa:
  - dovremo usare la nozione di teorema (calcolo), la semantica non è sufficiente.

Insiemi soddisfacibili

- Γ (insieme di fbf) è **soddisfacibile** se esiste una struttura S e un ambiente ξ^S tali che per ogni formula $P \in \Gamma$ si ha $(S,\xi^S) \models P$.
- S è un **modello** di Γ se S è un modello per ogni $P \in \Gamma$ (scriveremo $S \models \Gamma$).
- Γ è **valido** se ogni struttura è un modello di Γ (scriveremo $\models \Gamma$).

Conseguenza semantica

- Dati un insieme di fbf Γ e una formula P, diremo che P è una **conseguenza semantica** di Γ ($\Gamma \models P$) se per ogni struttura S e ogni ambiente ξ^S tali che
  - per ogni $Q \in \Gamma$ si ha $(S, \xi^S) \models Q$
  - si ha anche che $(S, \xi^S) \models P$.

Insoddisfacibilità

- Una fbf P è **falsa in una struttura** S sse non è soddisfacibile in S ($S \not\models P$).
- Una fbf P è **insoddisfacibile** (o contraddittoria) sse è falsa in ogni struttura.

Paradosso del barbiere

- In un paese esiste un solo barbiere.
- Il barbiere rade tutti e solo coloro che non si radono da soli.
- Chi rade il barbiere?
- **Il barbiere non si rade** da solo poiché lui rade solo chi non si rade da solo.
- Ma se non si rade da solo deve essere raso dal barbiere!
- La definizione di barbiere è contraddittoria.

Paradosso del barbiere

- Formalizziamo la definizione di barbiere:
  - R(x,y) = “x rade y”
  - b = barbiere
- Il barbiere rade tutti coloro che non si radono da soli:
  - $\forall x(\sim R(x,x) \Rightarrow R(b,x))$
- Il barbiere rade solo coloro che non si radono da soli:
  - $\forall x(R(b,x) \Rightarrow \sim R(x,x))$

Paradosso del barbiere

- Il barbiere rade tutti e solo coloro che non si radono da soli:
  - $\forall x(R(b,x) \Leftrightarrow \sim R(x,x))$
- È insoddisfacibile (è falsa in ogni struttura)
- Dim: Sia S una struttura qualsiasi
  $v^{(S, \xi)}(\forall x(R(b,x) \Leftrightarrow \sim R(x,x)))=$
  $\min \{v^{(S, \xi[a/x])}(R(b,x) \Leftrightarrow \sim R(x,x)) \mid a \in D\} = 0.$
  Per b ($I(b) \in D$) otteniamo:
  - $R(b,b) \Leftrightarrow \sim R(b,b)$ che è falso!
- La definizione di Barbiere è contraddittoria: non esiste nessun barbiere con tale definizione! Da questo deriva il paradosso.

Proprietà

- P è valida sse ~P è insoddisfacibile
- P è soddisfacibile sse ~P non è valida
- **$\Gamma \models$ P sse $\Gamma \cup$ {~P} è insoddisfacibile**
- P non è valida
  - è equivalente a dire ~P è soddisfacibile
  - **non** è equivalente a dire che P è insoddisfacibile ovvero che ~P è valida.

Lemma di soddisfacibilità

- Il seguente lemma stabilisce la relazione tra la definizione di soddisfacibilità e il significato intuitivo dei connettivi.
- **Lemma** Per ogni fbf P e Q ed ogni interpretazione (S, ξ^S) con S=(D,I):
  1. $(S, \xi^S) \models P \wedge Q$ sse $(S, \xi^S) \models P$ e $(S, \xi^S) \models Q$
  2. $(S, \xi^S) \models P \vee Q$ sse $(S, \xi^S) \models P$ o $(S, \xi^S) \models Q$
  3. $(S, \xi^S) \models {\sim}P$ sse $(S, \xi^S) \not\models P$
  4. $(S, \xi^S) \models P \Rightarrow Q$ sse $(S, \xi^S) \not\models P$ o $(S, \xi^S) \models Q$
  5. $(S, \xi^S) \models \forall x P$ sse $(S, \xi^S[a/x]) \models P$ per ogni $a \in D$
  6. $(S, \xi^S) \models \exists x P$ sse $(S, \xi^S[a/x]) \models P$ per qualche $a \in D$

Dimostrazione Lemma

4. $(S, \xi^S) \models P \Rightarrow Q$ sse $(S, \xi^S) \not\models P$ o $(S, \xi^S) \models Q$
   Dim ($\Rightarrow$). Se $(S, \xi^S) \not\models P$ abbiamo finito. Se invece $(S, \xi^S) \models P$ allora $v^{(S, \xi)}(P)=1$. Ma: $1=v^{(S, \xi)}(P \Rightarrow Q) = \max(1-v^{(S, \xi)}(P), v^{(S,\xi)}(Q)) = v^{(S, \xi)}(Q)$ e quindi $(S, \xi^S) \models Q$.
   Dim ($\Leftarrow$). Supponiamo per assurdo che $(S, \xi^S) \not\models P \Rightarrow Q$ ovvero $v^{(S, \xi)}(P \Rightarrow Q) = 0 = \max(1-v^{(S, \xi)}(P), v^{(S,\xi)}(Q))$ e quindi $v^{(S, \xi)}(P)=1$ e $v^{(S,\xi)}(Q) =0$ che contraddice l'ipotesi.

Dimostrazione Lemma

5. $(S, \xi^S) \models \forall xP$ sse $(S, \xi^S[a/x]) \models P$ per ogni $a \in D$
   Dim.
   $(S, \xi^S) \models \forall xP$ sse $\min \{v^{(S, \xi[a/x])}(P) \mid a \in D\} =1$ sse per ogni $a \in D$ $v^{(S, \xi[a/x])}(P) =1$ sse $(S, \xi^S[a/x]) \models P$ per ogni $a \in D$.

- Per esercizio: dimostrare tutti gli altri casi.

Equivalenza semantica

- Due fbf P e Q sono semanticamente equivalenti ($P \equiv Q$) se per tutte le interpretazioni (S, ξ^S) di P e Q si ha:
  - $v^{(S, \xi)}(P) = v^{(S, \xi)}(Q)$
- Due fbf P e Q sono equivalenti sse
  - $\models P \Leftrightarrow Q$

15 Logica Matematica Valentina Ciriani

Proprietà

- Siano P una fbf e z una variabile che **non** occorre in P. Allora:
  - $\exists xP \equiv \exists zP[z/x]$
  - $\forall xP \equiv \forall zP[z/x]$
- Sia P una fbf:
  - $\sim\exists xP \equiv \forall x\sim P$
  - $\sim\forall xP \equiv \exists x\sim P$
  - $\exists xP \equiv \sim\forall x\sim P$
  - $\forall xP \equiv \sim\exists x\sim P$

16 Logica Matematica Valentina Ciriani

Proprietà

- Sia P una fbf:
  - $\exists x \exists y P \equiv \exists y \exists x P$
  - $\forall x \forall y P \equiv \forall y \forall x P$
  - $\exists x P \equiv P$ se $x \notin FV(P)$
  - $\forall x P \equiv P$ se $x \notin FV(P)$

Proprietà

- Sia P e Q fbf:
  - $\forall x(P \wedge Q) \equiv \forall x P \wedge \forall x Q$
  - $\exists x(P \vee Q) \equiv \exists x P \vee \exists x Q$
  - $\forall x(P \vee Q) \equiv \forall x P \vee Q$ se $x \notin FV(Q)$
  - $\exists x(P \wedge Q) \equiv \exists x P \wedge Q$ se $x \notin FV(Q)$
- Osservazione, in generale:
  - $\forall x(P \vee Q) \not\equiv \forall x P \vee \forall x Q$
  - $\exists x(P \wedge Q) \not\equiv \exists x P \wedge \exists x Q$

Esempio

- Per ogni intero x, x è pari o x è dispari:
  - $\forall x(P(x) \vee D(x))$
- Ogni intero x è pari o ogni intero x è dispari:
  - $\forall x P(x) \vee \forall x D(x)$

Proprietà

- Basta ridenominare le variabili!
- Sia P e Q fbf e z una **nuova** variabile:
  - $Q_1xP \vee Q_2xQ \equiv Q_1xQ_2z(P \vee Q[z/x])$
  - $Q_1xP \wedge Q_2xQ \equiv Q_1xQ_2z(P \wedge Q[z/x])$

  con $Q_i \in \{\forall, \exists\}$.
- Esempio:
  - $\forall x\ (P(x) \wedge Q(x)) \vee \exists x\ (Q(x) \wedge P(z))$
  - $\equiv \forall x\ \exists y\ ((P(x) \wedge Q(x)) \vee (Q(y) \wedge P(z)))$

Proprietà

- Sia P e Q fbf e $x \notin FV(Q)$:
  - $\forall xP \Rightarrow Q \equiv \exists x(P \Rightarrow Q)$
  - $\exists xP \Rightarrow Q \equiv \forall x(P \Rightarrow Q)$
  - $Q \Rightarrow \exists xP \equiv \exists x(Q \Rightarrow P)$
  - $Q \Rightarrow \forall xP \equiv \forall x(Q \Rightarrow P)$
- Nel caso $x \in FV(Q)$ basta ridenominare la variabile x in P con una nuova.
- Es: $\forall xA(x) \Rightarrow B(x) \equiv \forall yA(y) \Rightarrow B(x) \equiv$ $\equiv \exists y(A(y) \Rightarrow B(x))$

Forme normali prenesse

- Una fbf P è in forma normale prenessa se ha la forma:
  - $Q_1x_1Q_2x_2 \ldots Q_nx_n P'$
  - con $Q_i \in \{\forall, \exists\}$ e P' non contiene quantificatori.
- Es: $\forall y\exists x(A(y) \Rightarrow B(x))$
- È sempre possibile trasformare una formula in forma normale prenessa

Forme normali prenesse

- Per trasformare una f.b.f in forma FNP
  - basta controllare che tutti i quantificatori abbiano variabili diverse tra loro e che abbiano nomi diversi dalle variabili libere, in caso contrario ridenominarle con variabili **nuove**.
  - poi utilizzare le proprietà descritte prima per portare i quantificatori fuori dalla formula

Algoritmo di trasformazione

1. ridenominare, con nomi **nuovi**, tutte le variabili legate che hanno lo stesso nome di alcune variabili libere delle formula.
2. ridenominare, con nomi **nuovi**, tutte le variabili legate che hanno lo stesso nome di altre variabili legate.
3. estrarre i quantificatori fuori dalle formule usando le seguenti regole:
   - $\sim\exists xP \equiv \forall x\sim P$ e $\sim\forall xP \equiv \exists x\sim P$
   - $Q_1xP \vee Q_2yQ \equiv Q_1xQ_2y(P \vee Q)$ con $Q_i \in \{\forall,\exists\}$
   - $Q_1xP \wedge Q_2yQ \equiv Q_1xQ_2y(P \wedge Q)$ con $Q_i \in \{\forall,\exists\}$
   - $\forall xP \Rightarrow Q \equiv \exists x(P \Rightarrow Q)$ e $\exists xP \Rightarrow Q \equiv \forall x(P \Rightarrow Q)$
   - $P \Rightarrow \exists xQ \equiv \exists x(P \Rightarrow Q)$ e $P \Rightarrow \forall xQ \equiv \forall x(P \Rightarrow Q)$

Correttezza dell'algoritmo

- Le regole del passo 3 dell'algoritmo sono corrette poiché
  - tutti i parametri dei quantificatori sono diversi (per il passo 2)
  - non ci sono variabili libere con lo stesso nome di variabili legate (per il passo 1)
  - posso quindi usare le equivalenze discusse prima in quanto le loro ipotesi sono vere.

Esempio

- $\forall x A(x) \Rightarrow \sim\forall x B(x)$
- $\forall x A(x) \Rightarrow \sim\forall y B(y)$
- $\forall x A(x) \Rightarrow \exists y \sim B(y)$
- $\exists x (A(x) \Rightarrow \exists y \sim B(y))$
- $\exists x \exists y (A(x) \Rightarrow \sim B(y))$

Esempio

$\exists x\,(A(x)\vee C(y)) \Rightarrow \sim(\forall x\, A(x)\wedge(\exists x\, B(x)\Rightarrow \exists y{\sim}B(y)))$

1. $\exists x\,(A(x)\vee C(y)) \Rightarrow \sim(\forall x\, A(x)\wedge(\exists x\, B(x)\Rightarrow \exists h{\sim}B(h)))$
2. $\exists x\,(A(x)\vee C(y)) \Rightarrow \sim(\forall z\, A(z)\wedge(\exists k\, B(k)\Rightarrow \exists h{\sim}B(h)))$
3. $\exists x\,(A(x)\vee C(y)) \Rightarrow \sim(\forall z\, A(z)\wedge \forall k\,(B(k)\Rightarrow \exists h{\sim}B(h)))$

$\exists x\,(A(x)\vee C(y)) \Rightarrow \sim(\forall z\, A(z)\wedge \forall k\exists h\,(B(k)\Rightarrow {\sim}B(h)))$

$\exists x\,(A(x)\vee C(y)) \Rightarrow \sim\forall z\forall k\exists h\,(A(z)\wedge (B(k)\Rightarrow {\sim}B(h)))$

$\exists x\,(A(x)\vee C(y)) \Rightarrow \exists z\exists k\forall h \sim(A(z)\wedge (B(k)\Rightarrow {\sim}B(h)))$

$\forall x((A(x)\vee C(y)) \Rightarrow \exists z\exists k\forall h \sim(A(z)\wedge (B(k)\Rightarrow {\sim}B(h))))$

$\forall x\exists z\exists k\forall h\,((A(x)\vee C(y)) \Rightarrow \sim(A(z)\wedge (B(k)\Rightarrow {\sim}B(h))))$

Forma di Skolem

- Possiamo eliminare i quantificatori esistenziali introducendo dei simboli di funzione.
- La formula di Skolem ottenuta:
  - **non** è equivalente a quella originale
  - è soddisfacibile sse lo è quella di partenza.

