实验三 贝叶斯网络

一、实验目的

1. 了解贝叶斯网络的基本概念
2. 掌握训练贝叶斯网络模型的方法
3. 学会灵活运用模型

二、实验要求

1．独立完成实验

2．书写实验报告书

三、实验原理

朴素贝叶斯分类法假定属性值条件独立，然而在实践中，变量之间可能存在依赖关系。贝叶斯信念网络说明联合条件概率分布，它提供一种因果关系的图形模型。信念网络由两个成分定义——有向无环图和条件概率表的集合。有向无环图的每个节点代表一个随机变量，每条弧表示一个概率依赖。

四、实验内容

1. Python 实现贝叶斯网络

（1）Python 库安装

使用基于 pgmpy 来构造贝叶斯网络和进行建模训练。pgmpy 是一款基于 Python 的概率图模型包，主要包括贝叶斯网络和马尔可夫蒙特卡洛等常见概率图模型的实现以及推断方法。本节使用 pgmpy 包来实现简单的贝叶斯网络。

安装命令：

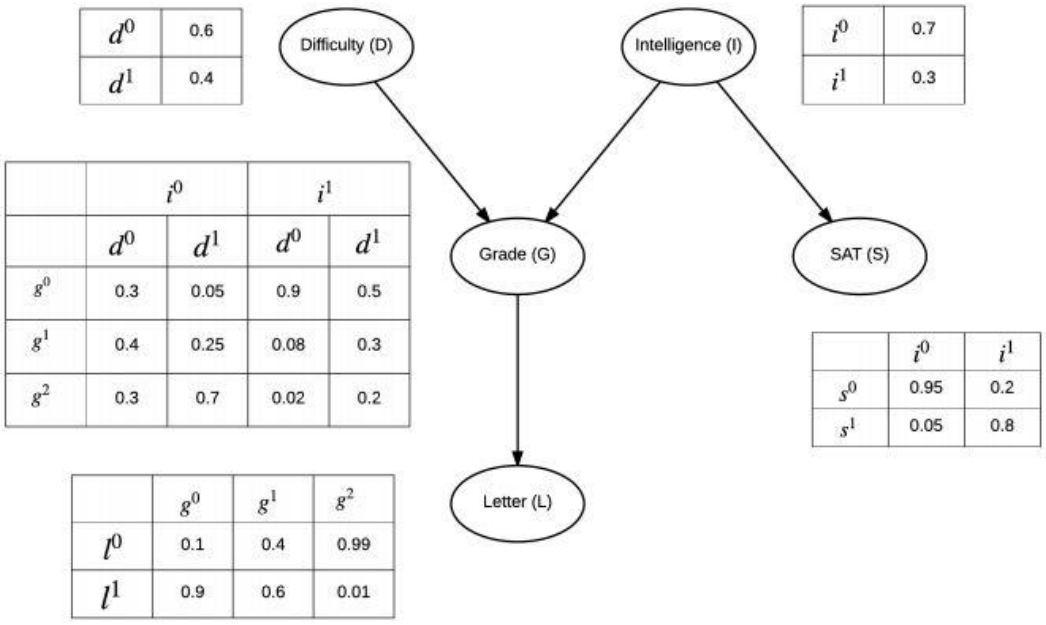
conda install -c ankurankan pgmpy

或

pip install pgmpy

（2）案例背景

以学生获得推荐信质量这样一个例子来进行贝叶斯网络的构造。具体有向图和概率表如下图所示：



	i^0		i^1	
	d^0	d^1	d^0	d^1
g^0	0.3	0.05	0.9	0.5
g^1	0.4	0.25	0.08	0.3
g^2	0.3	0.7	0.02	0.2

图 1 学生推荐信质量 DAG 和概率表

考试难度、个人聪明与否都会影响到个人成绩，另外个人聪明与否也会影响到 SAT 分数，而个人成绩好坏会直接影响到推荐信的质量。用 pgmpy 实现上述贝叶斯网络。

（3）实现步骤

1) 构建模型框架，指定各变量之间的依赖关系。
2) 构建各个节点和传入概率表并指定相关参数。
3) 将包含概率表的各节点添加到模型中。
4) 获取模型的条件概率分布和各节点之间的依赖关系。
5) 进行贝叶斯推断。

