

Inferența prin enumerare în rețele bayesiene, aplicată pe un model al bolilor respiratorii

Profesor coordonator:

Prof. dr. ing. Florin LEON

Studenti:

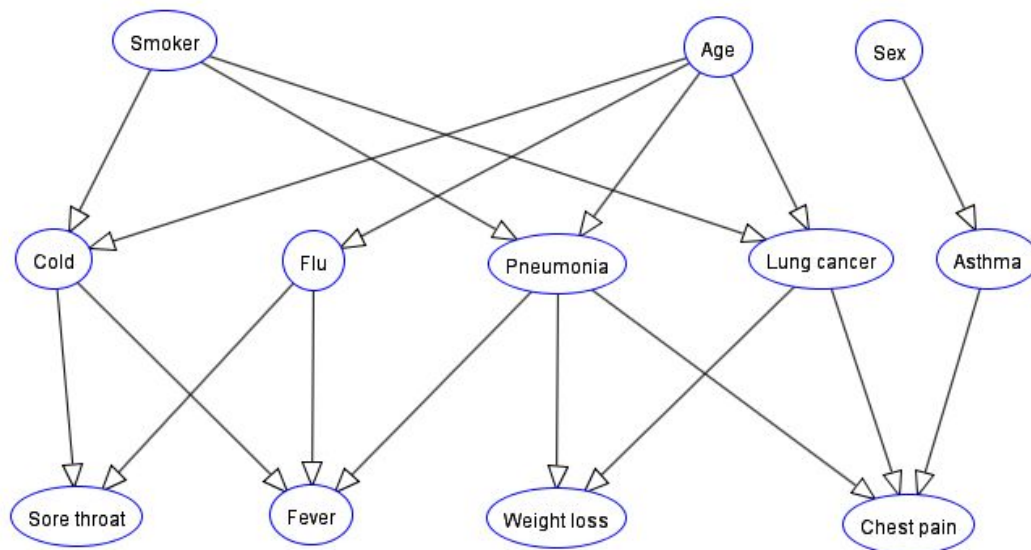
Paul-Cristian BRÎNZĂ

Andrei-Iulius COVAȘ

Mircea-Constantin DOBREANU

Problema considerată

Aplicația își propune determinarea probabilității ca un pacient să aibă o anumită boală respiratorie, date fiind anumite simptome și/sau factori de risc (vârstă, sex, fumător). În realizarea scopului propus anterior, ne vom folosi de inferența prin enumerare în rețele bayesiene. Datele medicale vor fi preluate în principal de pe site-ul CDC-ului american sau alte site-uri de specialitate. Modelul propus de noi este ilustrat mai jos:



Aspecte teoretice privind algoritmul

Probabilitatea este fracțiunea care cuantifică așteptarea ca un eveniment să aibă loc. Acestea poate lua valori reale cuprinse între 0 și 1.

Probabilitatea lui A, condiționat de B, reprezintă fracțiunea de lumi în care B este adevărată și atunci și A este adevărată. Probabilitatea condiționată este măsura numerică a șansei întâmplării unui eveniment după ce s-au luat în considerare anumite evidențe, dovezi.

Inferența este o operație logică de trecere de la premise la consecințe logice.

O rețea bayesiană este un graf orientat aciclic în care toate nodurile pot fi descrise prin informații probabilistice. Mai riguros, o rețea bayesiană are următoarele proprietăți:

- Oricare nod corespunde unei variabile aleatorii (fie ea discretă sau continuă).
- Dacă rețeaua conține o muchie de la nodul X la nodul Y spunem că X este părintele (sau cauza) lui Y. Vom spune că Y este copilul (sau efectul) lui X. Rețeaua nu conține cicluri.
- Toate variabilele din rețea (nodurile) au asociate o distribuție de probabilitate condiționată de stările părinților de forma $P(X_i | \text{parents}(X_i))$.

Sarcina de bază pentru orice sistem inferențial probabilistic este să determine distribuția de probabilități condiționate a unui eveniment dat fiind anumite dovezi. Prin dovezi, ne referim la faptul că valorile altor evenimente au fost deja setate.