Esempio 1

- $\exists x A(x)$
  - Esiste un x per cui vale A(x)
  - introduco un simbolo di costante c e scrivo: A(c)

Esempio 2

- $\forall x \exists y B(x,y)$
  - Per ogni x esiste un y t.c. vale B(x,y)
  - Es: B(x,y) = y è multiplo di x
  - Quindi per due x diversi posso avere due y diversi (non posso fissare y con una costante).
  - Introduciamo un simbolo di funzione che dipende da x!
  - $\forall x B(x,f(x))$
  - Ha significato diverso ma è ancora soddisfacibile.

Forma di Skolem

- Sia P = $Q_1x_1Q_2x_2 \dots Q_nx_n\ P_1$ una fbf in FNP. Si dice **forma di Skolem** di P (P^s) la fbf così ottenuta su ogni $\exists x_i$:
  - Se $\exists x_i$ non è nello scope di un $\forall$ si introduce c (nuova costante) al posto di x_i in P_1 ($P_1[c/x_i]$) e si elimina $\exists x_i$.
  - Se $\exists x_i$ è nello scope di alcuni $\forall$ ovvero $Q_{i,1}x_{i,1},\dots, Q_{i,n}x_{i,n}$ allora si introduce un nuovo simbolo di funzione $g(x_{i,1},\dots,x_{i,n})$ al posto di x_i in P_1 ($P_1[g(x_{i,1},\dots,x_{i,n})/x_i]$) e si elimina $\exists x_i$.

Esempio di trasformazione

- $\exists x \forall y \forall z \exists t\ B(x, y, z, t)$

  $\forall y \forall z \exists t\ \ B(c, y, z, t)$

  $\forall y \forall z\ \ B(c, y, z, g(y,z))$

Esempio di trasformazione

$\forall x \exists z \exists k \forall y\ ((A(x) \vee C(h)) \Rightarrow \sim(A(z) \wedge (B(k) \Rightarrow \sim B(y))))$

f è una nuova funzione

g è una nuova funzione

$\forall x \forall y\ ((A(x) \vee C(h)) \Rightarrow \sim(A(f(x)) \wedge (B(g(x)) \Rightarrow \sim B(y))))$

Teorema di Skolem

- Per ogni fbf P, P è soddisfacibile sse P^s lo è.
- Basta dimostrare che le due regole di eliminazione del quantificatore esistenziale preservano la soddisfacibilità della formula.

Dimostrazione

- **Dim.** Siano $Q_i \in \{\forall, \exists\}$ e
  - $R = Q_1 x_1 \ldots \exists x_t \ldots Q_n x_n \, A(x_1, \ldots x_t, \ldots x_n)$
  - $R' = Q_1 x_1 \ldots Q_{t-1} x_{t-1} \, Q_{t+1} x_{t+1} \ldots Q_n x_n$
    $A(x_1, \ldots, x_{t-1}, f(x_{i,1}, \ldots, x_{i,m}), x_{t+1}, \ldots x_n)$
- dove $Q_{i,1}, \ldots Q_{i,m}$ sono i $\forall$ che hanno $\exists x_t$ nel campo d'azione in R.
- **Caso 1:** sia R insoddisfacibile, allora anche R' lo è. Altrimenti esisterebbe un'interpretazione che è modello per R' ovvero (per il lemma di soddisf.), si avrebbe che per ogni $x_{i,1}, \ldots, x_{i,m}$ esiste $f(x_{i,1}, \ldots, x_{i,m})$ che verifica:
  $Q_{t+1} x_{t+1} \ldots Q_n x_n \, A(x_1, \ldots, x_{t-1}, f(x_{i,1}, \ldots, x_{i,m}), x_{t+1}, \ldots x_n)$
- Ma questo e' assurdo poiché R è insoddisfacibile.

Dimostrazione cont.

- **Dim.** Siano $Q_i \in \{\forall, \exists\}$ e
  - $R = Q_1 x_1 \ldots \exists x_t \ldots Q_n x_n \, A(x_1, \ldots x_t, \ldots x_n)$
  - $R' = Q_1 x_1 \ldots Q_{t-1} x_{t-1} \, Q_{t+1} x_{t+1} \ldots Q_n x_n$
    $A(x_1, \ldots, x_{t-1}, f(x_{i,1}, \ldots, x_{i,m}), x_{t+1}, \ldots x_n)$
- dove $Q_{i,1}, \ldots Q_{i,m}$ sono i $\forall$ che hanno $\exists x_t$ nel campo d'azione in R.
- **Caso 2:** sia R soddisfacibile, allora anche R' lo è. Sia (S, ξ^S) un modello per R. Allora per ogni $x_{i,1}, \ldots, x_{i,m}$ in D esiste x_t in D che verifica:
  $Q_{t+1} x_{t+1} \ldots Q_n x_n \, A(x_1, \ldots, x_{t-1}, x_t, x_{t+1}, \ldots x_n)$
- Ma ponendo $f^S(x_{i,1}, \ldots, x_{i,m}) = x_t$ abbiamo un modello per R'.

Forma di Skolem di fbf aperte

- Se P è una fbf aperta possiamo trasformarla in una fbf chiusa e in forma di Skolem.
- Algoritmo:
  - sia $FV(P)= \{x_1, ..., x_n\}$ allora porre $P' = \exists x_1, ..., x_n\ P$
  - trasformare P' in forma di Skolem

Fondamenti di Logica Matematica
Lezione 13

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Calcolo del primo ordine:
  - Deduzione naturale
  - Sequenti
  - Traduzione dal linguaggio naturale

Teorema di Church

- Nella logica predicativa:
  - Non possiamo ispezionare una tavola di verità: abbiamo infinite interpretazioni!
  - Il calcolo è indispensabile.
- Teorema di Church
  - La nozione di verità logica è **indecidibile**.
  - Ovvero non esiste un algoritmo che, data una formula predicativa P, termini sempre dicendo che P è vera o meno.

Semidecidibilità

- Il calcolo predicativo è **semidecidibile**.
- Ovvero data una formula predicativa P:
  - se P è vera allora il calcolo terminerà sempre dicendo che P è vera.
  - se P non è vera allora il calcolo potrebbe non terminare mai, se termina dice che P non è vera.

Calcolo predicativo

- Verranno aggiunte delle regole al calcolo proposizionale.
- Le regole aggiunte saranno per i due quantificatori:
  - *Deduzione naturale*: eliminazione e introduzione di ∀ e ∃.
  - *Sequenti*: regole per l'introduzione a destra e a sinistra di ∀ e ∃.

Deduzione naturale

- Quantificatore universale:

$$\frac{P[y/x]}{\forall x P}$$ y non compare libera nelle foglie derivate da P[y/x] (∀i)

$$\frac{\forall x P}{P[t/x]}$$ t termine qualsiasi (∀e)

Deduzione naturale

- Quantificatore esistenziale:

$$\frac{P[t/x]}{\exists x P} \quad t \text{ termine qualsiasi} \qquad (\exists i)$$

$$\frac{\exists x P \qquad \begin{matrix}[P[y/x]\,] \\ Q\end{matrix}}{Q} \quad \text{y non compare libera nelle foglie derivate da P[y/x]} \qquad (\exists e)$$

Esempio di deduzione

- $\forall x(A \Rightarrow B(x)) \vdash_{DN} A \Rightarrow \forall x\, B(x)$

$$\dfrac{\dfrac{\dfrac{\dfrac{[\forall x(A \Rightarrow B(x))]}{A \Rightarrow B(x)} \qquad [A]}{B(x)}}{\forall x B(x)}}{A \Rightarrow \forall x\, B(x)}\ \text{ip. A}$$

Calcolo dei sequenti

- $\dfrac{\Gamma, P[y/x] \vdash \Delta}{\Gamma, \exists x P \vdash \Delta}$ (∃-l)

- $\dfrac{\Gamma \vdash P[t/x], \exists x P, \Delta}{\Gamma \vdash \exists x P, \Delta}$ (∃-r)

Calcolo dei sequenti

- $\dfrac{\Gamma, \forall x P, P[t/x] \vdash \Delta}{\Gamma, \forall x P \vdash \Delta}$ (∀-l)

- $\dfrac{\Gamma \vdash P[y/x], \Delta}{\Gamma \vdash \forall x P, \Delta}$ (∀-r)

Osservazione

- Nelle regole (∃-l) e (∀-r):
  - la variabile y deve essere **nuova** (non esistente nella formula)

- Nelle regole (∃-r) e (∀-l):
  - il termine t deve essere scelto tra i **vecchi** se ce ne sono. Altrimenti ne prendiamo uno nuovo (es. una var nuova)
  - non si scelgono le variabili legate
  - si scelgono solo le variabili libere e gli altri termini

Osservazione

- Nelle regole (∃-r) e (∀-l) c'è la ricopiatura della formula quantificata.
- Questo rende i sequenti reversibili:
  - ovvero la conclusione è semanticamente valida sse lo sono le premesse.
- Ma il calcolo può rivelarsi infinito!

Esempio

...

$$\cfrac{\cfrac{\cfrac{\cfrac{}{A(r), A(s), A(t), \forall x\, A(x) \vdash \forall x\, (A(x) \wedge B)}}{A(s), A(t), \forall x\, A(x) \vdash \forall x\, (A(x) \wedge B)}}{A(t), \forall x\, A(x) \vdash \forall x\, (A(x) \wedge B)}}{\forall x\, A(x) \vdash \forall x\, (A(x) \wedge B)}$$

- potrebbe andare avanti all'infinito.

Criterio di equità

- Criterio di equità:
  - Nell'esplorare un ramo, nessuna regola deve essere applicata all'infinito.
- Usando questo criterio ogni formula vera verrà dimostrata.

Utilizzo del calcolo

- Come per la logica proposizionale il calcolo dei sequenti può essere usato per:
  - la ricerca di dimostrazioni
  - la ricerca di contromodelli

Ricerca di dimostrazioni

- Per dimostrare che un sequente $\Gamma \vdash \Delta$ è una verità logica:
  - lo si scrive in basso
  - si usano le regole proposizionali e predicative (con il criterio di equità)
  - la ricerca finisce quando si arriva, in tutti i rami della dimostrazione, ad un assioma del tipo $A(t_1, ..., t_n) \vdash A(t_1, ..., t_n)$.

Esempio

$$\forall x\,(A(x) \wedge B(x)) \vdash \forall x\, A(x) \wedge \forall x\, B(x)$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A(y)\vdash A(y)}{\forall x(A(x)\wedge B(x)),A(y),B(y)\vdash A(y)}}{\forall x(A(x)\wedge B(x)),A(y)\wedge B(y)\vdash A(y)}}{\forall x\,(A(x)\wedge B(x))\vdash A(y)}}{\forall x\,(A(x)\wedge B(x))\vdash \forall x\, A(x)} \quad \dfrac{\dfrac{\dfrac{\dfrac{B(y)\vdash B(y)}{\forall x(A(x)\wedge B(x)),A(y),B(y)\vdash B(y)}}{\forall x(A(x)\wedge B(x)),A(y)\wedge B(y)\vdash B(y)}}{\forall x\,(A(x)\wedge B(x))\vdash B(y)}}{\forall x\,(A(x)\wedge B(x))\vdash \forall x B(x)}}{\forall x\,(A(x)\wedge B(x)) \vdash \forall x\, A(x) \wedge \forall x\, B(x)}$$

Esempio

$$\exists x\,(A(x) \wedge B(x)) \vdash \exists x\, A(x)$$

$$\boxed{(\exists\text{-r})}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{A(y) \vdash A(y)}{A(y),\, B(y) \vdash \exists x\, A(x),\, A(y)}\ \text{(y già esistente)}}{A(y),\, B(y) \vdash \exists x\, A(x)}}{A(y) \wedge B(y) \vdash \exists x\, A(x)}\ \text{(y nuova var)}}{\exists x\,(A(x) \wedge B(x)) \vdash \exists x\, A(x)}$$

$$\boxed{(\exists\text{-l})}$$

Ricerca di contromodelli

- Si procede come per la ricerca di dimostrazioni.
- Si considerano solo due tipo di rami:
  - quelli che terminano senza arrivare ad un assioma.
  - quelli che non terminano nonostante l'uso del criterio di equità.