（4） 参考代码

1）导入 pgmpy 库

```
from pgmpy.factors.discrete import TabularCPD
from pgmpy.models import BayesianModel
from pgmpy.inference import VariableElimination
```

2）构建模型框架

```
student_model = BayesianModel([("D", "G"),
                               ("I", "G"),
                               ("G", "L"),
                               ("I", "S")])
```

3）构建各个节点和传入概率表

```
difficulty_cpd = TabularCPD(
            variable="D",# 节点名称
            variable_card=2,# 节点取值个数
            values=[[0.6, 0.4]]# 该节点的概率表
)
intel_cpd = TabularCPD(
            variable="I",
            variable_card=2,
            values=[[0.7, 0.3]]
)
grade_cpd = TabularCPD(
    variable="G",
    variable_card=3,
    values=[[0.3, 0.05, 0.9, 0.5],
    [0.4, 0.25, 0.08, 0.3],
    [0.3, 0.7, 0.02, 0.2]],
    evidence=["I", "D"], # 该节点的依赖节点
    evidence_card=[2, 2] # 依赖节点的取值个数
)
```

请据此写出变量 S 和 L 的节点，变量分别为 sat 分数和推荐信，概率如图 1 中所示。

4）添加到模型中

```
student_model.add_cpds(
    difficulty_cpd,
    intel_cpd,
    grade_cpd,
    letter 变量名,
    sat 变量名
)
```

5）条件概率分布和依赖关系

```
student_model.get_cpds()
student_model.get_independencies()
```

6）贝叶斯推断

```
student_infer = VariableElimination(student_model)
prob_G = student_infer.query(
            variables=["G"],
            evidence={"I": 1, "D": 0})
print(prob_G)
```

（5） 结果

推断聪明的学生在考试难度低时第一等成绩的概率高达 0.9

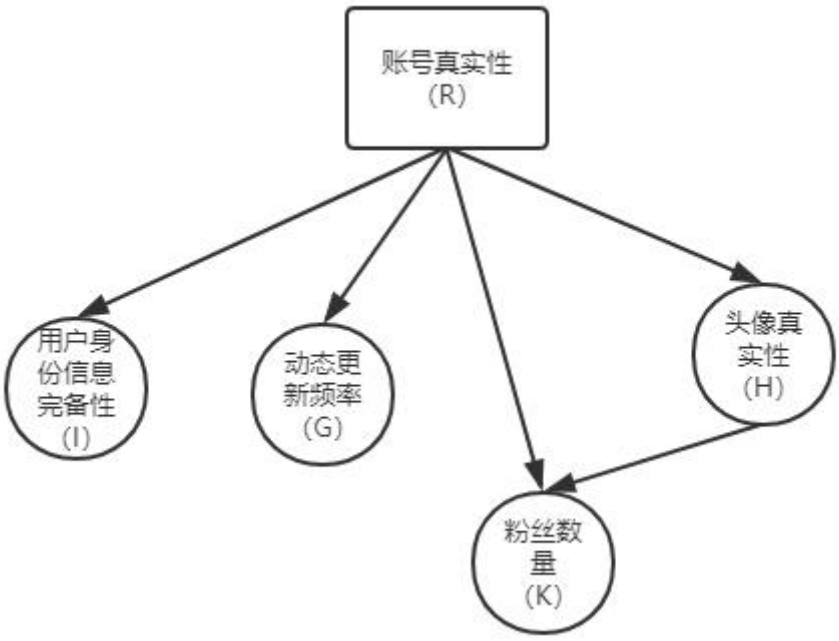
```
| G    |   phi(G) |
+======+==========+
| G(0) |   0.9000 |
+------+----------+
| G(1) |   0.0800 |
+------+----------+
| G(2) |   0.0200 |
+------+----------+
```

