Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea in Informatica

Benchmark di sistemi per la programmazione Bayesiana: tempo, accuratezza, concisione

Relatore:
Chiar.mo Prof.
UGO DAL LAGO

Presentata da:
VALENTINA FERRAIOLI

Sessione II
Anno Accademico 2018/2019

Indice

Introduzione

Capitolo 1

Probabilità Bayesiana

1.1 Probabilità

La probabilità è una misura che permette di quantificare la possibilità che un evento si verifichi oppure no. Ha lo scopo di formulare e studiare modelli matematici che permettono di descrivere il comportamento di un evento, nonostante non si sia in grado di predirlo con totale certezza. Secondo la definizione classica la probabilità di un evento casuale e' data dal rapporto tra il numero di casi favorevoli al presentarsi di tale evento ed il numero totale di casi possibili, supponendo che tutti i possibili casi siano equiprobabili. In termini matematici:

$$P(E) = \frac{n_E}{n} \tag{1.1}$$

dove n_E corrisponde al numero di volte in cui potrebbe verificarsi E ed n è il numero di esperimenti fatti. Tuttavia è possibile riconoscere due diverse interpretazioni della probabilità:

- interpretazione frequentista: la probabilità viene definita in modo oggettivo come il rapporto tra il numero di volte in cui si verifica l'evento di interesse ed il numero totale di prove effettuate, per un numero di prove che tende all'infinito. In termini matematici:

$$P(E) = \lim_{x \to \infty} \frac{n_e}{n} \tag{1.2}$$

- interpretazione bayesiana: la probabilità viene definita in modo soggettivo, infatti alle ipotesi viene associata una probabilità che indica il grado di fiducia che un individuo ha nel verificarsi di un certo evento, il quale dipende dalle informazioni che ha riguardo l'evento e la.

Non si può definire se un approccio aia migliore dell'altro, ma si può notare che entrambe le interpretazioni soddisfano gli assiomi della probabilità:

Assumendo che Ω sia l'insieme di casi possibili, che $|\Omega| = n$ sia la sua cardinalità e che E_i, con i=1,2..,n siano gli eventi di Ω, allora:

$$0 \leq P(E_i) \leq 1 \tag{1.3}$$

$$P(\Omega) = 1 \tag{1.4}$$

$$P(\cup_{n\to\infty}^{\infty} E_i) = \Sigma_{n\to\infty}^{\infty} P(E_i) \tag{1.5}$$

La differenza sostanziale dei due approcci si deve cercare nel concetto che viene assegnato alla probabilità, che inciderà sull'utilizzo e l'importanza del teorema di Bayes, anch'esso comune nelle due interpretazioni.

In questa sezione verrà analizzata in maniera più approfondita la probabilità e l'inferenza Bayesiana. Per cominciare verranno richiamate alcune definizioni che sono alla base di questa, quali il teorema di Bayes, probabilità condizionata, distribuzione a priori e a posteriori.

1.2 Richiamo definizioni

1.2.1 Distribuzione

La funzione di distribuzione descrive la probabilità che un evento assuma un particolare valore.

1.2.2 Probabilità condizionata

In teoria della probabilità la probabilità condizionata di un evento A rispetto a un evento B è la probabilità che si verifichi A, sapendo che B si è verificato. Viene indicata come:

$$P(A|B) = \frac{P(A,B)}{P(A)} \tag{1.6}$$

dove $P(B) \neq 0$ e P(A,B) indica la probabilità congiunta.

La probabilità condizionata esprime una "correzione" delle aspettative per A, dettata dall'osservazione di B.