Modalitatea de rezolvare

În rezolvarea problemei, ne vom folosi de inferența prin enumerare (o metodă de inferență exactă). Inferența prin enumerare se bazează pe următoarea ecuație:

$$P(X|e) = \alpha P(X, e) = \alpha \sum_y P(X, e, y)$$

Cu alte cuvinte, orice probabilitate condiționată poate fi aflată prin sumarea unor produse de probabilități condiționate deja cunoscute. Pseudocodul pentru a face sumările:

```
function ENUMERATION-ASK( $X, \mathbf{e}, bn$ ) returns a distribution over  $X$ 
  inputs:  $X$ , the query variable
            $\mathbf{e}$ , observed values for variables  $\mathbf{E}$ 
            $bn$ , a Bayes net with variables  $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$  /*  $\mathbf{Y} = \text{hidden variables}$  */

   $Q(X) \leftarrow$  a distribution over  $X$ , initially empty
  for each value  $x_i$  of  $X$  do
     $Q(x_i) \leftarrow$  ENUMERATE-ALL( $bn.VARS, \mathbf{e}_{x_i}$ )
    where  $\mathbf{e}_{x_i}$  is  $\mathbf{e}$  extended with  $X = x_i$ 
  return NORMALIZE( $Q(X)$ )

function ENUMERATE-ALL( $vars, \mathbf{e}$ ) returns a real number
  if EMPTY?( $vars$ ) then return 1.0
   $Y \leftarrow$  FIRST( $vars$ )
  if  $Y$  has value  $y$  in  $\mathbf{e}$ 
    then return  $P(y | \text{parents}(Y)) \times$  ENUMERATE-ALL(REST( $vars$ ),  $\mathbf{e}$ )
  else return  $\sum_y P(y | \text{parents}(Y)) \times$  ENUMERATE-ALL(REST( $vars$ ),  $\mathbf{e}_y$ )
    where  $\mathbf{e}_y$  is  $\mathbf{e}$  extended with  $Y = y$ 
```

Implementarea

Am implementat aplicația în limbajul C#, iar modelul rețelei bayesiene este reprezentat într-un fișier XML. Soluția cuprinde mai multe proiecte:

- **BayesianNetwork** - modul librărie care se ocupă cu inferența propriu-zisă
- **BayesianNetwork.Tests** - modul de unit tests pentru modulul de inferență
- **NetworkParser** - modul librărie care se ocupă de parsarea fișierelor XML
- **NetworkParser.Tests** - modul de unit tests pentru modulul de parsare
- **UserInterface** - aplicația WinForms care expune utilizatorului modelul propus

Diagrama de mai jos, reprezintă diagrama de clase generată de mediul de programare Visual Studio 2015 pentru modulul de inferență.

Network
Class

Properties

- Nodes { get; } : IDictionary<string, Node>

Methods

- addLink(Node cause, Node effect) : Network
- addLink(string cause, string effect) : Network
- addNode(string _name) : Network
- addNode(string _name, IEnumerable<string> _domainValues) : Network
- answer(Query question) : double
- enumerateAll(ICollection<Node> nodes, ICollection<Fact> facts) : double
- enumerationAsk(Node target, ICollection<Fact> facts) : Dictionary<Query, double>
- getNodeByName(string nodeName) : Node
- normalizeDistribution(Dictionary<Query, double> distribution) : Dictionary<Query, double>
- topologicalSort(ICollection<Node> nodes) : ICollection<Node>

Query
Class

Properties

- Facts { get; } : ICollection<Fact>
- Target { get; } : Node
- Value { get; } : string

Methods

- addFact(Fact item) : Query
- Equals(object obj) : bool
- GetHashCode() : int
- Query(Node _target, string _value, IEnumerable<Fact> _facts)

Node
Class

Properties

- Causes { get; } : ICollection<Node>
- DomainValues { get; } : ICollection<string>
- Effects { get; } : ICollection<Node>
- Name { get; set; } : string
- Network { get; } : Network
- ProbabilityDistribution { get; } : IDictionary<Query, double>

Methods

- addCause(Node cause) : void
- addEffect(Node effect) : void
- addProbability(Query query, double value) : void
- Node(string _name, IEnumerable<string> _domainValues, Network _network)
- Node(string _name, Network _network)
- ToString() : string

Fact
Class

Properties

- Node { get; } : Node
- Value { get; } : string

Methods

- Equals(object obj) : bool
- Fact(Node _node, string _value)
- Fact(string _nodeName, string _value, Network _network)
- GetHashCode() : int
- ToString() : string