请修改代码后得出 I=1，D=0 时推荐信和 SAT 成绩的概率推断结果

2. 基于 pgmpy 进行数据训练

（1）案例背景

假设在社交网络中，我们需要通过头像真实性、粉丝数量、身份信息和动态更新频率来判断一个微博账号是否为真实账号。各特征属性之间的关系如下图所示：



1) 真实账号比非真实账号平均具有更大的动态更新频率、各大的粉丝数量以及更多的使用真实头像和更全面的用户身份信息（性别、职业、背景等）。
2) 动态更新频率与粉丝数量、动态更新频率、用户身份信息与是否使用真实头像在账号真实性给定的条件下是独立的。
3) 使用真实头像的用户比使用非真实头像的用户平均有更大的粉丝数量。

有向无环图(DAG)每个节点表示一个特征或者随机变量，特征之间的关系用箭头连线来表示，比如说动态的更新频率、粉丝数量、用户身份信息和头像真实性都会对一个微博账号的真实性有影响，而头像真实性又对粉丝数量有一定影响。

表 1 账号真实性条件概率表

R=0	0.4
R=1	0.6

账号真实性节点没有父节点，直接用先验概率来表示账号真实与否的概率。（R=0 虚假，R=1 真实）

表 2 头像真实性条件概率表

	R=0	R=1
H=0	0.9	0.2
H=1	0.1	0.8

表 2 是已知账号为真假地情况下，使用真实或虚假头像的概率分布。（H=0 虚假头像、H=1 真实头像）

表 3 动态更新频率条件概率表

	R=0	R=1
G=0	0.82	0.1
G=1	0.18	0.9

动态更新频率（G），0 表示更新较少，0 表示更新较多。

表 4 用户信息完备性条件概率表

	R=0	R=1
I=0	0.75	0.3
I=1	0.25	0.7

用户信息完备性（I），0 表示用户信息不完善，1 表示用户信息较为完备。

表 5 粉丝数量条件概率表

	R=0		R=1	
	H=0	H=1	H=0	H=1
K=0	0.89	0.01	0.6	0.9
K=1	0.11	0.99	0.4	0.1

粉丝数量（K），0 粉丝数量较少，1 表示粉丝数量较多。

（2）实验目的

1) 构造贝叶斯网络，实现账户真实性推断。

2) 假设使用模拟数据，训练模型。

（3）实现步骤

1) 利用上一实验的方法，构造贝叶斯网络。

2) 推断当账户为假时,粉丝数量较多的概率。

3) 生成模拟数据并以上述模型变量进行命名。

4) 基于数据进行模型训练，打印条件概率分布。

（4）参考代码

1)构建贝叶斯网络

和上一实验类似

```
user_model.add_cpds(
    account_cpd,
    infor_cpd,
    Dynamic_cpd,
    header_cpd,
    fannum_cpd
)
```

2）模型推断

推断当账户为假时,粉丝数量较多的概率，仿照上一实验完成。

3）模拟数据

```
import numpy as np
import pandas as pd
raw_data = np.random.randint(low=0, high=2, size=(1000, 5))
data = pd.DataFrame(raw_data, columns=["R", "I", "G", "H", "K"])
```

4）模型训练

```
from pgmpy.models import BayesianModel
from pgmpy.estimators import MaximumLikelihoodEstimator, BayesianEstimator
model = BayesianModel([("R", "I"),("R", "G"),("R", "H"),("R", "K"),("H", "K")])
```

```
model.fit(data, estimator=MaximumLikelihoodEstimator)
for cpd in model.get_cpds():
    # 打印条件概率分布
    print("CPD of {variable}:".format(variable=cpd.variable))
    print(cpd)
```

五、提交要求

1. 实验一提交代码及运行结果
   (1) 条件概率分布和依赖关系。
   (2) I=1，D=0 时推荐信和 SAT 成绩的概率推断结果。
2. 实验二提交代码及运行结果
   (1) 输出推断当账户为假时,粉丝数量较多的概率。
   (2) 打印使用模拟数据训练得到的条件概率分布。

注：两个题目可选做一个