Stochastic Logic Programs

Johannes Lerch

Seminar aus maschinellem Lernen

Agenda

- Motivation
- Definition
- Wahrscheinlichkeitsverteilungen
- Anwendungsbeispiele
- Parameter Estimation

Kontextfreie Sprachen

- Ausdruck:

  $a^n b^n$

- Grammatik:

  $S \rightarrow \emptyset$

  $S \rightarrow aSb$

Transformation zu Prolog

- Grammatik G:

  $S \rightarrow \emptyset$

  $S \rightarrow aSb$

- Prolog:

```
s([]).
s(S) :- a(A),  append(A,X,S),  b(B), append(S1,B,X),  s(S1).
a([a]).
b([b]).
```

Stochastisch kontextfreie Grammatik

- Grammatik G:

  $0.5: S \rightarrow \emptyset$

  $0.5: S \rightarrow aSb$

$P(\emptyset|G) = 0.5$

$P(ab|G) = 0.25$

$P(aabb|G) = 0{,}125$

Stochastisch Logische Programme

- Modellierung einer Münze:

  coin(head).

  coin(tail).

- Eigentlich gemeint:

  0.5: coin(head).

  0.5: coin(tail).

Stochastisch Logische Programme

- Definition von SLP:
  - Eindeutig definiertes logisches Programm
  - Klauseln mit positiven Zahlen parametrisiert
- Pure SLP:
  - Alle Klauseln parametrisiert
- Impure SLP:
  - Nicht alle Klauseln parametrisiert
- Normalised SLP:
  - Parameter aller Klauseln mit gleichem Head summieren zu 1
    → Parameter entsprechen Wahrscheinlichkeiten

Beispiele für SLP

0.5: coin(head).
0.5: coin(tail).

1/6: roll_dice(1).
1/6: roll_dice(2).
1/6: roll_dice(3).
1/6: roll_dice(4).
1/6: roll_dice(5).
1/6: roll_dice(6).

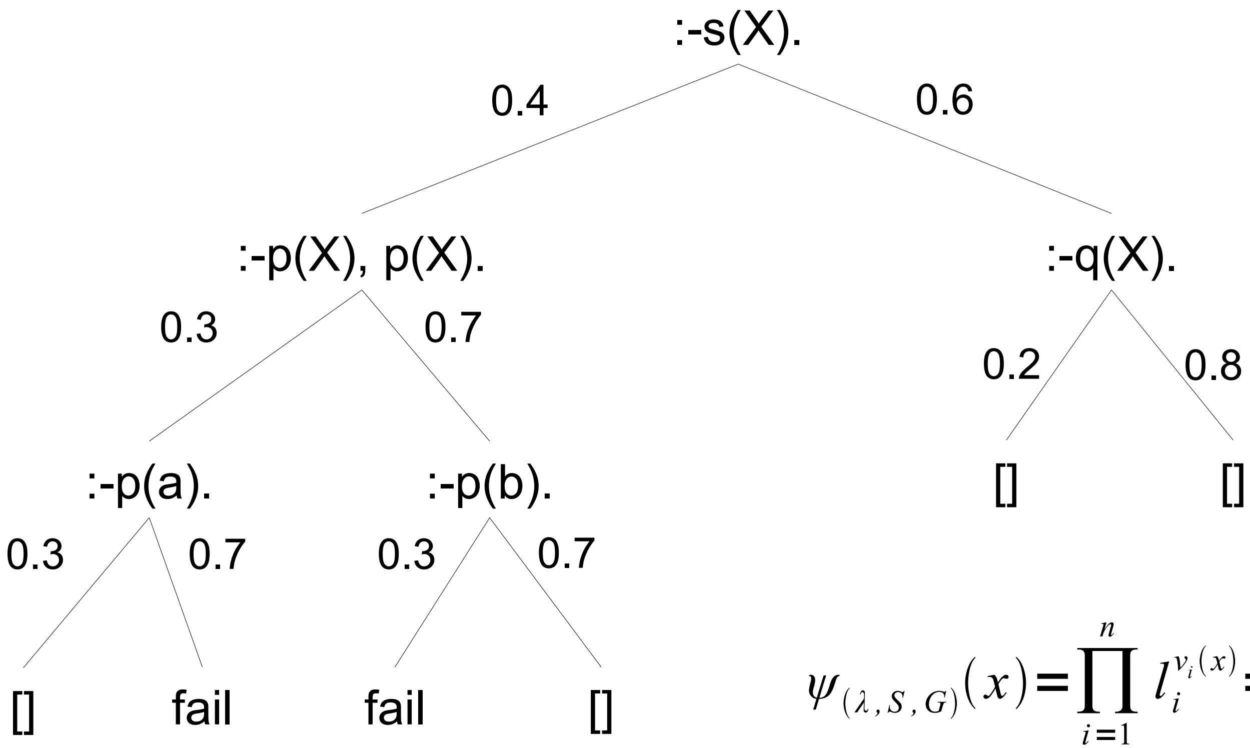
Wahrscheinlichkeitsverteilung auf Ableitungen