Rami che terminano

- Abbiamo lo stato terminale con fallimento quando:
  - ci sono solo formule atomiche o quantificatori universali prima di $\vdash$ o esistenziali dopo di $\vdash$.
  - sono state fatte tutte le possibili istanziazioni A(t/x), con termini t presenti nel ramo, per i quantificatori universali prima di $\vdash$ ed esistenziali dopo di $\vdash$.

Contromodello

- Struttura S=(D, I) e ambiente ξ^S
- D = {tutti i termini, tranne le variabile legate, nel ramo}
- $\xi^S(x) = x$ per tutte le variabili libere nel ramo
- $I(c) = c$ per tutte le costanti nel ramo
- $I(f(t_1, t_2, \ldots, t_n)) = f(t_1, t_2, \ldots, t_n)$ per tutte le funzioni nel ramo
- $I(A(t_1, t_2, \ldots, t_n)) = 1$ sse $A(t_1, t_2, \ldots, t_n)$ compare a sinistra di $\vdash$.

Esempio

- $\forall x(A(x) \Rightarrow B(x)) \vdash \exists x A(x) \Rightarrow \forall x B(x)$

$\underline{\forall x(A(x)\Rightarrow B(x)),\ B(y),\ A(y)\vdash B(z),\ A(z) \qquad ....}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ B(y),\ A(z)\Rightarrow B(z),\ A(y)\vdash B(z) \qquad ....}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ A(y)\Rightarrow B(y),\ A(z)\Rightarrow B(z),\ A(y)\vdash B(z)}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ A(y)\Rightarrow B(y),\ A(y)\vdash B(z)}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ A(y)\vdash B(z)}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ A(y)\vdash \forall x B(x)}$
$\underline{\forall x(A(x)\Rightarrow B(x)),\ \exists x A(x)\vdash \forall x B(x)}$
$\forall x(A(x)\Rightarrow B(x))\vdash \exists x A(x)\Rightarrow \forall x B(x)$

Esempio

- ramo terminato senza assiomi:
  - $\forall x(A(x) \Rightarrow B(x)), B(y), A(y) \vdash B(z), A(z)$
- contromodello:
- $D=\{y,z\}$
- $\xi^S(y) = y$, $\xi^S(z) = z$
- $I(B(y)) = I(A(y)) = 1$
- $I(B(z)) = I(A(z)) = 0$

$$\underbrace{\forall x(A(x) \Rightarrow B(x))}_{1} \vdash \underbrace{\exists x A(x) \Rightarrow \forall x B(x)}_{0}$$

Rami che non terminano

- Caso più difficile:
  - bisogna prevedere l'andamento del ramo infinito.
  - Non sempre è possibile trovare un contromodello.
- Vediamo un esempio.

Esempio

- $\exists x A(x), \forall y(A(y) \Rightarrow A(f(y))) \vdash$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A(z), A(f(z)), \ldots, A(f^i(z)), \forall y(A(y) \Rightarrow A(f(y))) \vdash \quad \ldots}{\ldots\ldots}}{A(z), \forall y(A(y) \Rightarrow A(f(y))), A(f(z)), A(f(z)) \Rightarrow A(f(f(z))) \vdash \ldots}}{A(z), \forall y(A(y) \Rightarrow A(f(y))), A(f(z)) \vdash \quad \ldots}}{A(z), \forall y(A(y) \Rightarrow A(f(y))), A(z) \Rightarrow A(f(z)) \vdash}}{A(z), \forall y(A(y) \Rightarrow A(f(y))) \vdash}}{\exists x A(x), \forall y(A(y) \Rightarrow A(f(y))) \vdash}$$

Esempio

$A(z), A(f(z)), \ldots, A(f^i(z)), \forall y(A(y) \Rightarrow A(f(y))) \vdash \ldots$

- contromodello:
- $D = \{z, f(z), f(f(z)), \ldots, f^i(z), \ldots\}$
- $\xi^S(z) = z$
- $I(f^i(z)) = f^i(z)$ per ogni $i > 0$
- $I(A(f^i(z))) = 1$ per ogni $i > 0$
- $I(A(z)) = 1$

$$\underbrace{\exists x A(x), \forall y(A(y) \Rightarrow A(f(y)))}_{1} \vdash$$

Esempio

- In realtà potevamo trovare un contromodello più semplice:
- D={a}
- I(a) = a
- I(f(a)) = f(a)
- I(A(a))= 1

$$\underbrace{\exists x A(x), \forall y(A(y) \Rightarrow A(f(y)))}_{1} \vdash$$

Teorema di correttezza

- Sia un sequente $\Gamma \vdash \Delta$ della logica predicativa, allora non esiste nessuna interpretazione (S, ξ^S) per cui si abbia $(S,\xi^S) \vDash P$ per ogni $P \in \Gamma$ e $(S,\xi^S) \nvDash Q$ ogni $Q \in \Delta$.
- Ovvero: se P è dimostrabile con i sequenti allora P è una verità logica.

Teorema di completezza

- Se il sequente $\Gamma \vdash \Delta$ non è dimostrabile allora esiste una interpretazione (S,ξ^S) tale che $(S,\xi^S) \models P$ per ogni $P \in \Gamma$ e $(S,\xi^S) \not\models Q$ ogni $Q \in \Delta$.
- Ovvero: se P non è dimostrabile con i sequenti allora P non è una verità logica.

Traduzione dal linguaggio naturale

- Con il linguaggio logico dei predicati possiamo tradurre molte più frasi del linguaggio naturale
- La traduzione non è però sempre banale

Quantificatore universale

- Spesso $\forall$ è abbinato ad una **implicazione**:
  - l'antecedente dell'implicazione restringe l'ambito in cui si applica il quantificatore
- Esempio:
  - Tutti gli uomini sono mortali
  - U(x) = "x è un uomo"
  - M(x) = "x è mortale"
  - $\forall x\ (U(x) \Rightarrow M(x))$
  - U(x) restringe l'ambito in cui si applica il $\forall$

31 Logica Matematica Valentina Ciriani

Quantificatore esistenziale

- Spesso $\exists$ è abbinato ad una **congiunzione**:
  - la congiunzione contiene le varie proprietà dell'elemento descritto
- Esempio:
  - Ci sono barbieri che radono se stessi
  - B(x) = "x è un barbiere"
  - R(x,y) = "x rade y"
  - $\exists x\ (B(x) \wedge R(x,x))$
  - B(x) e R(x,x) sono le proprietà che ha l'elemento x

32 Logica Matematica Valentina Ciriani

Esempi

- C'è un libro che non è utilizzato da nessun studente

  $\exists x\ (L(x) \wedge \forall y\ (S(y) \Rightarrow \sim U(y,x))$

  oppure $\exists x\ (L(x) \wedge \sim\exists y\ (S(y) \wedge U(y,x))$

- Se Ada è più colta di Sara allora almeno un libro di Sara è stato sfogliato da tutti gli amici di Ada

  $PC(a,s) \Rightarrow (\exists x\ (L(x,s) \wedge (\forall y\ (A(y,a) \Rightarrow S(y,x)))$

Esercizio

- "Aldo e Carlo sono fratelli"
- "Carlo ha un cugino"
- "Allora Aldo ha un cugino"

- Dobbiamo usare dei postulati di significato, ovvero:
  - Due fratelli sono figli dello stesso genitore
  - Due cugini sono figli di fratelli

Esercizio

- $G(x,y)$: x è genitore di y.
- $F(x,y)$: x è fratello di y.
- $C(x,y)$: x è cugino di y.
- $F(x,y) =$
  $\forall z((G(z,x) \Rightarrow G(z,y)) \wedge (G(z,y) \Rightarrow G(z,x)))$
- $C(x,y) =$
  $\exists v \exists w(F(v,w) \wedge G(w,y) \wedge G(v,x)) =$
  $\exists v \exists w(\forall z((G(z,v) \Rightarrow G(z,w)) \wedge (G(z,w) \Rightarrow G(z,v))) \wedge G(w,y) \wedge G(v,x))$

Esercizio

- Dobbiamo dimostrare che:
- $F(a,c), \exists y C(a,y) \models \exists y C(c,y)$
- Esercizio:
  - scrivere le formule in funzione di G,
  - dimostrare la conseguenza logica.

Fondamenti di Logica Matematica

Esercitazione 14

Valentina Ciriani

DTI - Università di Milano

www.dti.unimi.it/~ciriani

Es 4.2 libro pag. 130

- **Determinare le variabili libere nelle seguenti formule:**
  - $\forall y \exists x A(x,y) \Rightarrow B(x,y)$
  - $\exists y \exists x (A(x,y) \Rightarrow B(x,y))$
  - $\sim\forall y \exists x A(y) \Rightarrow (B(x,y) \wedge \forall z C(x,z))$
  - $\exists x \exists y (A(x,y) \Rightarrow B(x)) \Rightarrow \forall z C(z) \vee D(z)$

Es 4.3 libro pag. 130

- **Data la fbf Q:**
  - $Q = \exists x \forall y A(y,f(x),g(z))$
- **definire un'interpretazione che è un modello per Q.**
- **definire un'interpretazione che non è un modello per Q.**

Es 4.5 libro pag. 130

- **Sia (S, ξ^S) un'interpretazione tale che:**
  - **D** = $\mathbb{N}$ = {0,1,2,...}
  - I(A) = {(n,m) | n, m $\in$ D e n $\geq$ m}
  - I(B) = {(n,m,p) | n, m, p $\in$ D e n + m = p}
- Dire se le seguenti formule sono vere in **(S, ξ^S):**
  - $\forall x \forall y \forall z(B(x,y,z) \Rightarrow B(y,x,z))$
  - $\forall x \exists y(B(x,x,y) \Rightarrow B(y,x,y))$
  - $\exists x \exists y(A(x,y) \vee \sim B(y,x,y))$

Es 4.8 libro pag. 131

- **Stabilire quali delle seguenti formule sono valide e quali sono solo soddisfacibili. Nel secondo caso fornire un contromodello:**
  - $(\exists x A(x) \Rightarrow \forall x B(x)) \Rightarrow \forall x(A(x) \Rightarrow B(x))$
  - $(\exists x A(x) \Rightarrow \exists x B(x)) \Rightarrow \forall x(A(x) \Rightarrow B(x))$
- *Suggerimento:* usare il calcolo dei sequenti per verificare la validità e per trovare l'eventuale contromodello.

Es 4.15 libro pag. 132

- **Trasformare le seguenti fbf in forma di Skolem:**
  - $\forall y(\exists x A(x,y) \Rightarrow B(y,x)) \wedge \exists y(\forall x C(x,y) \vee B(x,y))$
  - $\exists x \forall y \exists z D(x,y,z) \vee \exists x \forall y A(x,y) \wedge \sim\exists x \exists y B(x,y)$
  - $\sim(\forall x \exists y A(x,y) \Rightarrow \exists x \exists y B(x,y)) \wedge \forall x \sim\exists y B(y,x)$

Es 4.17 libro pag. 132

- **Si scriva una tautologia P tale che la skolemizzazione di P (P^s) non sia una tautologia.**

Esercizio

- **Dimostrare con il calcolo dei sequenti che:**
  - $\forall x(A(x) \Rightarrow B(x) \wedge C(x)) \vdash \forall x(A(x) \Rightarrow B(x)) \wedge \forall x(A(x) \Rightarrow C(x))$
  - $\vdash \exists x(A(x) \Rightarrow \exists y A(y))$
  - $\vdash \exists x(\exists y A(y) \Rightarrow A(x))$

Esercizio

- **Si considerino le seguenti proposizioni:**
  - (a) Tutti gli uomini sono mortali.
  - (b) Socrate è un uomo.
  - (c) Socrate è mortale.
- **Dimostrare, utilizzando il calcolo dei sequenti, che:**
  - (a), (b) $\vdash$ (c)

Fondamenti di Logica Matematica

Lezione 15

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Metodo di risoluzione:
  - Teoria di Herbrand
    - Algoritmo di Herbrand
  - Risoluzione nella logica proposizionale
    - Algoritmo di risoluzione
    - Algoritmo di refutazione

Metodi di risoluzione

- Metodi per provare la validità di una formula:
  - teorema di Herbrand (semantico)
  - metodo di risoluzione (sintattico)
- Sono metodi per **refutazione**:
  - Per dimostrare che P è valida si dimostra che la negazione di P è insoddisfacibile.

Teoria di Herbrand

- Logica proposizionale:
  - il numero di interpretazioni è finito.
- Logica dei predicati:
  - il numero di possibili interpretazioni è infinito, infatti possiamo avere infiniti domini!
- Soluzione: *dominio canonico* H
  - P è insoddisfacibile se è falsa in tutte le interpretazioni di H.

Notazione

- Useremo solo formule chiuse e in forma di Skolem.
- Trasformazione di una fbf P:
  - Chiusura esistenziale, se $FV(P)=\{x_1, ..., x_n\}$ è: $\exists x_1,...,x_n P$.
  - si trasforma $\exists x_1,...,x_n P$ in forma di Skolem.
- Sia P una fbf. È possibile trasformare P in una formula P' chiusa e in forma di Skolem tale che:
  - P è soddisfacibile sse P' è soddisfacibile.