În continuare, este prezentat codul sursă pentru părțile relevante ale modulului, regăsite în implementarea clasei Network:

```
/**
 * Metoda wrapper pentru enumerationAsk
 */
public double answer(Query question)
{
    var distribution = enumerationAsk(question.Target, question.Facts);
    return distribution[question];
}

/**
 * Calculează distribuția unei variabile date fiind niște dovezi
 */
private Dictionary<Query, double> enumerationAsk(Node target, ICollection<Fact> facts)
{
    var distribution = new Dictionary<Query, double>();
    var nodes = topologicalSort(target.Network.Nodes.Values);

    foreach (var possibleValue in target.DomainValues)
    {
        var newFacts = new List<Fact>(facts);
        newFacts.Add(new Fact(target, possibleValue));

        distribution.Add(new Query(target, possibleValue, facts), enumerateAll(nodes, newFacts));
    }
}
```

```

    }

    return normalizeDistribution(distribution);
}
/**
 * Calculeaza probabilitatea date fiind anumite fapte. Acest fapt se realizează prin recursie, la
 * fiecare pas adăugându-se fapte noi în care a fost setată o altă variabilă aleatorie
 */
private double enumerateAll(ICollection<Node> nodes, ICollection<Fact> facts)
{
    if (nodes.Count == 0)
    {
        return 1.0;
    }
    else
    {
        var head = nodes.First();

        var nodesTail = new List<Node>(nodes);
        nodesTail.Remove(head);

        bool isHeadSet = facts.Any(fact => fact.Node == head);
        if (isHeadSet)
        {
            string headValue = facts.First(fact => fact.Node == head).Value;
            var causalFacts = facts.Where(fact => head.Causes.Any(node => node == fact.Node));
            double probabilityOfTarget = head.ProbabilityDistribution[new Query(head,
headValue, causalFacts)];

            return probabilityOfTarget * enumerateAll(nodesTail, facts);
        }
        else
        {
            double probabilityOfTarget = 0.0;
            foreach (string headValue in head.DomainValues)
            {
                var causalFacts = facts.Where(fact => head.Causes.Any(node => node ==
fact.Node));

                double probabilityOfTargetCase = head.ProbabilityDistribution[new
Query(head, headValue, causalFacts)];

                ICollection<Fact> newFacts = new List<Fact>(facts);
                newFacts.Add(new Fact(head, headValue));

                probabilityOfTarget += probabilityOfTargetCase * enumerateAll(nodesTail,
newFacts);
            }

            return probabilityOfTarget;
        }
    }
}
/**
 * Această funcție scalează probabilitățile obținute fiindcă ele nu mai au suma 1 (ci mai mică)
 */
private Dictionary<Query, double> normalizeDistribution(Dictionary<Query, double> distribution)
{
    var normalized = new Dictionary<Query, double>();
    double sumOfOldProbabilities = distribution.Values.Sum();
    double alpha = 1 / sumOfOldProbabilities;

```

```

foreach (var kvp in distribution)
{
    normalized[kvp.Key] = alpha * kvp.Value;
}

return normalized;
}

```

Rezultate obținute

- Pentru cazul în care pacientul prezintă simptome de durere în piept(*chest pain*), este de gen feminin, face parte din categoria oamenilor în vârstă și este fumător, rezultatele sunt următoarele:

The screenshot shows the BayesianInterface application window. The 'smoker' section has 'true' selected. The 'sex' section has 'f' selected. The 'age' section has 'elder' selected. The 'Symptoms' section has 'chest pain' selected. The results text box displays the following probabilities:

- Probability of having pneumonia = 45.3836 %
- Probability of having asthma = 43.3024 %
- Probability of having cold = 13 %
- Probability of having flu = 9 %
- Probability of having lung cancer = 1.8849 %

A 'Diagnose' button is located at the bottom right of the window.

- Pentru cazul în care pacientul prezintă simptomele: febră(*fever*) și durere în gât(*sore throat*), este adult, de gen masculin și este fumător, rezultatele sunt următoarele:

The screenshot shows the BayesianInterface application window. The 'smoker' section has 'true' selected. The 'sex' section has 'm' selected. The 'age' section has 'adult' selected. The 'Symptoms' section has 'fever' and 'sore throat' selected. The results text box displays the following probabilities:

- Probability of having flu = 75.2585 %
- Probability of having cold = 26.8088 %
- Probability of having asthma = 10 %
- Probability of having pneumonia = 9.4651 %
- Probability of having lung cancer = 0.033 %

A 'Diagnose' button is located at the bottom right of the window.