TECHNISCHE UNIVERSITÄT DARMSTADT

- $S = \{l_1{:}C_1, \ldots, l_n{:}C_n\}$
- $\lambda = (\ln l_1, \ldots, \ln l_n)$
- D(G) = {x: x Ableitung, die bei G beginnt}
- $v_i(x)$ Anzahl Anwendungen C_i in x

$$\forall\, x \in D(G): \qquad \psi_{(\lambda, S, G)}(x) = \prod_{i=1}^{n} l_i^{v_i(x)} = e^{\lambda v(x)}$$

Wahrscheinlichkeitsverteilung auf Ableitungen



C_1 0.4: s(X) :- p(X),p(X).
C_2 0.6: s(X) :- q(X).
C_3 0.3: p(a).
C_4 0.7: p(b).
C_5 0.2: q(a).
C_6 0.8: q(b).

$$\psi_{(\lambda, S, G)}(x) = \prod_{i=1}^{n} l_i^{v_i(x)} = e^{\lambda v(x)}$$

$$\psi_{(\lambda, S, :-s(X))}(C_1 C_3 C_3) = 0.4 * 0.3 * 0.3 = 0.036$$

$$\psi_{(\lambda, S, :-s(X))}(C_2 C_5) = 0.6 * 0.2 = 0.12$$

Wahrscheinlichkeitsverteilung auf Beweisen

TECHNISCHE UNIVERSITÄT DARMSTADT

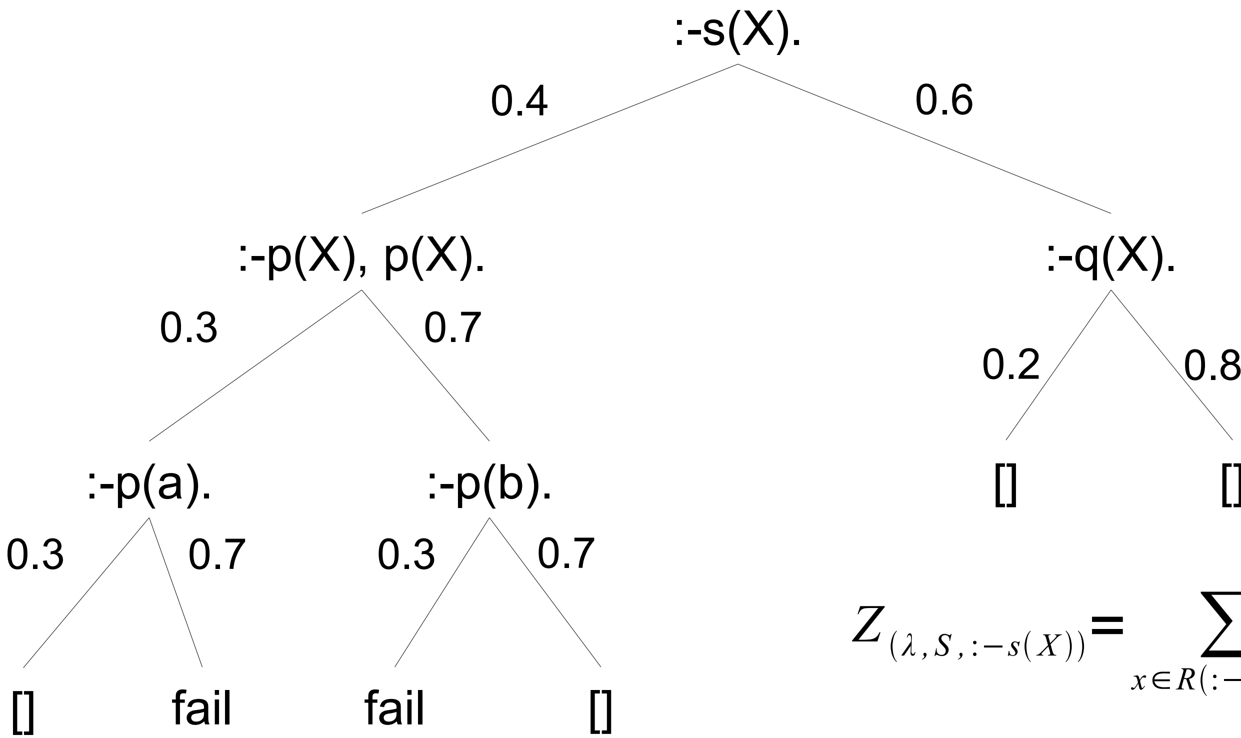
- $S = \{l_1:C_1,\ldots,l_n:C_n\}$
- $\lambda = (\ln l_1,\ldots,\ln l_n)$
- $R(G) = \{x$: x Beweis mit Anfang G$\}$
- $v_i(x)$ Anzahl Anwendungen C_i in x

$$\psi_{(\lambda,S,G)}(x) = \prod_{i=1}^{n} l_i^{v_i(x)} = e^{\lambda v(x)}$$

$$Z_{(\lambda,S,G)} = \sum_{x \in R(G)} \psi_{(\lambda,S,G)}(x)$$

$$\forall r \in R(G): \quad f_{(\lambda,S,G)}(r) = \frac{1}{Z_{(\lambda,S,G)}} \psi_{(\lambda,S,G)}(r)$$