Universo di Herbrand

- L'universo di Herbrand, *H(P)*, di una formula chiusa del primo ordine P è l'insieme dei termini contenente:
  - tutte le costanti che compaiono in P
  - tutti i termini costruiti applicando i simboli di funzione che occorrono in P a tutti i termini di H(P)

  Se P non ha costanti allora scelgo un simbolo di costante a, e pongo $a \in H(P)$

Esempio

- Sia P = $\forall x A(x,c) \vee B(g(c,f(c)),c,a)$, P contiene:
  - i simboli di costante a e c,
  - i simboli di funzione f unaria e g binaria
  - i simboli di predicato A binario e B ternario
- Universo di Herbrand H(P):
  - {a, c, f(a), f(c), g(a,a), g(c,c), g(a,c), g(c,a), f(f(a)), f(g(a,a)), g(a,f(a)), ... }

Formule ground

- **Formule ground**: formule che non contengono variabili.
- **Sostituzione ground**: sostituzione che elimina le variabili.
- **Istanze ground**: formula ottenuta mediante una sostituzione ground delle formula di partenza.

Base di Herbrand

- La **base di Herbrand** di una fbf P, B(P), è l'insieme di tutte le formule atomiche ground che si possono costruire:
  - utilizzando i simboli di predicato che compaiono in P
  - con argomento gli elementi di H(P).

Esempio

- Sia P = $\forall x A(x,c) \vee B(g(c,f(c)),c,a)$
- Universo di Herbrand H(P):
  - {a, c, f(a), f(c), g(a,a), g(c,c), g(a,c), g(c,a), f(f(a)), f(g(a,a)), g(a,f(a)), ... }
- B(P):
  - {A(a,a), A(a, f(a)), A(g(a,a), c), B(a,a,a),...}

Interpretazione di Herbrand

- Sia P una fbf chiusa, $H = (D_H, I_H)$ è un'**interpretazione di Herbrand** per P sse valgono le seguenti condizioni:
  - $D_H = H(P)$
  - $I_H(c) = c$, c simbolo di costante
  - $I_H(f(t_1,\dots,t_n)) = f(t_1,\dots,t_n)$, f simbolo di funzione di arietà n.
- *Osservazione*: non ci sono restrizioni per i simboli di predicato.

Esempio

- $P = \forall x A(x,c) \vee B(g(c,f(c)),c,a)$
- Un'interpretazione di Herbrand di P:
  - $D_H = H(P) = \{a, c, f(a), f(c), g(a,a), g(c,c), g(a,c), g(c,a), f(f(a)), f(g(a,a)), g(a,f(a)), \dots\}$
  - $I_H(a) = a$, $I_H(c) = c$
  - $I_H(f(a)) = f(a)$, $I_H(g(a,c)) = g(a,c)$, ...
  - $I_H(A(x,y)) = \{(x,y) \mid x,y \in D_H \text{ e } x = y\}$
  - $I_H(B(x,y,z)) = \{(x,y,z) \mid x,y,z \in D_H \text{ e } x = g(y,z)\}$

Modello di Herbrand

- Data una fbf P, un **modello di Herbrand** di P è un'interpretazione di Herbrand che la soddisfa.

Esempio

- $P = \forall x A(x,c) \vee B(g(c,f(c)),c,a)$
- Un'interpretazione di Herbrand H di P è:
  - $D_H = H(P) = \{a, c, f(a), f(c), g(a,a), g(c,c), g(a,c), g(c,a), f(f(a)), f(g(a,a)), g(a,f(a)), \dots \}$
  - $I_H(a) = a$, $I_H(c) = c$
  - $I_H(f(a)) = f(a)$, $I_H(g(a,c)) = g(a,c)$, ...
  - $I_H(A(x,y)) = \{(x,y) \mid x,y \in D_H \text{ e } x = y\}$
  - $I_H(B(x,y,z)) = \{(x,y,z) \mid x,y,z \in D_H \text{ e } x = g(y,z)\}$
- H **non** è un modello per P, infatti:
  - $A(x,c)$ è falso per $x = a$
  - $B(g(c,f(c)),c,a)$ è falso poiché $g(c,f(c)) \neq g(c,a)$

Esempio

- P = $\forall x A(x,c) \vee B(g(c,f(c)),c,a)$
- L'interpretazione di Herbrand H' di P:
  - $D_{H'}$ = H'(P) = {a, c, f(a), f(c), g(a,a), g(c,c), g(a,c), g(c,a), f(f(a)), f(g(a,a)), g(a,f(a)), ... }
  - $I_{H'}(a) = a$, $I_{H'}(c) = c$
  - $I_{H'}(f(a)) = f(a)$, $I_{H'}(g(a,c)) = g(a,c)$, ...
  - **$I_{H'}(A(x,y)) = \{(x,y) \mid x,y \in D_{H'}$ e $y = c\}$**
  - $I_{H'}(B(x,y,z)) = \{(x,y,z) \mid x,y,z \in D_{H'}$ e $x = g(y,z)\}$
- H' è un modello di Herbrand per P, infatti:
  - A(x,c) è vero sia per tutti gli x in $D_{H'}$.

Teoremi

- Sia P una fbf chiusa e in forma di Skolem:
  - P è soddisfacibile sse ha un modello di Herbrand

Equivale a dire:

- Sia P una fbf chiusa e in forma di Skolem:
  - P è insoddisfacibile sse non ha un modello di Herbrand

Esempio

- È necessario che la fbf sia in forma di Skolem.
- Sia $P = R(c) \wedge \exists x \sim R(x)$
- è soddisfacibile, infatti ha il seguente modello $M = (D,I)$:
  - $D = \{a,c\}$
  - $I(c) = c$, $I(a) = a$
  - $I(R(x)) = \{x \mid x = c\}$
- Quindi in M, R(c) è vero e R(a) è falso

Esempio cont.

- Ma $P = R(c) \wedge \exists x \sim R(x)$ non ha un modello di Herbrand!
- Tutte le possibili interpretazioni di Herbrand sono formate da:
  - $D_H = H(P) = \{c\}$
  - $I_H(c) = c$
- ho due possibili interpretazioni di R(x):
  - $I_H(R(c)) = 1$ (vero R(c) ma falso $\exists x \sim R(x)$)
  - $I_H(R(c)) = 0$ (vero $\exists x \sim R(x)$ ma falso R(c))
- Entrambi i casi non sono modelli per P.
- Ciò è possibile perché P **non** è in forma di Skolem chiusa!

Espansione di Herbrand

- Sia P = $\forall x_1,\ldots,x_n P'$ una fbf chiusa in forma di Skolem. L'**espansione di Herbrand** di P, E(P), è l'insieme delle formule ground ottenute sostituendo i termini dell'universo di Herbrand di P alle variabili di P' in tutti i possibili modi.
- Ovvero:
  - $E(P) = \{P'[t_1,\ldots,t_n/x_1,\ldots,x_n] \mid t_1,\ldots,t_n \in H(P)\}$.

Esempio

- Sia P = $\forall x \forall y((A(a) \vee A(f(x))) \wedge \sim A(y))$
- H(P) = {a, f(a), f(f(a)), ...}
- E(P) = {
  - x=a, y=a: $(A(a) \vee A(f(a))) \wedge \sim A(a)$
  - x=a, y=f(a): $(A(a) \vee A(f(a))) \wedge \sim A(f(a))$
  - x=f(a), y=a: $(A(a) \vee A(f(f(a)))) \wedge \sim A(a)$
  - x=f(a), y=f(a): $(A(a) \vee A(f(f(a)))) \wedge \sim A(f(a))$
  - x=a, y=f(f(a)): $(A(a) \vee A(f(a))) \wedge \sim A(f(f(a)))$
  - ... }

Teoremi

- Una formula chiusa P in forma di Skolem è soddisfacibile sse E(P) è soddisfacibile.
- **Teorema di Herbrand:**
  - Una formula chiusa P in forma di Skolem è insoddisfacibile sse esiste un sottoinsieme finito di E(P) che è insoddisfacibile.

Algoritmo di Herbrand

- Sia P una fbf chiusa e in forma di Skolem e sia E(P)= $\{P_1, P_2, \ldots\}$ l'espansione di Herbrand di P.
- Algoritmo di Herbrand per P:
  - n=0
  - **repeat**
    - n++

    until $(P_1 \wedge P_2 \wedge \ldots \wedge P_n)$ è insoddisfacibile
  - output "P è insoddisfacibile"

Esempio

- E(P) = {
  - x=a, y=a: $(A(a) \vee A(f(a))) \wedge \sim A(a)$
  - x=a, y=f(a): $(A(a) \vee A(f(a))) \wedge \sim A(f(a))$
  - ...}
- n=1: $(A(a) \vee A(f(a))) \wedge \sim A(a)$ è soddisfacibile

A(a)	A(f(a))	$(A(a) \vee A(f(a))) \wedge \sim A(a)$
0	*0*	0
0	*1*	1
1	*0*	0
1	*1*	0

Esempio

- n=2: $(A(a) \vee A(f(a))) \wedge \sim A(a) \wedge (A(a) \vee A(f(a))) \wedge \sim A(f(a))$ è insoddisfacibile
- Per il teorema di Herbrand
  $\forall xy((A(a) \vee A(f(x))) \wedge \sim A(y))$ è insoddisfacibile

A(a)	A(f(a))	$(A(a) \vee A(f(a))) \wedge \sim A(a) \wedge (A(a) \vee A(f(a))) \wedge \sim A(f(a))$
0	*0*	0
0	*1*	0
1	*0*	0
1	*1*	0

Riassumendo 1:

- Il metodo di Herbrand può essere usato per dimostrare che **P è insoddisfacibile**:

1. Calcolo la forma chiusa di Skolem, P^S, di P
2. Calcolo $H(P^S)$
3. Calcolo $E(P^S)$
4. Applico l'algoritmo di Herbrand a P^S

Riassumendo 2:

- Il metodo di Herbrand può essere usato per dimostrare che **P è valida**:
  - P è valida sse ~P è insoddisfacibile

1. Calcolo ~P
2. Calcolo la forma chiusa di Skolem, $(\sim P)^S$, di ~P
3. Calcolo $H((\sim P)^S)$
4. Calcolo $E((\sim P)^S)$
5. Applico l'algoritmo di Herbrand a $(\sim P)^S$

Riassumendo 3:

- Il metodo di Herbrand può essere usato per dimostrare che **P è conseguenza logica di Γ**:
  - **$\Gamma \models P$ sse $\Gamma \cup \{\sim P\}$ è insoddisfacibile**

1. Calcolo $\Gamma \cup \{\sim P\}$
2. Sia $\Gamma = \{P_1, P_2, \dots, P_n\}$ allora calcolo $Q = \sim P \wedge P_1 \wedge P_2 \wedge \dots \wedge P_n$
3. Calcolo la forma chiusa di Skolem, Q^S, di Q
4. Calcolo $H(Q^S)$
5. Calcolo $E(Q^S)$
6. Applico l'algoritmo di Herbrand a Q^S

Risoluzione nella logica proposizionale

- L'algoritmo di Herbrand è molto inefficiente.
- Esaminiamo ora una procedura molto più efficiente che è alla base della programmazione logica.
- Anche in questo caso usiamo la refutazione.
- Prima vediamo la versione **proposizionale**

Clausole

- Una **clausola** è una disgiunzione finita di zero o più letterali.
- Una clausola che non contiene letterali è detta **clausola vuota**, □.
- Una clausola è **soddisfatta** se ha ameno un letterale vero.
- La clausola vuota è sempre falsa.
- Una clausola è una **tautologia** se contiene un letterale e il suo negato.

Rappresentazione

- La clausole sono rappresentate come insiemi.
- Esempio:
  - la clausola $A \vee B \vee \sim C \vee B \vee \sim D$
  - viene rappresentata con $\{A,B,\sim C,\sim D\}$

Trasformazione

- Ogni fbf **proposizionale** P può essere rappresentata in forma a clausole.
- Trasformazione di P in forma a clausole:
  - P è trasformata in FNC
  - ogni fattore (clausola) è rappresentato con un insieme.
- Esempio: $P = (A \vee B) \wedge (A \vee \sim C \vee \sim D) \wedge (\sim B \vee D)$ diventa $P^c = \{\{A,B\}, \{A,\sim C,\sim D\}, \{\sim B,D\}\}$
- □ = falso,
- l'insieme vuoto di clausole {} = vero

Risolvente

- Siano C_1 e C_2 e R clausole. R è una **risolvente** di C_1 e C_2 se esiste un letterale tale che:
  - $l \in C_1$
  - $\sim l \in C_2$
  - $R = (C_1 - \{l\}) \cup (C_2 - \{\sim l\})$
- Esempio
  - $C_1 = \{A, \sim B\}$ e $C_2 = \{B, C\}$
  - $R = \{A, C\}$

Derivazione

- Sia S un insieme di clausole, **una derivazione** (o prova) **per risoluzione** di S_1 da S è una sequenza $C_1, \ldots, C_m$ di clausole tali che:
  - $C_m = S_1$
  - C_i, $\forall i \in \{1,2,\ldots,m\}$, è
    - una clausola in S oppure
    - un risolvente di due clausole C_j e C_k con $j,k < i$.