1.2.3 Probabilità congiunta

La probabilità congiunta indica la probabilità che due eventi A e B si verifichino nello stesso momento. Si indica come:

$$P(A,B) = P(A|B)P(A) \tag{1.7}$$

Nel caso in cui due eventi siano indipendenti, ovvero conoscere lo stato di A non ci fornisce alcuna informazione in più sullo stato di B, la probabilità congiunta equivale a:

$$P(A,B) = P(A)P(B) \tag{1.8}$$

1.2.4 Marginalizzazione

Conoscendo la distribuzione congiunta P(A,B) è possibile ricavare la distribuzione di P(X) usando la marginalizzazione

$$P(A) = \Sigma_B P(A, B)) \tag{1.9}$$

1.2.5 Teorema di Bayes

Il teorema di Bayes mette in relazione la probabilità condizionata con quella congiunta. Enunciato: La probabilità a posteriori è uguale alla probabilità a priori per likelihood ratio.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}) \tag{1.10}$$

dove:

- probabilità a priori: corrisponde a P(A), ovvero alla probabilità che l'evento A si verifichi prima di prendere in considerazione le osservazioni.
- probabilità a posteriori: corrisponde a $P(A|B)$, ovvero alla probabilità che l'evento si verifichi avendo considerato le osservazioni. Può essere considerata un aggiustamento della probabilità a priori.
- funzione di verosomiglianza (likelihood): corrisponde a $P(B|A)P(A)$,ovvero alla probabilità che si verifichi un evento A, avendo osservato B.

1.3 Probabilistic Reasoning

Probabilistic reasoning combina la nostra conoscenza rispetto ad una situazione con le regole della probabilità. Determina fattori critici che non sono stati analizzati,ai fini di prendere una buona decisionein situazioni di incertezza. Per fare ciò ha bisogno di un modello, dell'applicazione delle evidenze che si possiedono sul modello e di tratte delle conclusioni. Utilizzando il probabilistic reasoning è possibile eseguire tre diversi tipi di ragionamento:

- predire eventi futuri: attraverso le osservazioni possedute in questo momento si cerca di predire la probabilità che un evento accada.
- inferire le cause agli effetti: abbiamo un osservazione sul presente e vogliamo definire la probabilità che alcuni eventi si siano verificati.
- imparare da eventi futuri per predire in modo migliore i futuri: vengono prese tutte le informazioni riguardo a situazioni precedenti più quella corrente, verrà fatta inferenza che migliorerà i risultati dell'evento predetto.

Inferenza

The process of using the model to answer queries based on the evidence is called probabilistic inference, or simply inference. Fortunately, computer algorithms have been developed that do the math for you

Il processo di inferenza permette partendo da un modello di rispondere a domande basate sull'osservazione di alcuni eventi. E' un processo che viene fatto quotidianamente dalla mente umana in situazioni di incertezza, per esempio:
"Attraversando la strada tutti abbiamo fatto ciò: Sei di fretta, quindi invece di arrivare alle prossime strisce pedonali guardi a destra e a sinistra e vedi una macchina ad una distanza sufficiente perchè tu riesca ad attraversare la strada senza essere investito. Quindi tu cammini (e immagino) attraversi la strada illeso.[Las12]

In questo esempio non si sa con certezza che la macchina vista non stesse andando troppo veloce da investirci eppure conosciamo un modello, composto anche da esperienze passate, e nel vedere la macchina ad una certa distanza e con una certa velocità , riusciamo a decidere se attraversa la strada senza rischiare di essere investiti. In queste situazioni si sta compiendo un processo di inferenza.

1.3.1 Bayesian reasoning

Come per la definizione di probabilità anche il probabilistic reasoning si divide in Bayesian e Frequentist reasoning.

Nella prospettiva bayesiana, abbiamo detto che la probabilità descrive l'incertezza con il quale è possibile prevedere l'esito del fenomeno in questione e che questa incertezza può derivare dall'intrinseca variabilità del processo, ma anche dalla nostra imperfetta conoscenza di esso.
Ora l'idea base è quella di fare inferenza (bayesiana), ossia sviluppare delle previsioni (probabilità a posteriori), su quantità non note in base all'evidenza disponibile e a un'ipotesi iniziale(probabilità a priori). "E' possibile fare tutto ciò utilizzando il teorema di Bayes che viene interpretato in questo modo:

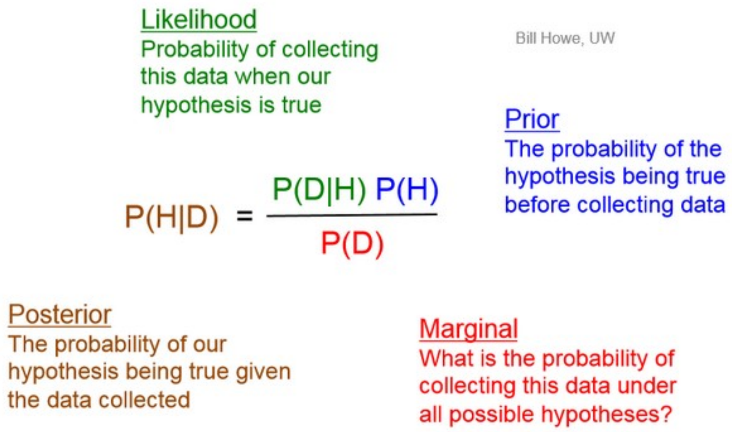

Figura 1.1: Bayes theorem.Fonte:

1.3.2 Analisi Bayesiana

L'analisi Bayesiana consiste quindi nel:

- creare un modello di probabilità che rappresenti i dati
- identificare le evidenze
- eseguire l'inferenza
-

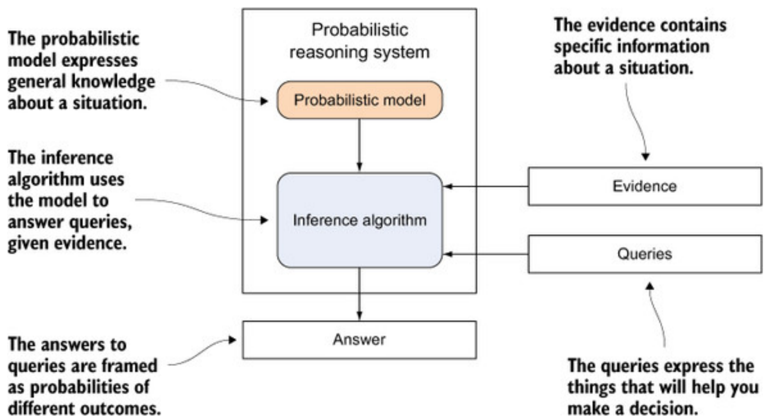

Figura 1.2: Bayes theorem.
Fonte:"https://dpzbhybb2pdcj.cloudfront.net/pfeffer/Figures/01fig03alt.jpg"

Esempio 1:

Un giorno Tracey si sveglia è realizza che l'erba del suo giardino è bagnata. Avrà piovuto durante la notte oppure si è dimenticata di spegnere l'irrigatore? Dopodichè si rende conto che anche l'erba del suo vicino è bagnata. Qual è la probabilità che Tracey abbia lasciato l'irrigatore acceso durante la notte?

Modello:

$R \in 0, 1 \rightarrow R = 1$ ha piovuto, 0 altrimenti
$S \in 0, 1 \rightarrow S = 1$ Tracey ha lasciato l'irrigatore acceso, 0 altrimenti
$J \in 0, 1 \rightarrow J = 1$ l'erba di Jack $\grave{e}$ bagnata, 0 altrimenti
$T \in 0, 1 \rightarrow T = 1$ l'erba di Tracey $\grave{e}$ bagnata, 0 altrimenti

Il modello per il problema di Tracey corrisponde alla probabilita congiunta di P(T,J,R,S). Per procedere con la risoluzione e' necessario, inoltre, definire il tipo di dipendenza che c'è tra i diversi eventi: $P(T|S, R)$ e $P(J|R)$.

$$P(T, J, R, S) = P(J|S, R)P(J|R)P(R)P(S)$$

Queste relazioni possono essere rappresentate anche in modo grafico attraverso una Bayes Network, che verrà spiegata approfonditamente successivamente.