- Rezultate se obțin și în momentul în care nu se cunosc date ca vârsta, sex-ul sau dacă pacientul este fumător sau nefumător, ci doar simptomele acestuia:

The screenshot shows the 'BayesianInterface' window. It has four main sections: 'smoker', 'sex', 'age', and 'Symptoms'. The 'smoker' section has 'false' and 'true' checkboxes. The 'sex' section has 'f' and 'm' checkboxes. The 'age' section has 'child', 'adult', and 'elder' checkboxes. The 'Symptoms' section has 'fever', 'chest pain', 'weight loss', and 'sore throat' checkboxes, with 'sore throat' selected. A text box in the center displays the following probabilities:

- Probability of having asthma = 59.1994 %
- Probability of having flu = 38.0696 %
- Probability of having cold = 30.6271 %
- Probability of having pneumonia = 27.3271 %
- Probability of having lung cancer = 0.1951 %

A 'Diagnose' button is located at the bottom right.

- Alte rezultate pot fi observate în continuare:

The screenshot shows the 'BayesianInterface' window with the same layout as the first screenshot. In this instance, 'false' is selected for 'smoker', 'f' for 'sex', and 'child' for 'age'. The 'Symptoms' section has 'fever', 'chest pain', 'weight loss', and 'sore throat' checkboxes, with 'sore throat' selected. The text box in the center displays the following probabilities:

- Probability of having cold = 17 %
- Probability of having flu = 13 %
- Probability of having asthma = 7 %
- Probability of having pneumonia = 6 %
- Probability of having lung cancer = 0 %

A 'Diagnose' button is located at the bottom right.

The screenshot shows the 'BayesianInterface' window with the same layout. In this instance, 'true' is selected for 'smoker', 'm' for 'sex', and 'elder' for 'age'. The 'Symptoms' section has 'fever', 'chest pain', 'weight loss', and 'sore throat' checkboxes, with 'sore throat' selected. The text box in the center displays the following probabilities:

- Probability of having pneumonia = 91.8275 %
- Probability of having flu = 39.9269 %
- Probability of having cold = 32.6549 %
- Probability of having asthma = 15.8454 %
- Probability of having lung cancer = 4.0795 %

A 'Diagnose' button is located at the bottom right.

Concluzii

Având în vedere rezultatele bune obținute în urma rulării câtorva scenarii posibile, prezentate mai sus, putem spune că metoda inferenței prin enumerare, a fost folosită cu succes pentru rezolvarea problemei propuse. Cu toate acestea, metoda utilizată, este aplicabilă numai pe o rețea de dimensiuni reduse, gestionarea unei rețele mai mari (50 - 70 sau mai multe noduri, din care unele cu distribuții non-binare) devenind foarte dificilă.

Contribuții

Brînză Paul-Cristian

- Documentație
- Modulul de parsat XML

Covaș Andrei-Iulius

- Documentație
- Interfață
- Creare model (fișierul XML)

Dobreanu Mircea-Constantin

- Modulul de logică a rețelelor bayesiene
- Documentație

Bibliografie

Russell, Stuart J., and Peter Norvig. *Artificial Intelligence: a Modern Approach*. Pearson, 2016.

Leon, Florin, “Curs 10 IA - Probabilități”, *Cursuri Inteligență Artificială*, 2018,

http://florinleon.byethost24.com/Curs_IA/IA10_Probabilitati.pdf

“Asthma.” *Centers for Disease Control and Prevention*, Centers for Disease Control and Prevention, 15 May 2018, www.cdc.gov/asthma/most_recent_data.htm.

“Influenza (Flu).” *Centers for Disease Control and Prevention*, Centers for Disease Control and Prevention, 25 Oct. 2018, www.cdc.gov/flu/about/burden/2014-2015.html.

Monto, Arnold S. “Clinical Signs and Symptoms Predicting Influenza Infection.” *JAMA*, American Medical Association, 27 Nov. 2000, jamanetwork.com/journals/jamainternalmedicine/fullarticle/485554.

Simasek, Madeline, and David Blandino. "Treatment of the Common Cold." *AAFP Home*, 15

Feb. 2007, www.aafp.org/afp/2007/0215/p515.html.

"Symptoms - Chest Pain." *LungCancer.net*, lungcancer.net/symptoms/chest-pain/.

"USCS Data Visualizations." *Centers for Disease Control and Prevention*, Centers for

Disease Control and Prevention, gis.cdc.gov/Cancer/USCS/DataViz.html.

"Viral Pneumonia." *NeuroImage*, Academic Press, 22 Mar. 2011,

www.sciencedirect.com/science/article/pii/S0140673610614596?via=ihub.

"The World Factbook: World." *Central Intelligence Agency*, Central Intelligence Agency, 1

Feb. 2018, www.cia.gov/library/publications/the-world-factbook/geos/xx.html.

Arroll, Bruce "Common Cold." *National Center for Biotechnology Information*, 16 Mar.

2011, www.ncbi.nlm.nih.gov/pmc/articles/PMC3275147/