Refutazione proposizionale

- Una derivazione di □ da S si dice **refutazione.**
- Esempio:
- S = {{A, ~B} , {B, C}, {~A, C}, {~C}}
  - C_1 = {A, ~B} clausola di S
  - C_2 = {B, C} clausola di S
  - C_3 = {A, C} risolvente di C_1 e C_2
  - C_4 = {~A, C} clausola di S
  - C_5 = {C} risolvente di C_3 e C_4
  - C_6 = {~C} clausola di S
  - C_7 = □ risolvente di C_5 e C_6

Lemma di Risoluzione

- **Lemma di risoluzione:**
  - Siano C_1 e C_2 clausole, e R un loro risolvente, allora R è conseguenza logica di C_1 e C_2.
- **Dim.** Sia v modello per C_1 e C_2, basta dim che v(R)=1. Sia $R = (C_1 - \{l\}) \cup (C_2 - \{\sim l\})$ ci sono due casi:
  - v(l) = 1: allora dato che $v(C_2) = 1$ e $v(\sim l) = 0$ (ovvero non è ~l a rendere vera C_2) segue che $v(C_2 - \{\sim l\}) = 1$ e quindi v(R)=1.
  - v(l) = 0: allora dato che $v(C_1) = 1$ e v(l) = 0 (ovvero non è l a rendere vera C_1) segue che $v(C_1 - \{l\}) = 1$ e quindi v(R)=1.

Teorema di Risoluzione

- **Teorema di Risoluzione**
  - Un insieme di clausole S è insoddisfacibile sse $S \vdash_R \square$.
- La dimostrazione deriva dal lemma di risoluzione.
- **Corollario**
  - $S \models P$ sse $S \cup \{\sim P\} \vdash_R \square$

Refutazione per $S \models P$

- Per verificare che $S \models P$ dobbiamo dimostrare che $S^c \cup (\sim P)^c \vdash_R \square$.
- **Algoritmo di refutazione:** dati S e P
  - scriviamo S e ~P in FNC
  - trasformiamo le FNC negli insiemi di clausole S^c e $(\sim P)^c$
  - troviamo una refutazione di $S^c \cup (\sim P)^c$

Refutazione per $\models P$

- Per calcolare la validità di P ($\models P$) basta refutare ~P (ovvero $(\sim P)^c \vdash_R \square$).
- **Algoritmo di refutazione:** dato P
  - scriviamo ~P in FNC
  - trasformiamo le FNC negli insieme di clausole $(\sim P)^c$
  - troviamo una refutazione di $(\sim P)^c$

Esempio: provare S $\models$ P

- Siano S = {(~A⇒B) ∧ (A⇒B), ~A⇒~B} e P = {A ∧ B} provare che S $\models$ P.
- FNC di S e ~P:
  - FNC(S) = {(A∨B) ∧ (~A∨B), A∨~B}
  - FNC(~P) = {~A ∨ ~B}
- Insiemi di clausole:
  - S^c = {{A, B}, {~A, B}, {A, ~B}}
  - $(\sim P)^c$ = {{~A ,~B}}

Esempio cont.

- Insiemi di clausole:
  - S^c = {{A, B}, {~A, B}, {A, ~B}}
  - $(\sim P)^c$ = {{~A ,~B}}
- Risoluzione per refutazione:
  - C_1 = {A, B} clausola di S^c
  - C_2 = {~A, B} clausola di S^c
  - C_3 = {B} risolvente di C_1 e C_2
  - C_4 = {A, ~B} clausola di S^c
  - C_5 = {~A ,~B} clausola di $(\sim P)^c$
  - C_6 = {~B} risolvente di C_4 e C_5
  - C_7 = □ risolvente di C_3 e C_6

Insieme dei risolventi

- Sia S un insieme di clausole, l'**insieme dei risolventi** di S, Ris(S), è definito come:
  - $Ris(S)= S \cup \{C_{i,j} \mid C_{i,j}$ è la risolvente di $C_i, C_j \in S\}$
- Inoltre
  - $Ris^0(S) = S$
  - $Ris^{n+1}(S) = Ris(Ris^n(S))$ per $n \geq 0$
  - $\mathbf{Ris^*(S)} = Ris^0(S) \cup Ris^1(S) \cup Ris^2(S) \cup \ldots$
- Teorema
  - Un insieme di clausole S è insoddisfacibile sse $\square \in Ris^*(S)$

Algoritmo di risoluzione

Sia P da refutare:

1. calcolare la FNC di ~P
2. calcolare la forma in clausole $S = (\sim P)^c$
3. **repeat**
   R = S
   S = Ris(S) // calcola tutti i risolventi di S
   until ($\square \in$ S or S == R)
4. **if** ($\square \in$ S) output "P è valida"
   else output "P non è valida"

Osservazione

- L'algoritmo di risoluzione nel caso proposizionale ha una strategia che termina e che calcola la validità di P.
- È possibile dimostrare P anche dando una sola derivazione che da ~P porta a □ (algoritmo di refutazione).
- L'algoritmo di risoluzione le prova invece tutte.

Es. algoritmo di risoluzione

$P = \sim A \vee \sim(A \Rightarrow B) \vee (A \wedge B)$

1. calcolare la FNC di ~P:
   - $\sim P = A \wedge (A \Rightarrow B) \wedge (\sim A \vee \sim B)$
   - $FNC(\sim P) = A \wedge (\sim A \vee B) \wedge (\sim A \vee \sim B)$
2. Insieme di clausole:
   - $S = (\sim P)^c = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}\}$

Es. algoritmo di risoluzione

3. repeat (primo ciclo):
   $R = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}\}$
   $S = Ris(S) = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}\} \cup \{\{B\}, \{\sim B\}, \{\sim A\}\}$
   $S \neq R$ e $\square$ non è in S: devo andare avanti.

   repeat (secondo ciclo):
   $R = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}, \{B\}, \{\sim B\}, \{\sim A\}\}$
   $S = Ris(S) = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}, \{B\}, \{\sim B\}, \{\sim A\}\} \cup \{\square\}$
   $\square$ è in S: ho finito
4. output "P è valida"

Es. algoritmo di refutazione

- Invece di calcolare tutti i risolventi diamo direttamente una refutazione per $\sim P$
  - $P = \sim A \vee \sim(A \Rightarrow B) \vee (A \wedge B)$
  - $\sim P = A \wedge (A \Rightarrow B) \wedge (\sim A \vee \sim B)$
  - $FNC(\sim P) = A \wedge (\sim A \vee B) \wedge (\sim A \vee \sim B)$
  - $(\sim P)^c = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}\}$

Es. algoritmo di refutazione

- $(\sim P)^c = \{\{A\}, \{\sim A, B\}, \{\sim A, \sim B\}\}$
- Refutazione per ~P:
  - $C_1 = \{A\}$ clausola di $(\sim P)^c$
  - $C_2 = \{\sim A, B\}$ clausola di $(\sim P)^c$
  - $C_3 = \{B\}$ risolvente di C_1 e C_2
  - $C_4 = \{\sim A, \sim B\}$ clausola di $(\sim P)^c$
  - $C_5 = \{\sim A\}$ risolvente di C_3 e C_4
  - $C_7 = \square$ risolvente di C_1 e C_5

Osservazioni

- Nell'algoritmo di Herbrand per controllare che $(P_1 \wedge P_2 \wedge \ldots \wedge P_n)$ sia insoddisfacibile si può usare il metodo di risoluzione invece delle tabelle di verità.
- Spesso il metodo di risoluzione è più efficiente delle tavole di verità.
- A volte però è necessario un numero di passi esponenziale.

Fondamenti di Logica Matematica
Esercitazione 16

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Esercizio

- **Sia P la fbf:**
  - $\forall x \forall y \forall z (A(f(y),g(y)) \wedge B(h(x,z),z))$
- **determinare**
  - H(P)
  - B(P)
  - E(P)
  - un modello di Herbrand per P

Es 6.2 libro pag. 217

- Sia P una fbf predicativa contenente i simboli di costante a, b, c e il predicato unario A.
- Quante sono le possibili interpretazioni di Herbrand per P?

Esercizio

- Provare, utilizzando l'algoritmo di Herbrand con il metodo di risoluzione al posto delle tavole di verità, che la seguenti formule sono insoddisfacibili:
  - $\forall x\, A(x) \wedge \sim A(f(a))$
  - $\forall x(\sim A(x) \vee B(f(x),x)) \wedge A(g(b)) \wedge \forall y \forall z \sim B(y,z)$
  - $A(a) \wedge \forall x(\sim A(x) \vee B(f(x))) \wedge \sim B(f(a))$
  - $\forall x \sim A(f(x)) \wedge \forall x A(x)$

Es 6.7 libro pag. 218

- **Mostrare che:**
  - A $\wedge$ B$\wedge$ C
- **è una conseguenza logica dell'insieme di clausole:**
  - {{~B,C}, {~A,B}, {A,~C}, {A,B,C}}
- **usando il metodo di risoluzione.**

Es 6.9 libro pag. 219

- **Quali dei seguenti insiemi di clausole sono soddisfacibili?**
- **Per gli insiemi soddisfacibili trovare un'interpretazione che è un modello:**
  - {}
  - {{~A,B}, {A,~B}}
  - {{A,B}, {~A,~B}, {~A,B}}
  - {{~A}, {A,~B}, {B}}
  - {{~A,C}, {~B}, {B}, □}
  - {{A,~B}, {A,B}, {~A}}

Esercizio

- **Mostrare che:**
  - $P = A \Rightarrow C$
- **è una conseguenza logica di :**
  - $S = \{A \Rightarrow (B \Rightarrow C), A \Rightarrow B\}$

Fondamenti di Logica Matematica

Lezione 17

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Metodo di risoluzione:
  - Unificazione
  - Risoluzione nella logica dei predicati

Confronto

- Metodo di Herbrand:
  - costruisce tutte le formule ground
  - applica la risoluzione alle formule proposizionali ottenute.
- Metodo di risoluzione:
  - generalizza l'algoritmo di risoluzione alla logica predicativa facendo il minimo numero di sostituzioni (senza generare tutte le possibili clausole ground).

Quali sostituzioni fare?

- Useremo la tecnica di unificazione
  - usata anche in altri contesti
- Una **sostituzione** σ è un insieme finito, eventualmente vuoto, delle forma $\sigma=\{t_1/x_1, \ldots, t_n/x_n\}$ dove
  - per ogni $i \neq j$, $x_i \neq x_j$.
  - per ogni i, $t_i \neq x_i$.
- La **sostituzione vuota** è indicata con ε.

Sostituzione

- Sia $\sigma = \{t_1/x_1, \ldots, t_n/x_n\}$ una sostituzione ed E un'espressione. $E\sigma$ è l'applicazione di σ a E ottenuta cambiando ogni occorrenza della **variabile** x_i con il **termine** t_i.
- Esempio:
  - $\sigma = \{c/x, g(b)/y, a/z\}$
  - $E = A(x, f(y), z)$
  - $E\sigma = A(c, f(g(b)), a)$

Composizione

- Siano $\sigma=\{v_1/x_1, \ldots, v_n/x_n\}$ e $\vartheta=\{u_1/y_1, \ldots, u_m/y_m\}$ due sostituzioni. La **composizione** di σ e ϑ, $\sigma \circ \vartheta$, si ottiene:
  - cancellando dall'insieme $S = \{v_1\vartheta/x_1,\ldots,v_n\vartheta/x_n\}$, i $v_i\vartheta/x_i$ per i quali $v_i\vartheta = x_i$.
  - cancellando dall'insieme $\vartheta = \{u_1/y_1,\ldots, u_m/y_m\}$, i u_i / y_i per i quali $y_i \in \{x_1, \ldots, x_n\}$.
  - unendo i due insiemi risultanti.
- *Esempio:*
  - $\sigma = \{f(u)/x, b/y, y/z\}$
  - $\vartheta = \{c/u, z/y, b/z\}$
  - $S = \{f(u)\vartheta/x, b\vartheta/y, y\vartheta/z\} = \{f(c)/x, b/y, z/z\}$ e ottengo $\{f(c)/x, b/y\}$ cancellando z/z.
  - Cancello z/y e b/z da $\{c/u, z/y, b/z\}$ e ottengo $\{c/u\}$
  - $\sigma\circ\vartheta = \{f(c)/x, b/y, c/u\}$

Proprietà

- Siano σ_1, σ_2, σ_3 tre sostituzioni ed E un'espressione, allora vale che:
  - $\sigma_1 \circ \varepsilon = \varepsilon \circ \sigma_1 = \sigma_1$
  - $(E\sigma_1)\sigma_2 = E(\sigma_1 \circ \sigma_2)$
  - $(\sigma_1 \circ \sigma_2) \circ \sigma_3 = \sigma_1 \circ (\sigma_2 \circ \sigma_3)$
- La composizione di sostituzioni non è commutativa:
  - In generale: $\sigma_1 \circ \sigma_2 \neq \sigma_2 \circ \sigma_1$

Unificatore

- Una sostituzione σ si dice **unificatore** di un insieme $E = \{E_1, E_2, \ldots, E_n\}$ di espressioni se $E_1\sigma = E_2\sigma = \ldots = E_n\sigma$. Se esiste una unificazione σ di E, E è detto **unificabile**.
- Esempio:
  - $E=\{A(x,a), A(y,a)\}$
  - $\vartheta=\{b/x,b/y\}$, $A(x,a)\vartheta= A(y,a)\vartheta = A(b,a)$
  - $\sigma=\{y/x\}$, $A(x,a)\sigma= A(y,a)\sigma = A(y,a)$
  - σ più generale di ϑ ($\vartheta=\sigma\circ\{b/y\}$)

Mgu

- Un unificatore σ per un insieme $\{E_1, E_2, \ldots, E_n\}$ di espressioni è detto **unificatore più generale** (**mgu**) sse per ogni altro unificatore ϑ dello stesso insieme esiste una sostituzione ρ tale che $\vartheta=\sigma\circ\rho$.
- l'mgu se esiste è unico (a meno di ridenominazione delle variabili)
- Esempio di prima: $\sigma=\{y/x\}$ e $\sigma'=\{x/y\}$ sono mgu.

