

Prova 2 Bimestre Análise de Algoritmos de Raciocínio Probabilístico

Luís Aurélio Campos

luis.campos@utp.edu.br

0.1 Introdução

Nesse artigo será realizada a análise de um trabalho realizado pelo GEORGE MIGUEL GIORDANI, pela UNIVERSIDADE FEDERAL DE RORAIMA com o título de "Programação Probabilística Aplicada a Classificação de Documentos com Máquina de Ponto de Bayes e Alocação de Dirichlet Latente", que se refere a pesquisa sobre o entendimento da área de raciocínio probabilístico na área da inteligência artificial, onde será verificado uma breve explicação sobre o tema, alguns problemas relacionados a essa área e os algoritmos para resolver os mesmos.

0.2 Fundamentação Teórica

0.2.1 Raciocínio Probabilístico

O objetivo principal do raciocínio probabilístico, ou lógica probabilística, é tentar encontrar uma combinação de probabilidades para lidar com a incerteza de problemas que não possuem uma solução padrão, ou seja, problemas que lidam com a sorte. Nesse artigo, será verificado os seguintes algoritmos abordados pelo George:

- **Bayes Point Machine (BPM)**

- **infer.NET**

0.2.2 Bayes Point Machine (BPM)

O método de Bayes Point Machine (BPM) é um classificador linear, onde o mesmo realiza a classificação a partir de uma função linear, onde o resultado é atribuído uma classificação positiva, caso seja maior que zero o resultado da função, ou uma classificação negativa, sendo o resultado da função menor que 0. Funções lineares utilizam fronteiras, onde separadas por uma linha, é buscado o melhor resultado próximos a essa fronteira, onde também é necessário possuir conhecimento desses resultados.

0.2.3 infer.NET

O infer.NET é um framework gratuito e open source criada pela equipe de desenvolvedores da Microsoft Research, onde o mesmo propõe uma abordagem para resolver diversos tipos de problemas, incluindo problemas do tipo padrão como classificação, recomendação ou clustering, e também dos que requerem aprendizagem de máquina. Para utilizar esse framework, é necessário fazer uso de uma API de modelação, onde é obrigatório utilizar

o namespace `MicrosoftResearch.Infer.Models` no projeto, e assim podendo fazer o uso da mesma através de métodos estáticos de classe `Variable` (C Sharp).

```
Variable<double> mean = Variable.GaussianFromMeanAndVariance(0, 100);
Variable<double> precision = Variable.GammaFromShapeAndScale(1, 1);
VariableArray<double> data = Variable.Constant(new double[] { 11, 5, 8, 9 });
Range i = data.Range;
data[i] = Variable.GaussianFromMeanAndPrecision(mean, precision).ForEach(i);
```

FIGURA 1 – Fonte: GEORGE MIGUEL GIOR-DANI

0.3 Análise dos Problemas

0.3.1 Bayes Point Machine (BPM)

O problema consiste em, com informações de idade, salário e uma indicação se a pessoa irá ou não comprar o produto, deseja - se descobrir se as mesmas irão ou não comprar determinado produto informado pelo usuário. Esse problema é de característica de aprendizado, onde é necessário "ensinar" a máquina antes de resolver o problema.

0.3.2 infer.NET

O problema que será resolvido utilizando esse framework consiste em descobrir qual a probabilidade em, ao lançar duas moedas juntas, obter duas caras nessas moedas. Esse problema tem como característica que nem sempre o resultado será o mesmo, apesar de se ter conhecimento da chance de determinada situação ocorrer, assim dificultando uma previsão exata de determinado resultado.

0.4 Resultados experimentais

0.4.1 Bayes Point Machine (BPM)

O primeiro passo é definir as variáveis para realizar o treinamento do algoritmo:

```
//training set("Data Set")
double[] salario = { 63, 16, 28, 55, 22, 20 };
double[] idades = { 38, 23, 40, 27, 18, 40 };
bool[] compra = { true, false, true, true, false, false };
//cria o vetor y
VariableArray<bool> y = Variable.Observed(compra).Named("y");
```

