

B003725 Intelligenza Artificiale (2019/20)

Studente: Diego Biagini — <2020-02-25 Tue>

Elaborato assegnato per l'esame finale

Istruzioni generali

Il lavoro svolto sarà oggetto di discussione durante l'esame orale e dovrà essere sottomesso per email due giorni prima dell'esame, includendo:

1. Sorgenti e materiale sviluppato in autonomia (non includere eventuali datasets reperibili online, per i quali basta fornire un link);
2. Un file README che spieghi:
 - come usare il codice per riprodurre i risultati sottomessi
 - se vi sono parti del lavoro riprese da altre fonti (che dovranno essere **opportunamente citate**);
3. Una breve relazione (massimo 4 pagine in formato pdf) che descriva il lavoro ed i risultati sperimentali. Non è necessario ripetere in dettaglio i contenuti del libro di testo o di eventuali articoli, è invece necessario che vengano fornite informazioni sufficienti a *riprodurre* i risultati riportati.

La sottomissione va effettuata preferibilmente come link ad un repository **pubblico** su [github](#), [gitlab](#), o [bitbucket](#). In alternativa è accettabile allegare all'email un singolo file zip; in questo caso è **importante evitare di sottomettere files eseguibili** (inclusi files `.jar` o `.class` generati da Java), al fine di evitare il filtraggio automatico da parte del software antispam di ateneo!

Inferenza con junction trees nei belief networks

Questo esercizio riguarda l'inferenza su reti Bayesiane. Assumendo di conoscere il junction tree di una rete Bayesiana su un universo U (p.es. visualizzabile utilizzando il programma [Hugin Lite](#) o costruendolo manualmente), si scriva un programma (in un linguaggio di programmazione a scelta) per la propagazione in modo da calcolare le probabilità condizionali $P(Q|e)$ per arbitrari $Q \in U$ e $e \subset U$. Si confrontino i risultati con quelli ottenuti da Hugin Lite quando si usa la stessa rete con gli stessi parametri, utilizzando un numero ragionevole di queries. Alcuni semplici esempi di reti da utilizzare possono essere reperiti nella cartella `samples` del software Hugin Lite. Dettagli sull'algoritmo del junction tree esposto in classe sono riportati in [Jensen 1997](#) (pw ai19) (Ch. 4, esclusa la parte di costruzione del junction tree che non va implementata).