Wahrscheinlichkeitsverteilung auf Beweisen



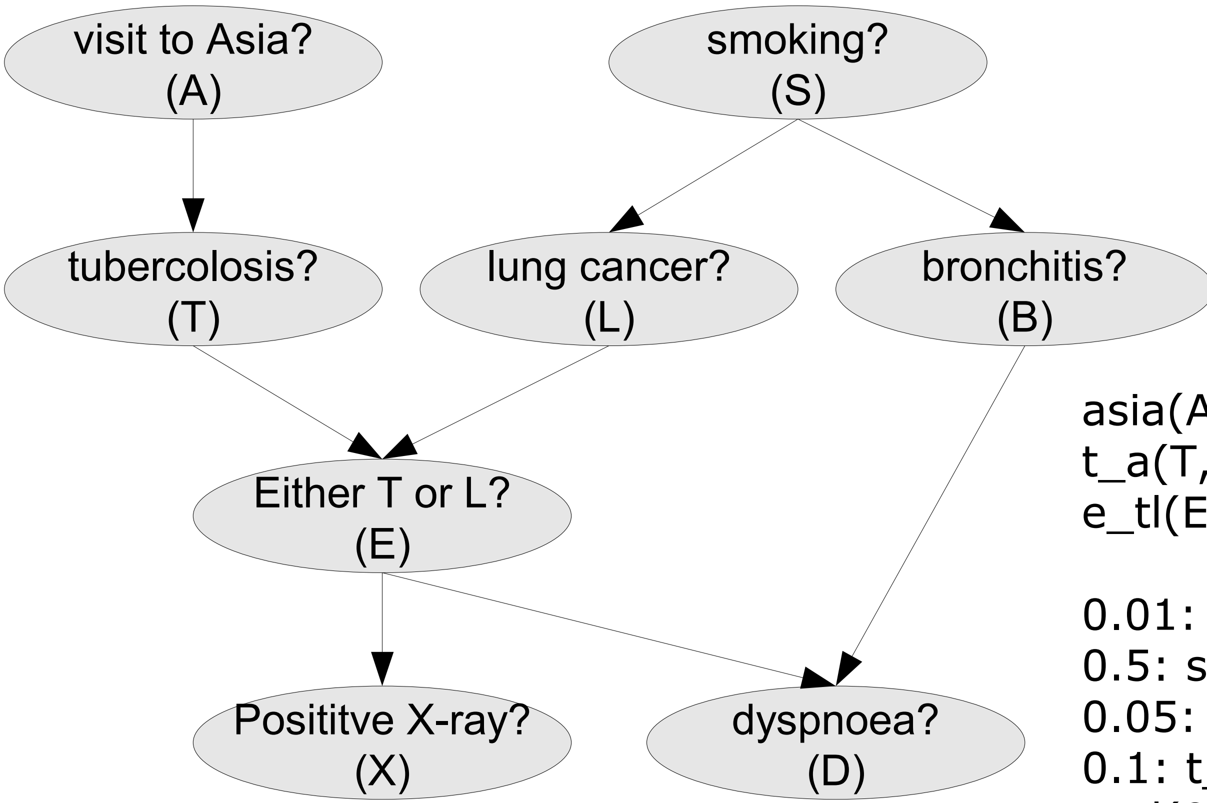
C_1 0.4: s(X) :- p(X),p(X).
C_2 0.6: s(X) :- q(X).
C_3 0.3: p(a).
C_4 0.7: p(b).
C_5 0.2: q(a).
C_6 0.8: q(b).

$$Z_{(\lambda,S,:-s(X))} = \sum_{x \in R(:-s(X))} \psi_{(\lambda,S,:-s(X))}(x) = 0.832$$