Dal problema evince che T=1 e R=1. Supponiamo che le probabilità a priori siano: P(R)=0.2 P(S)=0.1 Mentre le altre probabilità corrispondono a:

$$p(J = 1|R = 1) = 1, p(J = 1|R = 0) = 0.2$$

$$p(T = 1|R = 1, S = 0) = 1, p(T = 1|R = 1, S = 1) = 1$$

$$p(T = 1|R = 0, S = 1) = 0.9, p(T = 1|R = 0, S = 0) = 0$$

.

A questo punto si può iniziare ad eseguire l'inferenza andando ad applicare il teorema di Bayes. L'obiettivo è calcolare P(S—T,J). Iniziamo con il calcolare P(S—T) e vediamo come cambiano le due probabilità con l'aggiunta di un evidenza.

$$P(S|T) = \frac{P(S,T)}{P(T)} = \frac{\Sigma_{J,R}\, P(T, J, R, S)}{\Sigma_{J,R\,,S}\, P(T, J, R, S)} \tag{1.11}$$

$$= \frac{\Sigma_{J,R}\, P(J|R)P(T|R, S)P(R)P(S)}{\Sigma_{J,R\,,S}\, P(J|R)P(T|R, S)P(S)} \tag{1.12}$$

$$= \frac{\Sigma_{R} P(T|R, S)P(R)P(S)}{\Sigma_{R,S}\, P(T|R, S)P(S)P(R)} = 0,3340 \tag{1.13}$$

Si può notare che la probabilità a posteriori è aumentata rispetto a quella a priori, tenendo conto dell'evidenza che T = 1.

Ora andiamo a calcolare $P(S|T,J)$ e vediamo come si comporta la probabilità a posteriori in questo caso.

$$P(S|T,J) = \frac{(S,T,J)}{P(T,J)} = \frac{\Sigma_R P(T,J,R,S)}{\Sigma_{R,S}\, p(T,J,R,S)} \tag{1.14}$$

$$= \frac{\Sigma_R P(J|R)P(T|R,S)P(R)P(S)}{\Sigma_R P(J|R)P(T|R,S)P(R)P(S)} = 0.160 \tag{1.15}$$

La probabilità che Tracey abbia lasciato l'irrigatore acceso, data l'evidenza che anche l'erba di Jack è bagnata è inferiore di quella che tiene conto solo dell'erba di Tracey. Questo perchè l'osservazione che l'erba di Jack è bagnata aumenta la probabilità che sia vero che abbia piovuto.

1.4 Belief Network

Definizione:
Una belief Network è una distribuzione della forma:

$$P(x_1, ..., x_n) = \Pi_{i=1}^{D}(x_i|Pa(x_1)) \tag{1.16}$$

dove $Pa(x_i)$ equivale alle variabili parenti di x_i.

Come è stato accennato precedentemente è possibile rappresentare un modello probabilistico attraverso una Belief Network(Bayesian Belief Network) che non è altro che un DAG (Direct Acyclic Graph) che è in grado di rappresentare le dipendenze condizionali di un modello.

All'interno della Belief network ogni nodo rappresenta una variabile ed è associato ad una funzione di probabilità,che prende in input un insieme di possibili valore che potrebbe assumere il nodo padre e restituisce la distribuzione di probabilità del nodo. Gli archi, invece, rappresentano le dipendenze condizionali delle variabili. L'assenza di un cammino tra due nodi implica che le due variabili sono condizionalmente indipendenti. Rappresentare la dipendenza tuttavia non è così immediato, infatti bisognerebbe tenere conto di alcuni pattern, quali:

- $A \to B \to C$: viene chiamato "chain ". A primo impatto si potrebbe pensare che il valore di A influenzi C attraverso B, ma non è così. Infatti, è osservabile che condizionando la varibile B =1 il valore di C non varierà indipendentemente dal valore che assume B. Quindi le variabili A e C sono indipendenti.

- $A \leftarrow B \to C$: viene chiamato "fork". Anche in questo caso abbiamo che A e C sono indipendenti, nonostante sembrerebbero essere correlati attraverso B.