Insieme di disaccordo

- Sia E un insieme finito di espressioni, l'**insieme di disaccordo** di E, D(E), è così definito:
  - localizzare la posizione P più a sinistra in cui non tutte le espressioni di E hanno lo stesso simbolo
  - estrarre da ogni espressione di E la sottoespressione che comincia con P
  - D(E) è l'insieme di tali sottoespressioni
- Esempio:
  - E = {A(x, f(u)), A(g(y), f(h(z)))} D(E) = {x, g(y)}
  - E = {A(g(y), f(u)), A(g(y), f(h(z)))} D(E) = {u, h(z)}

Osservazioni

- Qualunque unificatore di E unifica D(E).
- $|E\sigma|$ = cardinalità dell'insieme E a cui abbiamo applicato la sostituzione σ.
- Dato un insieme di espressioni E, l'Algoritmo di unificazione calcola l'unificatore più generale (mgu) di E.

Algoritmo di Unificazione

- k=0; $\sigma_k = \varepsilon$;
- repeat
  - if ($|E\sigma_k|$ == 1)
    - then output σ_k; stop;
    - else trova $D(E\sigma_k)$;
  - if (esistono x e t in $D(E\sigma_k)$ t.c. x è una variabile non in t)
    - then $\sigma_{k+1} = \sigma_k \circ \{t/x\}$; k++;
    - else output "E non è unificabile"; stop;
- until ();

Osservazioni

- L'algoritmo può fare diverse scelte per x e t; questo porta alla formazione di mgu diversi (per nomi di variabili).
- La sostituzione si può fare solo se x non occorre in t (si controlla con l'occur check):
  - È necessario per garantire la terminazione dell'algoritmo
  - Es: E = {A(x), A(f(x))} non è unificabile poiché x occorre in f(x).

Quando posso unificare?

	costante $\mathbf{c_2}$	variabile $\mathbf{x_2}$	funzione $\mathbf{f_2}$
costante $\mathbf{c_1}$	se $\mathbf{c_1} = \mathbf{c_2}$	$\{\mathbf{c_1}/\mathbf{x_2}\}$	no
variabile $\mathbf{x_1}$	$\{\mathbf{c_2}/\mathbf{x_1}\}$	$\{\mathbf{x_2}/\mathbf{x_1}\}$ o $\{\mathbf{x_1}/\mathbf{x_2}\}$	$\{\mathbf{f_2}/\mathbf{x_1}\}$ dopo o. c.
funzione $\mathbf{f_1}$	no	$\{\mathbf{f_1}/\mathbf{x_2}\}$ dopo o. c.	se $\mathbf{f_1} = \mathbf{f_2}$ si va ricorsivamente sugli argomenti

Esempi

- a e f(x,y) non sono unificabili
- a e b non sono unificabili
- f(x) e f(g(x)) non sono unificabili (o.c.)
- f(h(x)) e f(h(P(y))) sono unificabili mgu= {P(y)/x}
- x e y sono unificabili mgu = {x/y}

Esempio

- E = {A(f(y,g(v)),h(b)), A(f(h(w),g(a)),t), A(f(h(b),g(v)),t)}, $\sigma_0 = \varepsilon$.
- $|E\sigma_0| > 1$; $D(E\sigma_0)$ = {y, h(w), h(b)}; σ_1 = {h(w)/y}
  - $E\sigma_1$ = {A(f(h(w),g(v)),h(b)), A(f(h(w),g(a)),t), A(f(h(b),g(v)),t)}
- $|E\sigma_1| > 1$; $D(E\sigma_1)$ = {w, b}; $\sigma_2 = \sigma_1 \circ$ {b/w}
  - $E\sigma_2$ = {A(f(h(b),g(v)),h(b)), A(f(h(b),g(a)),t), A(f(h(b),g(v)),t)}
- $|E\sigma_2| > 1$; $D(E\sigma_2)$ = {v, a}; $\sigma_3 = \sigma_2 \circ$ {a/v}
  - $E\sigma_3$ = {A(f(h(b),g(a)),h(b)), A(f(h(b),g(a)),t)}
- $|E\sigma_3| > 1$; $D(E\sigma_3)$ = {h(b), t}; $\sigma_4 = \sigma_3 \circ$ {h(b)/t}
  - $E\sigma_4$ = {A(f(h(b),g(a)),h(b))}
- $|E\sigma_4| = 1$; σ_4 = {h(b)/y, b/w, a/v, h(b)/t}

Risoluzione nella logica del primo ordine

- Ogni formula del primo ordine può essere trasformata in una formula $\forall x_1 x_2 \ldots x_n P$:
  - chiusa
  - senza quantificatori esistenziali
  - con P in FNC
- $\forall x_1 x_2 \ldots x_n (C_1 \wedge C_2 \wedge \ldots \wedge C_n)$

Osservazioni

- $\forall x_1 x_2 \ldots x_n (C_1 \wedge \ldots \wedge C_n)$
- è equivalente a
  - $\forall x_1 x_2 \ldots x_n C_1 \wedge \ldots \wedge \forall x_1 x_2 \ldots x_n C_n$
- Possiamo eliminare i quantificatori e otteniamo $(C_1 \wedge \ldots \wedge C_n)$ che in forma a clausole diventa:
  - $\{C_1, \ldots, C_n\}$

Risolvente

- Siamo C_1, C_2 e R clausole. R è una **risolvente** di C_1 e C_2 se sono verificate le seguenti condizioni:
  - esistono due sostituzioni s_1 e s_2 che ridenominano le variabili in modo che C_1s_1 e C_2s_2 non abbiano variabili in comune.
  - esistono due insiemi di letterali $\{l_1, \ldots, l_m\}$ in C_1s_1 $(m>0)$ e $\{l'_1, \ldots, l'_n\}$ in C_2s_2 $(n>0)$ tali che $L=\{\sim l_1,\ldots, \sim l_m, l'_1, \ldots, l'_n\}$ è unificabile con l'unificatore mgu σ.
  - R ha la forma
    - $((C_1s_1 - \{l_1, \ldots, l_m\}) \cup (C_2s_2 - \{l'_1, \ldots, l'_n\}))\sigma$.

Esempio

- $C_1 = \{\underline{A(z)}, \sim B(y), \underline{A(g(x))}\}$
- $C_2 = \{\underline{\sim A(x)}, \sim C(x)\}$
- $s_1 = \{\}$; $C_1s_1 = \{\underline{A(z)}, \sim B(y), \underline{A(g(x))}\}$
- $s_2 = \{u/x\}$; $C_2s_2 = \{\underline{\sim A(u)}, \sim C(u)\}$
- $L = \{\sim A(z), \sim A(g(x)), \sim A(u)\}$
- $\sigma = \{g(x)/z, g(x)/u\}$
- $R = \{\sim B(y), \sim C(g(x))\}$

Osservazione

- La ridenominazione è necessaria:
  - Es. C_1= {A(x)}, C_2= {~A(g(x))}.
  - senza ridenominazione non si può eseguire l'unificazione (occur check)
  - s_2 = {y/x}; C_2s_2 = {~A(g(y))}.
  - σ = {g(y)/x}
  - R = □.

Osservazione

- Nella logica dei predicati si cancellano più letterali alla volta, ma devono essere tutti unificabili tra loro!
- Non sempre è possibile eliminare un letterale alla volta da C_1 e C_2.
- Es. C_1={A(x), A(y)}, C_2={~A(x)}.
- R = □.
- Se si elimina solo un letterale per volta non si ottiene mai □.

Insieme dei risolventi

- Sia S un insieme di clausole, l'**insieme dei risolventi** di S, Ris(S), è definito come:
  - Ris(S)= S $\cup$ {$C_{i,j}$ |$C_{i,j}$ è la risolvente di C_i, $C_j \in$ S}
- Inoltre
  - $Ris^0(S) = S$
  - $Ris^{n+1}(S) = Ris(Ris^n(S))$ per $n \geq 0$
  - **$Ris^*(S)$** = $Ris^0(S) \cup Ris^1(S) \cup Ris^2(S) \cup \ldots$

Teorema di risoluzione

- **Teorema di risoluzione:**
  - Un insieme di clausole S è insoddisfacibile sse $S \vdash_R \square$.
- Oppure
  - Un insieme di clausole S è insoddisfacibile sse $\square \in Ris^*(S)$

Algoritmo di refutazione

- Sia P da refutare:
1. negare P
2. trasformare ~P in clausole
3. $S = \{(\sim P)^c\}$
4. repeat

   S = Ris(S)
5. until ($\square \in S$)
6. output "P è insoddisfacibile"

Esempio

- L'algoritmo può non terminare.
- $S = \{\{A(0)\}, \{\sim A(x), A(s(x))\}\}$

- $Ris^1(S) = S \cup \{\{A(s(0))\}\}$
- $Ris^2(S) = Ris^1(S) \cup \{\{A(s(s(0)))\}\}$
- $Ris^3(S) = Ris^2(S) \cup \{\{A(s(s(s(0))))\}\}$
- …

Raffinamenti

- Il metodo di risoluzione
  - è più efficiente dell'algoritmo di Herbrand,
  - ma può generare clausole ridondanti e irrilevanti.
- Si possono effettuare delle semplificazioni.

Semplificazioni

- Eliminazione delle tautologie, {A, ~A}.
- Eliminazione delle clausole già generate.
- Eliminazione delle clausole comprese in altre:
  - Esempio: {B} e {A, B}
    - {B} comprende {A, B}
    - eliminazione di {A, B}.

Fondamenti di Logica Matematica
Lezione 18

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Programmazione logica
  - clausole di Horn
  - risoluzione SLD
- Web semantico
  - introduzione

Risoluzione lineare

- Una prova per **risoluzione lineare** di C da un insieme di clausole S è una sequenza $C_1,\ldots, C_n$ tale che
  - $C_1 \in S$
  - $C_n = C$
  - $\forall i = 2, \ldots, n$ risulta:
    - C_i è la risolvente di C_{i-1} e B_{i-1} (clausola laterale), con $B_{i-1} \in S$ o $B_{i-1} = C_j$ con $j<i$.
- Osservazione: a ogni passo si utilizza la risolvente del passo precedente.

Esempio

- $S = \{\{A, B\}, \{A, \sim B\}, \{\sim A, B\}, \{\sim A, \sim B\}\}$
- $C_1 = \{A, B\}$, $B_1 = \{A, \sim B\}$
- $C_2 = \{A\}$, $B_2 = \{\sim A, B\}$
- $C_3 = \{B\}$, $B_3 = \{\sim A, \sim B\}$
- $C_4 = \{\sim A\}$, $B_4 = \{A\}$
- $C_5 = \square$

Completezza

- **Teorema.** Se un insieme di clausole S è insoddisfacibile, allora la clausola vuota è derivabile dalla risoluzione lineare
- La risoluzione lineare è completa per refutazione
- Dobbiamo però "ricordarci" tutti i risolventi calcolati fino ad ora
- C'è un'altra risoluzione che risolve questo problema: risoluzione da input

Risoluzione di input

- Una prova per **risoluzione di input** di C da un insieme di clausole S, è una sequenza $C_1,\ldots, C_n$ tale che $C_n = C$ e, ad ogni passo, una delle clausole risolventi è un'istanza di un elemento di S. Ovvero:
  - $C_1 \in S$
  - $C_n = C$
  - C_i è la risolvente di C_{i-1} e B_{i-1} (clausola laterale), con $B_{i-1} \in S$
- Esempio:
  - Es S={{A, B}, {A, ~B}, {~A, B}, {~A, ~B}} è refutabile ma non usando la risoluzione da input.

Esempio

- S={{A, B}, {A, ~B}, {~A, B}, {~A, ~B}}
- C_1 = {A, B}, B_1 = {A, ~B}
- C_2 = {A}, B_2 = {~A, B}
- C_3 = {B}, B_3 = {~A, ~B}
- C_4 = {~A}, B_4 = {A, ~B}
- C_5 = {~B}, B_5 = {A, B}
- C_6 = {A}, B_6 = {~A, ~B}
- C_7 = {~B}, B_7 = {~A, B}
- C_8 = {~A}, B_8 = {A, B}
- C_9 = {B}, B_9 = {A, ~B}
- C_{10} = {A}, B_{10} = {..., ...}
- ... non arrivo mai a □

Perché non ho nessuna clausola in S con un solo elemento

Programmazione logica

- Per la dimostrazione automatica dei teoremi sono necessarie strategie efficienti.
- La refutazione lineare da input è
  - particolarmente efficiente
  - ma non completa per refutazione.
- Se ci restringiamo a particolari clausole (clausole di Horn) la risoluzione da input diventa completa per refutazione.