$$f_{(\lambda,S,:-s(X))}(C_1 C_3 C_3) = \frac{0.4 * 0.3 * 0.3}{0.832} = 0.043$$

$$f_{(\lambda,S,:-s(X))}(C_2 C_5) = \frac{0.6 * 0.2}{0.832} = 0.144$$

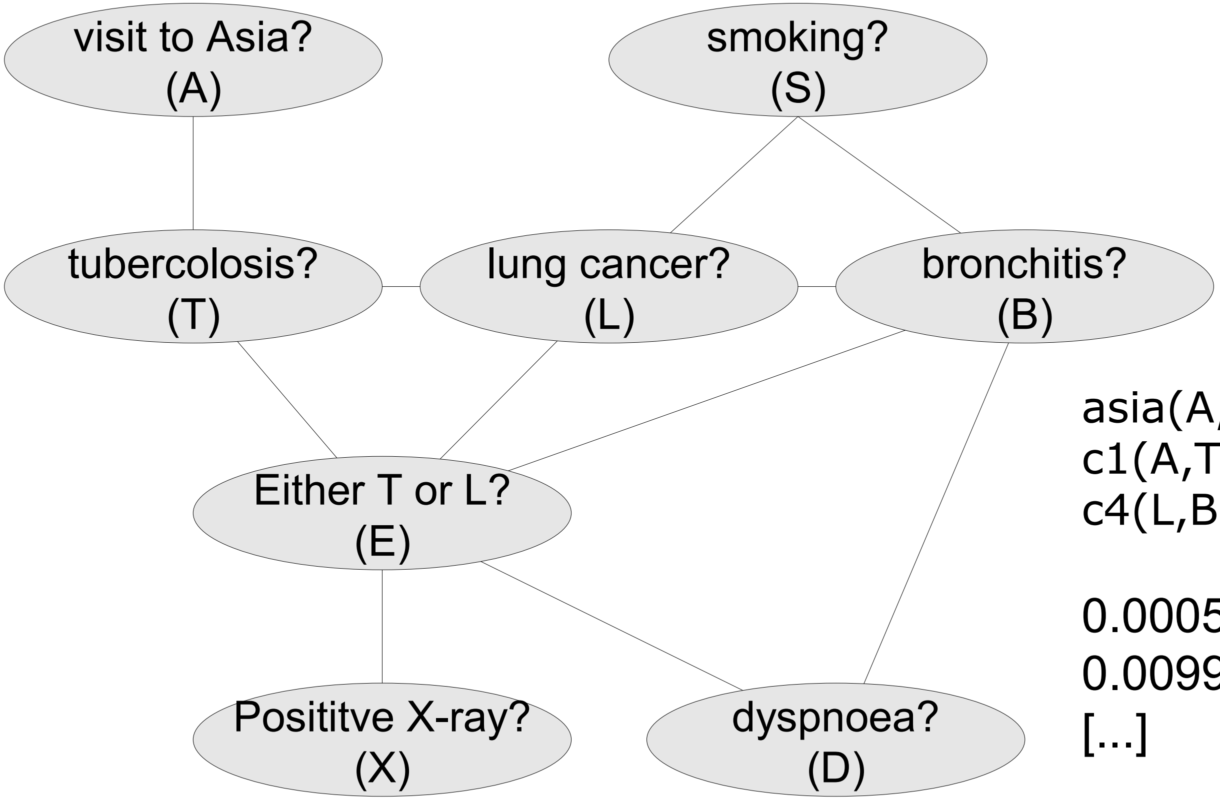
Anwendungsbeispiel: Bayes Netz




```
asia(A,T,E,S,L,B,X,D) :- a(A), s(S),
t_a(T,A), l_s(L,S), b_s(B,S),
e_tl(E,T,L), d_eb(D,E,B), x_e(X,E).

0.01: a(1).          0.99: a(0).
0.5: s(1).           0.5: s(0).
0.05: t_a(1,1).      0.95: t_a(0,1).
0.1: t_a(1,0).       0.9: t_a(0,0).
e_tl(0,0,0).         e_tl(1,0,1).
e_tl(1,1,0).         e_tl(1,1,1).
[...]
```

Anwendungsbeispiel: Markov Netz




```
asia(A,T,E,S,L,B,X,D) :-
c1(A,T), c2(L,S,B), c3(T,E,L),
c4(L,B,E), c5(E,D,B), c6(X,E).

0.0005: c1(1,1).   0.0095: c1(1,0).
0.0099: c1(0,1).   0.9801: c1(0,0).
[...]
```

Parameter Estimation

- Umkehrung des Problems:
  - Klauseln bekannt
  - Fakten bekannt durch Beobachtung von Daten
  - Parameter unbekannt
  - Schätze λ:
    - $S = \{l_1 : C_1, \ldots, l_n : C_n\}$
    - $\lambda = (\ln l_1, \ldots, \ln l_n)$

Parameter Estimation
Vereinfachte Problemstellung

- Zunächst Vereinfachung:
  - Annahme: Jeder Fakt hat nur eine Herleitung

- Bekannte Klauseln:

C_1	l_1: s(X,p) :- p(X),p(X).
C_2	l_2: s(X,q) :- q(X).
C_3	l_3: p(a).
C_4	l_4: p(b).
C_5	l_5: q(a).
C_6	l_6: q(b).

Parameter Estimation
Vereinfachte Problemstellung

C_1	l_1: s(X,p) :- p(X),p(X).
C_2	l_2: s(X,q) :- q(X).
C_3	l_3: p(a).
C_4	l_4: p(b).
C_5	l_5: q(a).
C_6	l_6: q(b).

