# 主观贝叶斯推理网络实验报告

|  |  |
| --- | --- |
| 班级: | 软工2203 |
| 学号: | 221310332 |
| 姓名: | 周立成 |
| 老师: | 肖黎丽 |

## 问题重述:

已知下列规则:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

且先验几率,通过用户得到,.求出

**要求**:**通过题目的样例,且保持语句主体不变的情况下,改变,以及,和也能得到正确的结果。**

## 分析和设计:

1. 在储存规则上,使用结构体**Rules**,来储存4条规则,其中包括,以及证据和结论的字符串。
2. 再储存额外的信息时,使用,以及来分别保存先验几率和用户观察
3. 同时应该支持以下贝叶斯运算(尽管我们只用了一部分)
   1. 根据求出:
   2. 从得到:
   3. 从得到:
   4. 从贝叶斯基本公式得到:
   5. 从更新的贝叶斯公式得到:
   6. 得到:
   7. 得到:
   8. EH公式:



* 1. CP公式:



* 1. 所有观察下的后验概率:



1. 根据IF-ELSE条件判断一一实现以上的函数即可
2. 运算部分根据书上例题部分计算,即:



1. 各函数头以及部分如下:

1. pair<long double, long double> Get\_P\_E\_H\_and\_P\_E\_not\_H(long double LS, long double LN)

2. long double get\_Not(long double P)

3. long double From\_P\_get\_O(long double P)

4. long double get\_P\_E\_from\_Bayes\_basic\_formula(long double P\_EH, long double P\_H, long double P\_HE)

5. long double get\_P\_HE(long double P\_EH, long double P\_H, long double P\_E)

6. long double get\_P\_NotH\_E(long double P\_E\_NotH, long double P\_NotH, long double P\_E)

7. pair<long double, long double> Updated\_Bayes\_O\_HE\_O\_H\_Not\_E(pair<long double, long double> LS\_LN, long double O\_H)

8. long double get\_P\_ES(long double C\_ES, long double P\_E)

9. long double get\_C\_ES(long double P\_ES, long double P\_E)

10. long double get\_P\_HS\_from\_Duda\_Original(long double P\_HE, long double P\_ES, long double P\_H\_NotE, long double P\_NotE\_S)

11. long double get\_P\_HS\_from\_EH\_Linear\_Interpolation(long double P\_H, long double P\_E, long double P\_ES, long double P\_HE,long double P\_H\_not\_E)

12. long double get\_P\_HS\_From\_CP\_formula(long double P\_H, long double P\_HE, long double P\_H\_NotE, long double C\_ES)

13. long double Get\_Posterior\_Probability\_From\_All\_Observation (vector<long double> O\_HS, long double O\_H)

14. long double process\_O\_H\_S(int index\_H, int index\_RULE, int index\_C)//采用1,2,3,4的下标输入

1. 整体的结构图如下:



## 实验测试:

### 实验环境:

* CPU: 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz
* 内存:16GB
* 操作系统:WINDOWS 11 22H2
* 编译工具:MVSC 17.0
* 编程语言:C++
* 编程环境:CLION 2023.2

### 测试:

输出结果:

D: \Bayes.exe

Input LS1 and LN1 for E1 and H1

**2 0.000001**

IF E1 THEN (2.000000,0.000001) H1

Input LS2 and LN2 for E2 and H1

**100 0.000001**

IF E2 THEN (100.000000,0.000001) H1

Input LS3 and LN3 for H1 and H2

**65 0.01**

IF H1 THEN (65.000000,0.010000) H2

Input LS4 and LN4 for E3 and H2

**300 0.0001**

IF E3 THEN (300.000000,0.000100) H2

Now input O(H1)

**0.1**

Now input O(H2)

**0.01**

Now input C(E1|S1)

**3**

Now input C(E2|S2)

**1**

Now input C(E3|S3)