- $A \rightarrow B \leftarrow C$: viene chiamato "collider". A e C sono indipendenti in linea generale, ma condizionando B diventano dipendenti. Più precisamente si ha che A e B sono inversamente proporzionali, quindi un valore alto di A spiega il perchè di un valore basso di C.

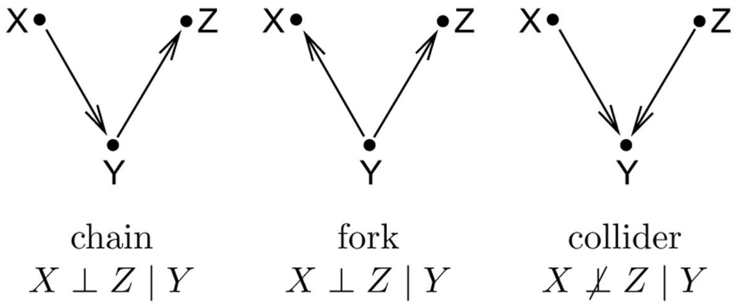

Figura 1.3: Fonte:
https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-8-S6-S5/figures/3

le Bayesian Network si prestano bene a fare inferenza, sfruttano la probabilità condizionale, del quale dispongono gia,data la definizione di belief network. A partire da questa viene fatta la valutazione della variabile desiderata. Per fare inferenza, si possono utilizzare anche altre tecniche, come ad esempio MCMC, che si presta bene ad inferire su reti di grandi dimensioni.

Esempio 2:

Si andrà a rappresentare una belief network per l'esempio 1.

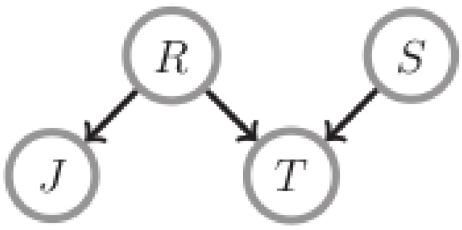

Figura 1.4:

Ogni nodo del grafo rappresenta una delle variabili della probabilità congiunta, e le variabili che alimentano altre, come ad esempio R e S , rappresentano le variabili che si trovano alla destra in una probabilità condizionata. E' possibile vedere che vengono anche rispettati i pattern spigati sopra; infatti R e S sono indipendenti e non presentano cammini tra essi, mentre J che è dipendente da R viene raggiunto solo da R e T che dipende da R e S viene raggiunto da entrambi.

Capitolo 2

Probabilistic Programming

Finora si è arrivati alla conclusione che grazie alla probabilità e all'inferenza è possibile ragionare in condizioni di incertezza. Negli ultimi anni c'è stato un grande sviluppo in campi come Artificial Intelligence e Data Science, nei quali si ha bisogno di fare questo tipo di ragionamento , ma soprattutto si ha la necessità di lavorare su modelli di dimensione sempre maggiore. Per semplificare la modellizzazione e rendere l'inferenza automatica in questo tipo di problemi è stata introdotta la probabilistic programming. Per eseguire la programmazione probabilistica è necessario utilizzare dei PPL (Probabilistic programming language), che permettono di esprimere i modelli come programmi e non come costrutti matematici e possiedono una suite di algoritmi di inferenza che potranno essere applicati ai programmi.
Tutti questi programmi nascono come estensioni di linguaggi già esistenti, ad esempio, il linguaggio logico problog, basato su prolog e anglican, linguaggio funzionale che estende Clojoure.

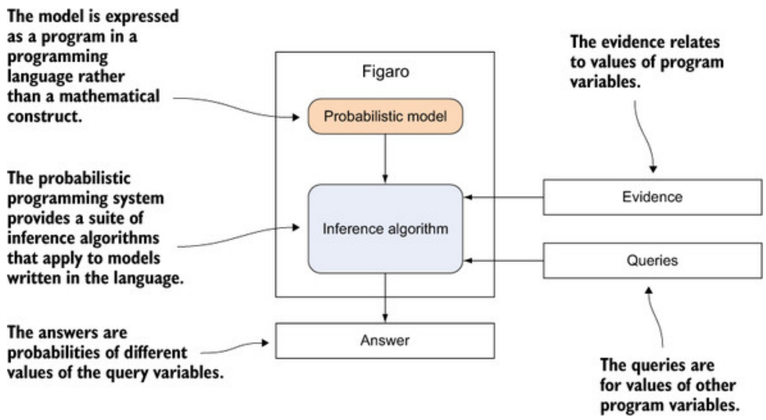

Figura 2.1: Bayes theorem.
Fonte:https://dpzbhybb2pdcj.cloudfront.net/pfeffer/Figures/01fig07alt.jpg