Beobachtung in den Daten

Fakt	$\tilde{f}$	Ableitung	f_λ
s(a,p)	1/3	$C_1C_3C_3$	$Z_\lambda^{-1}\, l_1\, l_3^2$
s(b,p)	1/6	$C_1C_4C_4$	$Z_\lambda^{-1}\, l_1\, l_4^2$
s(a,q)	1/4	C_2C_5	$Z_\lambda^{-1}\, l_2\, l_5$
s(b,q)	1/4	C_2C_6	$Z_\lambda^{-1}\, l_2\, l_6$

$$Z_\lambda = l_1 l_3^2 + l_1 l_4^2 + l_2 l_5 + l_2 l_6$$
$$= l_1 (l_3^2 + l_4^2) + l_2 (l_5 + l_6)$$

Parameter Estimation
Vereinfachte Problemstellung

TECHNISCHE UNIVERSITÄT DARMSTADT

Fakt	$\tilde{f}$	Ableitung	f_λ
s(a,p)	1/3	$C_1C_3C_3$	$Z_\lambda^{-1}\, l_1\, l_3^2$
s(b,p)	1/6	$C_1C_4C_4$	$Z_\lambda^{-1}\, l_1\, l_4^2$
s(a,q)	1/4	C_2C_5	$Z_\lambda^{-1}\, l_2\, l_5$
s(b,q)	1/4	C_2C_6	$Z_\lambda^{-1}\, l_2\, l_6$

Klausel	$\tilde{f}[v_i]$	$f_\lambda[v_i]$
C_1	1/2	$Z_\lambda^{-1}\, l_1\, (l_3^2 + l_4^2)$
C_2	1/2	$Z_\lambda^{-1}\, l_2\, (l_5 + l_6)$
C_3	2/3	$Z_\lambda^{-1}\, 2\, (l_1\, l_3^2)$
C_4	1/3	$Z_\lambda^{-1}\, 2\, (l_1\, l_4^2)$
C_5	1/4	$Z_\lambda^{-1}\, l_2\, l_5$
C_6	1/4	$Z_\lambda^{-1}\, l_2\, l_6$

Löse Gleichungssystem nach $l_1,\ldots,l_n$ auf

Improved Iterative Scaling

- Initialisiere $\lambda^{(0)}$ beliebig
- 1.) Bestimme für jede Klausel C_i den Wert $\gamma_i^{(h)} \in [-\infty, \infty]$ für:

$$f_{\lambda^{(h)}}[v_i e^{\gamma_i^{(h)} v_\#}] = \tilde{f}[v_i]$$

- 2.) Setze $\lambda^{(h+1)} = \gamma^{(h)} + \lambda^{(h)}$
- 3.) Setze $h \leftarrow h+1$ und mache weiter mit 1. bis $f_{\lambda^{(h)}}$ konvergiert