FIGURA 2 – Fonte: GEORGE MIGUEL GIOR-DANI

Após isso, é criado dois vetores X e Y, onde X é composto pelos dados de treinamento e Y pelas observações de compra, e armazenado esses valores, o George cria um método chamado `BayesPointMachine`, onde ao utilizar o mesmo:

```
//classifica os dados abaixo.
double[] salarioTest = { 58, 18, 22 };
double[] idadeTest = { 36, 24, 37 };

VariableArray<bool> yTest = Variable.Array<bool>(new Range(idadeTest.Length));
bpm.BayesPointMachine(salarioTest, idadeTest, Variable.Random(wPosterior), yTest);
WriteLine("classificação: \n" + engine.Infer(yTest));
```

FIGURA 3 – Fonte: GEORGE MIGUEL GIOR-DANI

É possível obter os seguintes resultados:

```
Compiling model...done.
classificação:
[0] Bernoulli(0,9555)
[1] Bernoulli(0,1565)
[2] Bernoulli(0,287)
```

FIGURA 4 – Fonte: GEORGE MIGUEL GIOR-DANI

0.4.2 infer.NET

de uma breve introdução sobre o que é o raciocínio probabilístico.

O primeiro passo é a definição de três variáveis aleatórias, onde a primeira e a segunda são as moedas que serão utilizadas, e a terceira é a variável que armazena o valor de "ambas caras". A quarta linha do código do George faz a ligação entre as 3 variáveis anteriormente criadas, para a probabilidade de cair duas vezes cara ao lançar as 2 moedas. Na 5 linha, é definido o motor de inferência, onde o mesmo irá nos apresentar qual a probabilidade de acontecer aquilo que é passado por parâmetro.

```
Variable<bool> fisrtCoin = Variable.Bernoulli(0.5).Named("Primeira Moeda");
Variable<bool> secondCoin = Variable.Bernoulli(0.5).Named("Segunda Moeda");
//Variavel aleatoria derivada
Variable<bool> bothHeads = Variable.New<bool>().Named("Ambas caras");
bothHeads = fisrtCoin & secondCoin;
//motor de inferencia
InferenceEngine ie = new InferenceEngine();
//print de resultados
Writeline("Probailidade de duas moedas serem cara: " + ie.Infer(bothHeads));
bothHeads.ObservedValue = false;
Writeline("probabildiade da primeira moeda ser cara: " + ie.Infer(fisrtCoin));
```

FIGURA 5 – Fonte: GEORGE MIGUEL GIOR-DANI

E com isso, é possível obter os seguintes resultados:

```
Compiling model...done.
Probailidade de duas moedas serem cara: Bernoulli(0,25)
Compiling model...done.
probabildiade da primeira moeda ser cara: Bernoulli(0,3333)
// print de resultados
```

FIGURA 6 – Fonte: GEORGE MIGUEL GIOR-DANI

0.5 Considerações Finais

Nesse artigo foram analisados dois algoritmos utilizados pelo GEORGE MIGUEL GIOR-DANI no seu trabalho apresentado pela UNIVERSIDADE FEDERAL DE RORAIMA, onde foi discutido o funcionamento dos mesmos, quais problemas é possível ser resolvido e as características desses problemas, além

REFERÊNCIAS

GEORGE MIGUEL GIORDANI. *Programação Probabilística aplicada a classificação de documentos com Máquina de Ponto de Bayes e Alocação de Dirichlet Latente*. 2017. Disponível em: <http://ufrr.br/computacao/index.php?option=com_phocadownload&view=category&download=219:\protect\@normalcr\relaxprogramacao-probabilistica-aplicada-a-classificacao-de-documentos-com-maquina-de-ponto-\protect\@normalcr\relaxde-bayes-e-alocacao-de-dirichlet-latente&id=23:tccs-defendidos-2017-1&Itemid=261>.

Luiz Eduardo S. Oliveira. *Reconhecimento de Padrões Classificadores Lineares*. —. Disponível em: <<http://www.inf.ufpr.br/lesoliveira/padroes/classificadoreslineares.pdf>>.

Wikipédia. *Lógica probabilística*. 2019. Disponível em: <https://pt.wikipedia.org/wiki/L%C3%B3gica_probabil%C3%ADstica>.

Wikipédia. *Infer.NET*. 2020. Disponível em: <<https://en.wikipedia.org/wiki/Infer.NET>>.

(GEORGE MIGUEL GIORDANI, 2017)

(Luiz Eduardo S. Oliveira, -)

(Wikipédia, 2019)

(Wikipédia, 2020)