

Übungsblatt 6: Belief Propagation

Abgabeschluss: Freitag, der 20. Juli 2007 um 18:00 Uhr.

Für dieses Aufgabenblatt sind sowohl Code als auch eine schriftliche Ausarbeitung abzugeben. Sendet Euren Code an `mikio@cs.tu-berlin.de` und `buenau@cs.tu-berlin.de` mit **Subject** "ML-Praktikum Abgabe *Name*". Gebt Eure Ausarbeitungen in unserem Sekretariat (FR 6-9) bei Frau Gerdes (Raum FR 6052) ab.

Bitte beachtet weiter die Coding-Richtlinien von Blatt 1

Aufgaben

Aufgabe 1: Implementation (30 Punkte)

In den folgenden beiden Aufgaben sollen zwei verschiedene Methoden zur Marginalisierung in Markov-Random-Fields (MRF) implementiert werden. Die Ein- und Ausgabeparameter haben jedoch dieselbe Bedeutung; fuer ein MRF mit n latenten Variablen und jeweils k Zuständen sind die Eingabeparameter wie folgt definiert:

- **A** ist eine $(n \times n)$ -Adjazenzmatrix welche die Graphenstruktur der n latenten Variablen definiert.
- **E** ist eine $(k \times n)$ -Matrix welche die Evidenz fuer jede latente Variable enthaelt, d.h. der (i, j) -te Eintrag von **E** ist der Wert $\phi_j(X_j = i)$.
- **L** ist eine $(k \times k)$ -Matrix welche die Kopplungen (compatibility function) der latenten Variablen enthaelt, d.h. der (i, j) -te Eintrag von **L** ist $\psi_{ij} = \psi(X_i = i, X_{i'} = j)$ fuer beliebige Variablen i, i' .

Das Ergebnis beider Funktionen ist:

- **m** ist eine $(k \times n)$ -Matrix deren i -te Spalte die normierte Randverteilung von x_i ist, d.h. der (i, j) -te Eintrag von **m** ist $p(X_j = i)$ und $\sum_{i=1}^k m_{ij} = 1$ fuer alle Variablen j .

Naïve Marginalisierung

Implementiere eine Funktion

```
m = simple_marg(A, E, L)
```

welche fuer das durch **A**, **E** und **L** definierte MRF die Randverteilungen aller latenten Variablen x_1, \dots, x_n durch einfache Summierung berechnet, d.h. unter Verwendung der Summenregel

$$p(x_i = k) \propto \sum_{x_1=1}^k \cdots \sum_{x_{i-1}=1}^k \sum_{x_{i+1}=1}^k \cdots \sum_{x_n=1}^k p(x_1, \dots, x_{i-1}, k, x_{i+1}, \dots, x_n).$$

Belief Propagation

Implementiere Marginalisierung durch Belief Propagation als Funktion

```
m = bp(A, E, L)
```

Teste Deine Implementation, indem Du die Ergebnisse mit `simple_marg` vergleichst.

Image Denoising

Entwickle und Implementiere eine Methode zum Entrauschen von binären Bildern (Pixel haben die Werte $\{0, 1\}$) mittels Belief Propagation als Funktion

$$\text{Id} = \text{bp_denoising}(\text{I}, \text{err_p}, \text{smooth_p})$$

mit den Parametern

- I ist eine $(n \times m)$ -Matrix welche das zu entauschende binäre Bild enthält.
- err_p ist ein Parameter im Intervall $[0, 1]$, welcher der Wahrscheinlichkeit im verwendeten Modell entspricht, dass ein Pixel verrauscht ist.
- smooth_p ist ein Parameter im Intervall $[0, 1]$, welcher proportional ist zur Wahrscheinlichkeit im verwendeten Modell, dass ein Pixel den gleichen Wert hat wie sein Nachbar.

Der Rückgabeparameter Id ist das entauschte Bild als $(n \times m)$ -Matrix wobei der (i, j) -te Eintrag von Id die Wahrscheinlichkeit (im Modell) ist, dass dieser Pixel den Wert 1 hat.

Beschreibe das von Dir verwendete Modell und den Algorithmus.

Aufgabe 2: Anwendung (20 Punkte)

1. Führe einen experimentellen Vergleich der Laufzeit der beiden Methoden zur Marginalisierung in MRFs durch. Erstelle einen Plot, in dem die benötigte Laufzeit der Methoden vs. die Größe des gewählten MRFs aufgetragen ist.
2. Verwende Dein Image Denoising Verfahren, um die auf der Website verfügbaren binären Bilder zu entauschen. Verrausche die Bilder, indem Du mit Wahrscheinlichkeit p den Wert eines Pixels umdrehst. Untersuche das Verhalten Deiner Methode in Abhängigkeit vom noise-level p und den Parametern err_p und smooth_p . Illustriere die Wirkungsweise der Methode durch Plots von verrauschten und entauschten Bildern.