Clausole di Horn

- Una **clausola di Horn** è una clausola che contiene al più una formula atomica positiva
- Esempio sono clausole di Horn:
  - {A, ~B},
  - {~A, ~B, ~C, D},
  - {A},
  - {~A, ~B},

Programmi logici

- $\{A_0, \sim A_1, \sim A_2, \ldots, \sim A_n\}$ è detta **clausola di Horn definita.**
- $\{\sim A_0, \sim A_1, \ldots, \sim A_n\}$ è detta **clausola goal**
- **Notazione:**
  - clausola definita: A_0:-$A_1, A_2, \ldots, A_n$
  - clausola goal: ?-$A_1, A_2, \ldots, A_n$
- Un **programma logico** è un insieme finito di clausole definite.

Risoluzione SLD

- Una **prova per risoluzione SLD** (Linear resolution for Definite clauses with Selection function) è una derivazione per risoluzione lineare di input tale che:
  - ogni risolvente laterale è una clausola di Horn definita,
  - ogni altro risolvente è una clausola goal.
- La risoluzione SLD è completa per clausole di Horn.
- Una **refutazione SLD** è una prova per risoluzione SLD nella quale l'ultimo risolvente è la clausola vuota.

Regola di selezione

- Supponiamo di avere la clausola goal
  - ?- B(x,y), C(y,z)
- e di poter unificare sia B(x,y) che C(y,z) con delle clausole del programma P.
- Quale unifichiamo per prima?
- Quale scegliere ci viene detto dalla **regola di selezione**.
- In Prolog generalmente si usa la regola **lefmost**:
  - si sceglie la formula atomica che può essere unificata più a sinistra.

Esempio

- Sia il programma logico:
  - A(x,z) :- B(x,y), C(y,z)
  - C(x,y) :- B(x,y)
  - C(x,y) :- D(x,y)
  - B(a,b)
  - D(b,c)
- Si dimostri la clausola goal:
  - ?- A(a,x)
- usando la regola leftmost.

Esempio

?- A(a,x) $A(x_0,z_0)$:- $B(x_0,y_0)$, $C(y_0,z_0)$

$\sigma_0 = \{a/x_0, x/z_0\}$

?- $B(a,y_0)$, $C(y_0,x)$ B(a,b)

$\sigma_1 = \{b/y_0\}$

?- C(b,x) $C(x_2,y_2)$:- $D(x_2,y_2)$

$\sigma_2 = \{b/x_2, x/y_2\}$

?- D(b,x)   D(b,c)

$\sigma_3 = \{c/x\}$

$\square$

Sostituzione di risposta

- Siano un programma logico P e un goal G. La **sostituzione di risposta** per $P \cup \{G\}$ è la composizione:
  - $\sigma = \sigma_0 \circ \sigma_1 \circ \ldots \circ \sigma_n$ delle sostituzioni mgu della risoluzione LSD di $P \cup \{G\}$,
  - ristretta alle sole variabili di G.

Esempio

- Nell'esempio volevamo dimostrare:
  - ?- A(a,x)
- Per quale x, A(a,x) è vera?
- Ci aiutano le sostituzioni fatte:
  - $\sigma_0 = \{a/x_0, x/z_0\}$
  - $\sigma_1 = \{b/y_0\}$
  - $\sigma_2 = \{b/x_2, x/y_2\}$
  - $\sigma_3 = \{c/x\}$
- Basta calcolare $\sigma_0 \circ \sigma_1 \circ \sigma_2 \circ \sigma_3$
  - $\sigma_0 \circ \sigma_1 \circ \sigma_2 \circ \sigma_3 = \{a/x_0, c/z_0, b/y_0, b/x_2, c/y_2, c/x\}$
  - La **sostituzione di risposta** è $\{c/x\}$

Programmi logici

- Significato di clausola definita**:**
  - A_0:-A_1, A_2, …, A_n
  - $A_0 \vee \sim A_1 \vee \sim A_2 \vee \ldots \vee \sim A_n \equiv$
  - $A_0 \vee \sim(A_1 \wedge A_2 \wedge \ldots \wedge A_n) \equiv$
  - $(A_1 \wedge A_2 \wedge \ldots \wedge A_n) \Rightarrow A_0$
- Significato di clausola goal:
  - ?-A_1, A_2, …, A_n
  - $\sim A_1 \vee \sim A_2 \vee \ldots \vee \sim A_n \equiv$
  - $\sim(A_1 \wedge A_2 \wedge \ldots \wedge A_n)$
  - ovvero vogliamo refutare $A_1 \wedge A_2 \wedge \ldots \wedge A_n$

Tipi di derivazione

- Una derivazione SLD può essere di tre tipi:
  - **refutazione** (finisce con la clausola vuota).
  - **derivazione con fallimento** (finisce con una clausola non vuota che non può essere ulteriormente risolta).
  - **derivazione infinita**.

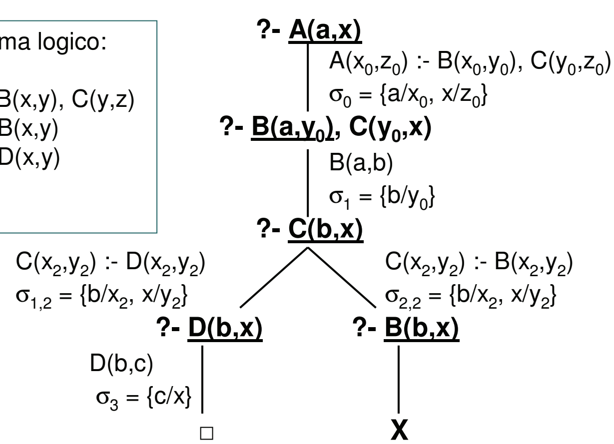
Alberi SLD

- Fissata una regola di selezione, sia P un programma logico e G una clausola goal. L'**albero SLD** di P∪{G} è un albero (eventualmente infinito) tale che:
  - la radice contiene G
  - ogni altro nodo contiene una clausola C_i che è la risolvente di C_j con una clausola in P. Dove C_j è la clausola contenuta del nodo padre di C_i.

Esempio

Programma logico:

A(x,z) :- B(x,y), C(y,z)
C(x,y) :- B(x,y)
C(x,y) :- D(x,y)
B(a,b)
D(b,c)



Esempio

- Sapendo che:
  - Un genitore è un avo
  - Un genitore di un genitore è a sua volta un avo
  - Un avo è un avo
  - Barbara è genitore di Anna
  - Davide è genitore di Barbara
  - Elena è genitore di Barbara
  - Federico è genitore di Davide
- Dimostrare che
  - Barbara ha un avo ed è genitore di qualcuno

Esempio

- Formalizzazione:
  - $\forall x \forall y\ (G(x,y) \Rightarrow A(x,y))$
  - $\forall x \forall y \forall z\ ((G(x,z) \wedge G(z,y)) \Rightarrow A(x,y))$
  - $\forall x \forall y\ (A(x,y) \Rightarrow A(x,y))$
  - $G(b,a)$
  - $G(d,b)$
  - $G(e,b)$
  - $G(f,d)$

  - $\exists y\ A(y,b) \wedge \exists z\ G(b,z)$

Forma a clausole

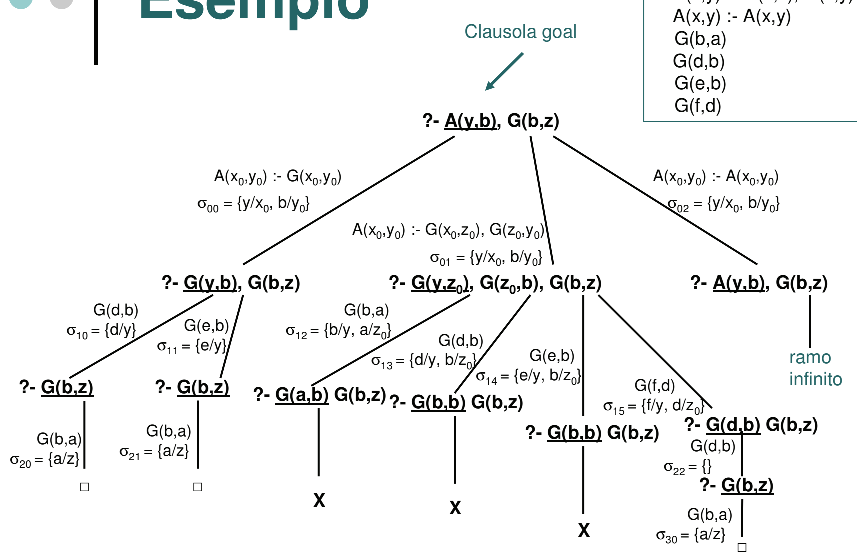
Programma logico

- $\forall x \forall y\ (G(x,y) \Rightarrow A(x,y))$
  - $\forall x \forall y\ (\sim G(x,y) \vee A(x,y))$
  - $\sim G(x,y) \vee A(x,y)$
  - **A(x,y) :- G(x,y)**
- $\forall x \forall y \forall z\ ((G(x,z) \wedge G(z,y)) \Rightarrow A(x,y))$
  - $\forall x \forall y \forall z\ (\sim(G(x,z) \wedge G(z,y)) \vee A(x,y)) \equiv \forall x \forall y \forall z\ (\sim G(x,z) \vee \sim G(z,y) \vee A(x,y))$
  - $(\sim G(x,z) \vee \sim G(z,y) \vee A(x,y))$
  - **A(x,y) :- G(x,z), G(z,y)**
- $\forall x \forall y\ (A(x,y) \Rightarrow A(x,y))$
  - $\forall x \forall y\ (\sim A(x,y) \vee A(x,y))$
  - $\sim A(x,y) \vee A(x,y)$
  - **A(x,y) :- A(x,y)**
- **G(b,a)**
- **G(d,b)**
- **G(e,b)**
- **G(f,d)**

- $\exists y\ A(y,b) \wedge \exists z\ G(b,z)$
  - lo nego per refutarlo $\sim(\exists y\ A(y,b) \wedge \exists z\ G(b,z)) \equiv \forall y \forall z\ (\sim A(y,b) \vee \sim G(b,z))$
  - $(\sim A(y,b) \vee \sim G(b,z))$
  - Clausola goal **?- A(y,b), G(b,z)**

Esempio

Programma logico:
A(x,y) :- G(x,y)
A(x,y) :- G(x,z), G(z,y)
A(x,y) :- A(x,y)
G(b,a)
G(d,b)
G(e,b)
G(f,d)



Esempio

- Sostituzioni di risposta, ci sono tre rami che portano alla clausola vuota:
  - $\sigma_{00} \circ \sigma_{10} \circ \sigma_{20} = \{d/x_0, b/y_0, d/y, a/z\}$
  - $\sigma_{00} \circ \sigma_{11} \circ \sigma_{21} = \{e/x_0, b/y_0, e/y, a/z\}$
  - $\sigma_{01} \circ \sigma_{15} \circ \sigma_{22} \circ \sigma_{30} = \{f/x_0, b/y_0, d/z_0, f/y, a/z\}$

- Le **sostituzioni di risposta** sono:
  - {d/y, a/z}
  - {e/y, a/z}
  - {f/y, a/z}

Infatti Barbara ha 3 avi e un figlio!

Applicazioni della logica

- Abbiamo visto le basi della logica matematica e della programmazione logica
- Ora passiamo alle applicazioni nell'informatica:
  - Semantic web (cenni)
  - Logica di BAN (per dimostrare le proprietà dei protocolli di autenticazione)
  - OBDD (strutture dati per la rappresentazione efficiente di funzioni booleane)
  - Logica fuzzy

Fondamenti di Logica Matematica
Esercitazione 19

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Es 6.13 libro pag. 219

- **Applicare l'algoritmo di unificazione a ciascun dei seguenti insiemi per trovare un mgu o mostrare che non esiste:**
  - {A(x,y), A(y,f(z))}
  - {B(a,y,f(y)), B(z,z,u)}
  - {A(x,g(x)), A(y,y)}
  - {B(x,g(x),y), B(z,u,g(a)), B(a,g(a),v)}
  - {A(g(x), y), A(y,y), A(y,f(u))}

Es 6.14 libro pag. 219

- **Trovare i risolventi delle seguenti clausole:**
  - {A(x,y), A(y,z)}, {~A(u,f(u))}
  - {B(x,x), ~C(x,f(x))}, {C(x,y), D(y,z)}
  - {A(x,y), ~A(x,x), B(x,z,f(x))}, {~B(f(x),z,x), A(x,z)}
- Quali di queste clausole sono clausole di Horn?