**-2**

P\_H1=0.0909091

O\_H1\_S1=0.157895

P\_H1=0.0909091

O\_H1\_S2=0.341463

O\_H1\_S1S2=0.539153

P\_H2=0.00990099

P\_H1\_S1S2=0.350292

O\_H2\_H1 =0.65

P\_H2\_H1=0.393939

P\_H2\_S1S2=0.119475

O\_H2\_S1S2=0.135686

P\_H1=0.00990099

O\_H2\_S3=0.0059765

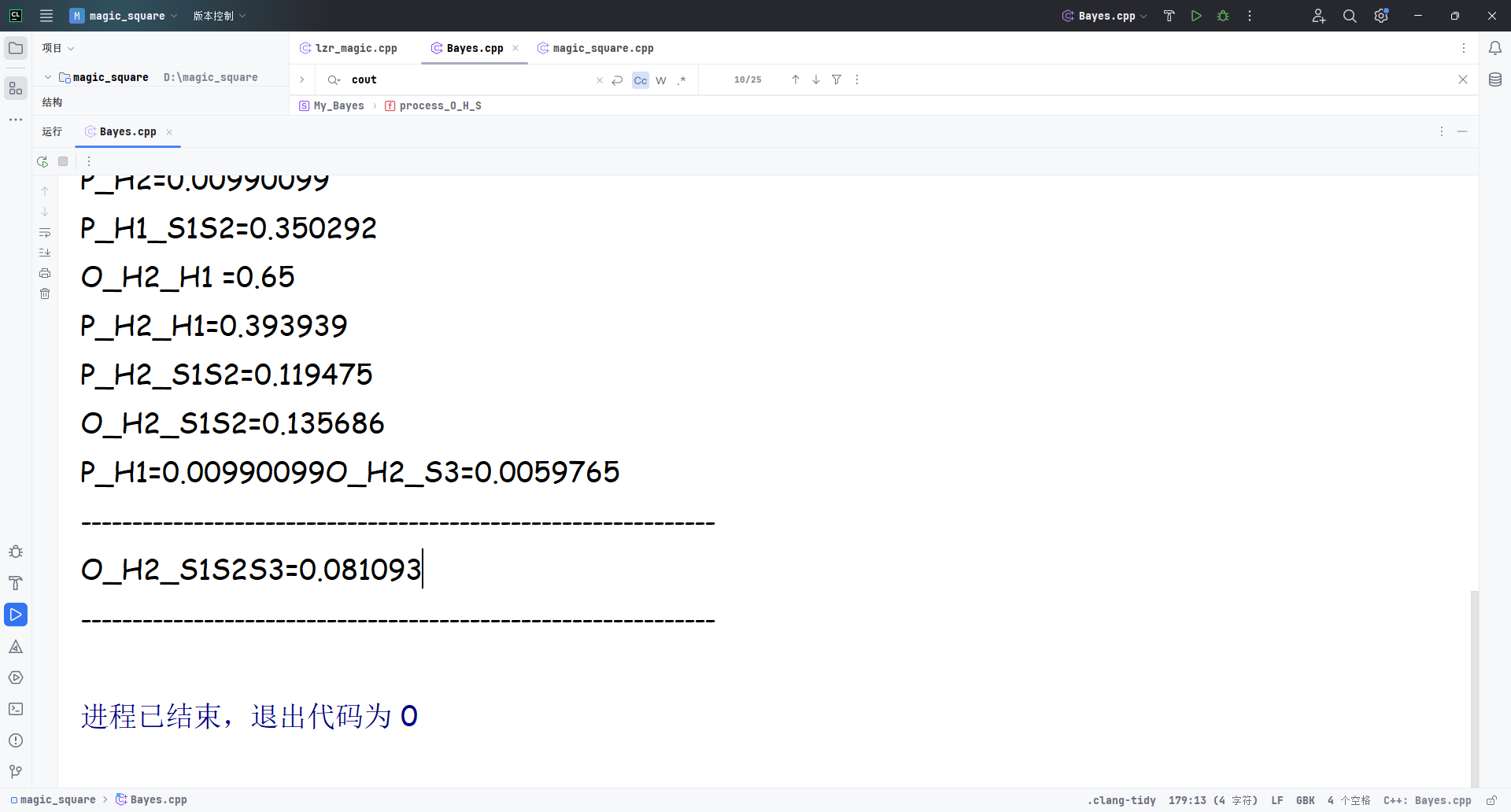
--------------------------------------------------------------

O\_H2\_S1S2S3=0.081093

--------------------------------------------------------------

进程已结束，退出代码为 0

**与书上结果一致,运行正确**



## 改进和不足:

* 希望能自己通过Rules生成推理网络自己推理