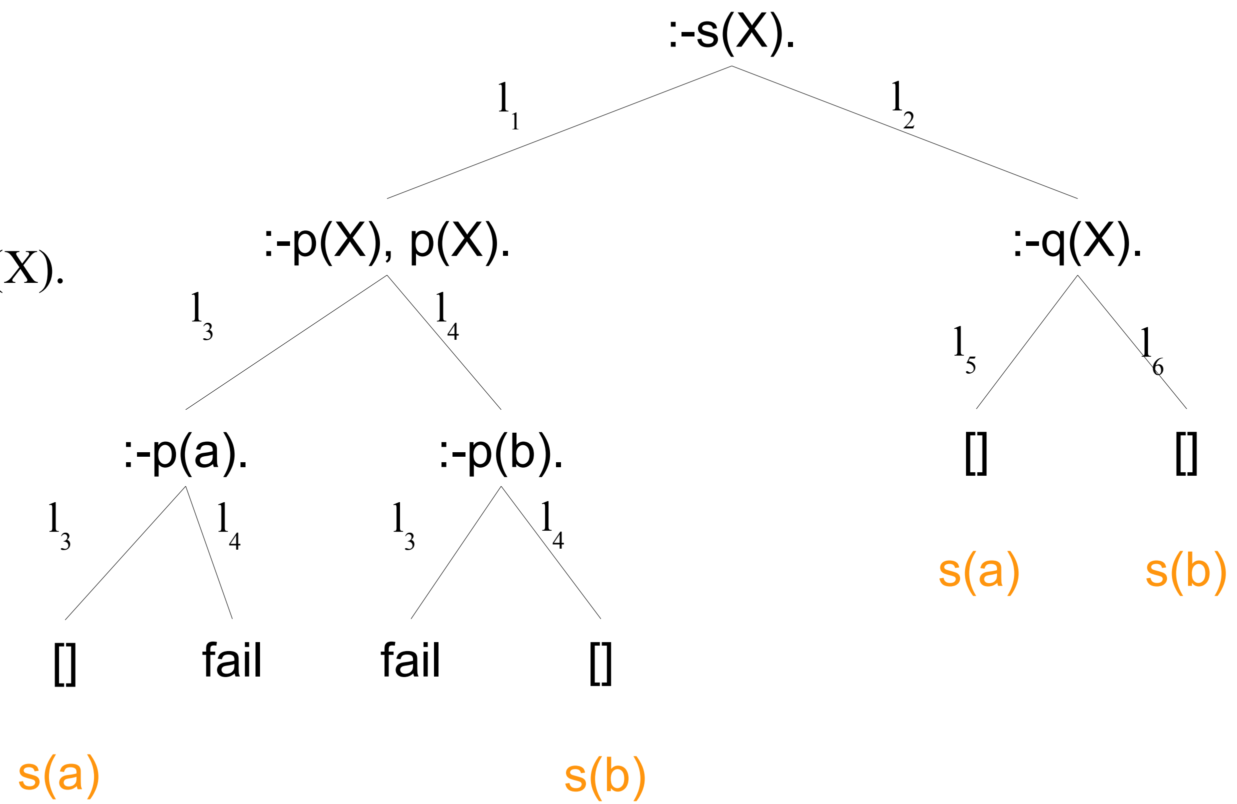
$$v_\#(r) = \sum_i v_i(r)$$

Parameter Estimation Verallgemeinerung

- Nun ohne Vereinfachung:
  - Fakt kann mehrere Ableitungen haben

- Bekannte Klauseln:

C_1	l_1: s(X) :- p(X),p(X).
C_2	l_2: s(X) :- q(X).
C_3	l_3: p(a).
C_4	l_4: p(b).
C_5	l_5: q(a).
C_6	l_6: q(b).



Parameter Estimation Verallgemeinerung

- Beobachtung in den Daten:

	s(a)	s(b)
Anzahl	7	5
$\tilde{p}$	7/12	5/12

y	r	$f_\lambda(r\|y)$	$f_\lambda(r)$
s(a)	$C_1C_3C_3$	$\frac{7}{12}\frac{l_1 l_3^2}{l_1 l_3^2+l_2 l_5}$	$Z_\lambda^{-1} l_1 l_3^2$
s(a)	C_2C_5	$\frac{7}{12}\frac{l_2 l_5}{l_1 l_3^2+l_2 l_5}$	$Z_\lambda^{-1} l_2 l_5$
s(b)	$C_1C_4C_4$	$\frac{5}{12}\frac{l_1 l_4^2}{l_1 l_4^2+l_2 l_6}$	$Z_\lambda^{-1} l_1 l_4^2$
s(b)	C_2C_6	$\frac{5}{12}\frac{l_2 l_6}{l_1 l_4^2+l_2 l_6}$	$Z_\lambda^{-1} l_2 l_6$

$$\begin{aligned} Z_\lambda &= l_1 l_3^2 + l_1 l_4^2 + l_2 l_5 + l_2 l_6 \\ &= l_1(l_3^2+l_4^2) + l_2(l_5+l_6) \end{aligned}$$