Es 6.16 libro pag. 219

- **Provare utilizzando la risoluzione che da:**

1. $\forall x \forall y \forall z((A(x,y) \wedge A(y,z)) \Rightarrow A(x,z))$
2. $\forall x \forall y(A(x,y) \Rightarrow A(y,x))$

- **segue** $\forall x \forall y \forall z((A(x,y) \wedge A(z,y)) \Rightarrow A(x,z))$

Esercizio

- Dimostrare tramite risoluzione che:

$\forall x(A(x) \Rightarrow \exists y B(x,f(y))) \wedge \forall x \forall y(B(x,y) \Rightarrow B(x,f(y))) \wedge A(c) \vDash \exists x \exists y B(x,f(f(y)))$

Esercizio

- Sia il programma logico:
  - Q(x,y) :- P(f(x),y), P(x,x)
  - P(f(c),a):- P(c,b)
  - P(c,c)
  - P(f(a),x)
  - P(x,b)
- Costruire l'albero SLD per il goal ?-Q(x,a) usando la regola di selezione leftmost, e calcolare le sostituzioni di risposta.

Fondamenti di Logica Matematica

Lezione 22 (prima parte)

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Contenuti della lezione

- Introduzione alla logica fuzzy:
  - Insiemi fuzzy
  - Variabili linguistiche
  - Operazioni su insiemi fuzzy
  - Logica fuzzy
  - Connettivi

Logica bivalente

- Logica classica:
  - **principio di non contraddizione**:
    - x non può appartenere sia ad A che al suo complemento A^c.
    - $A \cap A^c = \varnothing$
    - $A \wedge \sim A = F$
  - **principio del terzo escluso**:
    - se x non appartiene ad A allora appartiene al suo complemento A^c.
    - $A \cup A^c = U$ (U = universo)
    - $A \vee \sim A = T$

Esempio

- Persone alte / persone basse
- 1,92m: è alto
- 1,50m: è basso
- 1,71m: ???
- Logica classica:
  - valore soglia Es. soglia=1,70m
  - sotto la soglia: bassi
  - sopra la soglia: alti
- Non è però soddisfacente!

Logica fuzzy

- Insieme fuzzy:
  - È un insieme di oggetti nel quale non c'è un confine ben preciso tra gli oggetti che vi appartengono e quelli che non vi appartengono.
  - Ogni elemento x dell'universo ha un valore di appartenenza all'insieme A. Il valore è un reale nell'intervallo [0,1]:
    - 0 = non appartenenza
    - 1 = appartenenza completa
    - 0,3 = appartenenza parziale

Esempio

- A = Insieme delle persone alte
- 1,92m: appartiene a A con grado di appartenenza 0,95
- 1,50m: appartiene a A con grado di appartenenza 0,15
- 1,71m: appartiene a A con grado di appartenenza 0,62

Fuzziness e probabilità

- L'appartenenza ad un insieme fuzzy è un concetto diverso dalla probabilità.
- Esempio:
- Maria ha un grado di appartenenza alle donne alte di 0,7:
  - non è una probabilità.
  - l'altezza è un concetto fuzzy.
- Maria ha la probabilità di laurearsi quest'anno pari al 70%:
  - riguarda l'incertezza dell'evento.
  - l'evento (laurearsi) non è fuzzy in quanto ci si laurea o no (è un concetto bivalente).

Concetti fuzzy

- Concetti fuzzy:
  - bello
  - giovane
  - amico
  - vicino
- Concetti bivalenti:
  - sposato
  - maschio
  - fratello
  - laureato

Insiemi fuzzy

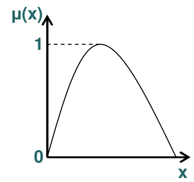
- Un **insieme fuzzy** è un insieme di coppie ordinate:
  - A = {(x, μ(x))}
  - dove μ(x) ∈ [0,1] è il grado di appartenenza di x all'insieme A.
- Se A è un **insieme ordinario**, μ può avere solo due valori:
  - μ(x) = 0 se x non appartiene ad A
  - μ(x) = 1 se x appartiene ad A

Funzione di appartenenza

- Dato un insieme fuzzy A
  - ogni elemento dell'universo U è membro di A con un certo grado di appartenenza (che può essere anche 0).
- L'insieme degli elementi con funzione di appartenenza ad A diversa da zero è chiamato **supporto** di A.

Rappresentazioni di μ

- Una funzione di appartenenza può essere:
  - **Continua**
    - Es:



  - **Discreta**
    - Es: U = [0; 10; 20; 40]
    - μ(x) = [0,1; 1; 0,3; 0]

Variabili linguistiche

- Una **variabile algebrica** ha come valori dei numeri.
- Una **variabile linguistica** ha come valori delle parole o delle frasi.
- L'**insieme dei termini** è l'insieme dei valori che può avere una variabile linguistica.
- Gli elementi nell'insieme dei termini sono **insiemi fuzzy.**

Esempio

- Sia x la variabile linguistica “età”.
- L’insieme dei termini è,
  - T= {Vecchio, MoltoVecchio, NonCosìVecchio, PiùOMenoGiovane, MoltoGiovane, Giovane}
  - Tutti i termini sono insiemi fuzzy.
    - Es. Vecchio è un insieme fuzzy.

Termini

- Un termine può essere di due tipi:
  - **Primario**: è un termine definito a priori.
    - Es: Vecchio, Giovane.
  - **Modificato**: è un termine che deriva da un termine primario tramite un modificatore.
    - Es: MoltoGiovane, NonCosìVecchio.

Operazioni su insiemi

- Dato un universo U. Siano i due insiemi fuzzy:
  - $A = \{(x, \mu_A(x)), x \in U\}$
  - $B = \{(x, \mu_B(x)), x \in U\}$
- allora
  - $A \cap B = \{(x, \min(\mu_A(x), \mu_B(x))), x \in U\}$
  - $A \cup B = \{(x, \max(\mu_A(x), \mu_B(x))), x \in U\}$
  - $A^c = \{(x, 1 - \mu_A(x)), x \in U\}$

Sottoinsiemi fuzzy

- Dato un universo U. Siano i due insiemi fuzzy:
  - $A = \{(x, \mu_A(x)), x \in U\}$
  - $B = \{(x, \mu_B(x)), x \in U\}$
- allora
  - $B \subseteq A$ sse $\forall x \in U, \mu_B(x) \leq \mu_A(x)$
  - $A \cup B = \{(x, \max(\mu_A(x), \mu_B(x))), x \in U\} = \{(x, \mu_A(x)), x \in U\} = A$

Esempio

- Una famiglia di 4 persone vuole comprare una casa.
- Criteri di scelta:
  1. confort derivante dal numero di camere da letto.
  2. grandezza della casa.
- Universo: case disponibili denominate con il numero di camere da letto:
  - U = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.

Esempio

- Universo:
  - U = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10]
- Il confort è un insieme fuzzy con i seguenti gradi di appartenenza:
  - C = [0,2; 0,5; 0,8; 1; 0,7; 0,3; 0; 0; 0; 0].
- La grandezza della casa è un insieme fuzzy con i seguenti gradi di appartenenza:
  - G = [0; 0; 0,2; 0,4; 0,6; 0,8; 1; 1; 1; 1].

Esempio

- C = [0,2; 0,5; 0,8; 1; 0,7; 0,3; 0; 0; 0; 0]
- G = [0; 0; 0,2; 0,4; 0,6; 0,8; 1; 1; 1; 1]
- C∩G = [0; 0; 0,2; 0,4; 0,6; 0,3; 0; 0; 0; 0]
  - L'ottimo è una casa con 5 camere da letto!
- C∪G = [0,2; 0,5; 0,8; 1; 0,7; 0,8; 1; 1; 1; 1]
  - una casa grande **o** confortevole ha 4, 7 o più stanze.
- G^c = [1; 1; 0,8; 0,6; 0,4; 0,2; 0; 0; 0; 0]

Osservazione

- Non valgono più i principi della logica bivalente.
- Principio di non contraddizione:
  - $A \cap A^c \neq \emptyset$
- Principio del terzo escluso:
  - $A \cup A^c \neq U$ (U = universo)

Esempio

- $G = [\ 0;\ \ 0; 0{,}2; 0{,}4; 0{,}6; 0{,}8; 1; 1; 1; 1]$
- $G^c = [\ 1;\ \ 1; 0{,}8; 0{,}6; 0{,}4; 0{,}2; 0; 0; 0; 0]$

- $G \cap G^c \neq \emptyset$
  - $G \cap G^c = [\ 0;\ \ 0; 0{,}2; 0{,}4; 0{,}4; 0{,}2; 0; 0; 0; 0]$
- $G \cup G^c \neq U$
  - $G \cup G^c = [\ 1;\ \ 1; 0{,}8; 0{,}6; 0{,}6; 0{,}8; 1; 1; 1; 1]$

Modificatori

- I **modificatori** sono operazioni che modificano il significato di un termine.
- Esempio:
  - insieme fuzzy “alto”
  - modificatore “molto” crea un nuovo insieme fuzzy “molto alto”.
- Modificatori:
  - molto, poco, quasi, sicuramente, ecc.

Modificatori

- È possibile approssimare i modificatori con delle funzioni matematiche:
- molto A = A^2
- più o meno A = $A^{1/2}$
- estremamente A = A^3
- esattamente A = A^∞ (considera solo gli 1)
- Es: U=[Anna; Paolo; Lucia; Marco; Carla]
  - giovane = [1; 0,6; 0,1; 0; 0]
  - molto giovane = [1; 0,36; 0,01; 0; 0]
  - esattamente giovane = [1; 0; 0; 0; 0]

Logica fuzzy

- Nella logica fuzzy una proposizione può essere **vera**, **falsa** o **forse vera**.
- Abbiamo quindi una logica a più valori.
- Usiamo un dominio discreto:
  - {0; 0,5; 1}
- È possibile considerare anche altri valori intermedi.
  - Es: {0; 0,25; 0,5; 0,75; 1}

Negazione

- $v(\sim P) = 1 - v(P)$
- Rappresentazione con la tavola di verità:

P	~P
0	1
0,5	0,5
1	0

Congiunzione

- $v(P \wedge Q) = \min(v(P), v(Q))$
- Rappresentazione con la tavola di verità:

P	Q	P ∧ Q
0	0	0
0	0,5	0
0	1	0
0,5	0	0
0,5	0,5	0,5
0,5	1	0,5
1	0	0
1	0,5	0,5
1	1	1

Disgiunzione

- $v(P \vee Q) = \max(v(P), v(Q))$
- Rappresentazione con la tavola di verità:

P	Q	$P \vee Q$
0	0	0
0	0,5	0,5
0	1	1
0,5	0	0,5
0,5	0,5	0,5
0,5	1	1
1	0	1
1	0,5	1
1	1	1

Implicazione

- Nella logica classica:
  - $P \Rightarrow Q \equiv \sim P \vee Q$
- Non va bene per la logica fuzzy.
- Si usa spesso l'implicazione di Gödel:
  - $P \Rightarrow Q \equiv (P \leq Q) \vee Q$

Fondamenti di Logica Matematica

Esercitazione 23 (prima)

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Esercizio

- **Sia il programma logico:**
  - A(a,b)
  - A(b,a)
  - B(b,c)
  - B(a,d)
  - C(x,y):-B(w,x), A(w,z), B(z,y)
- **Si dimostri il goal**
  - ?- C(x,y)
- **tramite l'albero SLD, con regola leftmost, e si diano le sostituzioni di risposta.**

Esercizio

- Dimostrare tramite risoluzione che:

$\forall x(A(x) \Rightarrow \exists y B(x,f(y))) \wedge \forall x \forall y(B(x,y) \Rightarrow B(x,f(y))) \wedge A(c) \models \exists x \exists y B(x,f(f(y))))$

Fondamenti di Logica Matematica

Esercitazione 23 (seconda)

Valentina Ciriani
DTI - Università di Milano
www.dti.unimi.it/~ciriani

Esercizio 1

- **Sia U=[1;2;3;4;5;6;7;8;9] un universo calcolare l'unione l'intersezione e il complemento delle seguenti coppie di insiemi fuzzy:**
  - A=[0;0,3;0,6;1;0;0,7;0,5;0;0,9]
    B=[1;0,4;0;1;0,8;0,2;0,34;0,99;0,1]
  - A=[0,7;0,44;0,1;0,7;1;0,7;0,33;0,8;0,11]
    B=[0,7;0,33;1;0,6;0,3;0;0,4;0,3;0,12]

Esercizio 2

- **Sia U=[1;2;3;4;5;6;7;8;9] un universo calcolare: molto A = A^2, più o meno A = $A^{1/2}$, esattamente A (solo gli 1) per i seguenti insiemi fuzzy:**
  - A=[0;0,3;0,6;1;0;0,7;0,5;0;0,9]
  - A=[1;0,4;0;1;0,8;0,2;0,34;0,99;0,1]
  - A=[0,7;0,44;0,1;0,7;1;0,7;0,33;0,8;0,11]
  - A=[1;0,33;1;0,6;0,3;0;0,4;0,3;0,12]

Esercizio 3

- **Supponendo di avere il dominio discreto {0; 0,3; 0,7; 1} scrivere le tabelle di verità per ~A, A∧B e A∨B**

- **Supponendo di avere il dominio discreto {0; 0,25; 0,5; 0,75; 1} scrivere le tabelle di verità per ~A, A∧B e A∨B**