Lo scopo di questa sezione sarà quella di approfondire alcuni linguaggi probabilistici; per questo verranno analizzati i paradigmi dal quale nascono e il modo che impiegano per affrontare la programmazione probabilistica.

2.1 Il paradigma funzionale

La programmazione funzionale è un paradigma di programmazione dichiarativo, ovvero, viene chiesto cosa bisogna calcolare, descrivendo la logica di una computazione senza definire come deve essere calcolato, ovvero viene omesso l'intero flusso di controllo. In questo tipo di programmazione gli statement vengono sostituiti con delle espressioni. La programmazione funzionale è composta da alcuni elementi che la caratterizzano:

- funzioni di ordine superiore: sono funzioni che possono ricevere altre funzioni come parametro o che possono restituire altre funzioni.
- funzioni pure: non hanno side-effects, questo evita le dipendenza di una funzione da altre, e permettere di modificare una funzione senza influire sull'altra.
- ricorsione: è il costrutto più utilizzato per eseguire dei cicli all'interno della programmazione funzionale.
- trasparenza referenziale: nei linguaggi funzionali non sono presenti assegnamenti, questo permette di essere certi che quando una funzione viene valutata il valore di una variabile non sia diverso da quello con cui è stata definita.

Esempio

Linguaggio imperativo vs funzionale:
Imperativo:

```
int factorial(int n){
    int result = 1;
    for( ; i < n; (n --){
        result*=n;
    }
    return result;
}
}
```

Funzionale:

```
fact::Integer -> Integer
fact 0 = 1
fact n | n > 0 = n * fact (n - 1)
```

2.2 Il paradigma logico

Quello logico è un paradigma di programmazione basato sulla logica del primo ordine. E' composto da:

- regole: corrispondono a delle implicazioni e sono della forma:
  Testa :- Corpo.
  che si legge: "Testa è vera se Corpo è vero". (Si noti che la regola termina con un punto.)

- fatti: Un singolo termine (anche composto), senza il segno ":-", viene chiamato fatto. I fatti equivalgono a regole senza corpo, che sono considerate automaticamente vere. Un esempio di fatto è:
  gatto(tommaso).

Fanno parte della famiglia dei linguaggi dichiarativi e sono utilizzati per verificare la correttezza di un programma e per ottimizzare linguaggi di tipo imperativo, infatti, e possibile passare da un programma logico ad uno imperativo che sia piu efficiente applicando delle tecniche di trasformazione. Sono caratterizzati da una netta separazione tra logica e controllo:
Programma logico = logic + control
dove la logica rappresenta il modello che si vuole rappresentare e il control si occupa di fornire una prova ed e completamente indipendente dal programmatore.

Esempio

Linguaggio imperativo vs funzionale:
Imperativo:

```
int factorial(int n){
    int result = 1;
    for( ; i < n; (n --){
        result*=n;
    }
    return result;
}
```

Logico:

```
factorial(0,1).

factorial(N,F) :-
   N>0,
   N1 is N-1,
```

```
  factorial(N1,F1),
  F is N * F1.
```

2.3 PPL

Capitolo 3

Risultati e conclusioni

Bibliografia

[Las12] Daniel Lassiter. «Probabilistic reasoning and statistical inference:An introduction (for linguists and philosophers)». In: *NASSLLI 2012 Bootcamp* (2012). DOI: `https://web.stanford.edu/~danlass/NASSLLI-coursenotes-combined.pdf`.