Parameter Estimation Verallgemeinerung

r	$f_\lambda(r\mid y)$	$f_\lambda(r)$
$C_1C_3C_3$	$\frac{7}{12}\frac{l_1 l_3^2}{l_1 l_3^2 + l_2 l_5}$	$Z_\lambda^{-1}\, l_1\, l_3^{\,2}$
C_2C_5	$\frac{7}{12}\frac{l_2 l_5}{l_1 l_3^2 + l_2 l_5}$	$Z_\lambda^{-1}\, l_2\, l_5$
$C_1C_4C_4$	$\frac{5}{12}\frac{l_1 l_4^2}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, l_1\, l_4^{\,2}$
C_2C_6	$\frac{5}{12}\frac{l_2 l_6}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, l_2\, l_6$

Klausel	$f_\lambda[v_i\mid y]$	$f_\lambda[v_i]$
C_1	$\frac{7}{12}\frac{l_1 l_3^2}{l_1 l_3^2 + l_2 l_5} + \frac{5}{12}\frac{l_1 l_4^2}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, l_1\, (l_3^{\,2} + l_4^{\,2})$
C_2	$\frac{7}{12}\frac{l_2 l_5}{l_1 l_3^2 + l_2 l_5} + \frac{5}{12}\frac{l_2 l_6}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, l_2\, (l_5 + l_6)$
C_3	$2\,\frac{7}{12}\frac{l_1 l_3^2}{l_1 l_3^2 + l_2 l_5}$	$Z_\lambda^{-1}\, 2\, (l_1\, l_3^{\,2})$
C_4	$2\,\frac{5}{12}\frac{l_1 l_4^2}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, 2\, (l_1\, l_4^{\,2})$
C_5	$\frac{7}{12}\frac{l_2 l_5}{l_1 l_3^2 + l_2 l_5}$	$Z_\lambda^{-1}\, l_2\, l_5$
C_6	$\frac{5}{12}\frac{l_2 l_6}{l_1 l_4^2 + l_2 l_6}$	$Z_\lambda^{-1}\, l_2\, l_6$

Anpassung: Improved Iterative Scaling

- Initialisiere $\lambda^{(0)}$ beliebig
- 1.) Bestimme für jede Klausel C_i den Wert $\gamma_i^{(h)} \in [-\infty, \infty]$ für:

  $$f_{\lambda^{(h)}}[v_i e^{\gamma_i^{(h)} v_\#}] = f_{\lambda^{(h)}}[v_i | y]$$

- 2.) Setze $\lambda^{(h+1)} = \gamma^{(h)} + \lambda^{(h)}$
- 3.) Setze $h \leftarrow h+1$ und mache weiter mit 1. bis $f_{\lambda^{(h)}}$ konvergiert

$$v_\#(r) = \sum_i v_i(r)$$

Zusammenfassung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Verallgemeinerung von stochastisch kontextfreien Grammatiken und Bayesschen Netzen
- Ausdrucksstarke, jedoch einfache Erweiterung logischer Programme
- Beschreibung & Lernen von Wahrscheinlichkeitsverteilungen

SLP Implementierungen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Progol
  - Regellerner
  - Erzeugt intern SLP
  - Hier definierte Syntax nicht erlaubt
  - http://www.doc.ic.ac.uk/~shm/progol.html

Literatur

- Chen, J., Kelley, L., Muggleton, S.: Protein Fold Discovery Using Stochastic Logic Programs
- Muggleton, S., Pahlavi, N.: Stochastic Logic Programs: A Tutorial
- Muggleton, S.: Stochastic Logic Programs
- Muggleton, S.: Learning Stochastic Logic Programs
- Cussens, J.: Parameter Estimation in Stochastic Logic Programs