

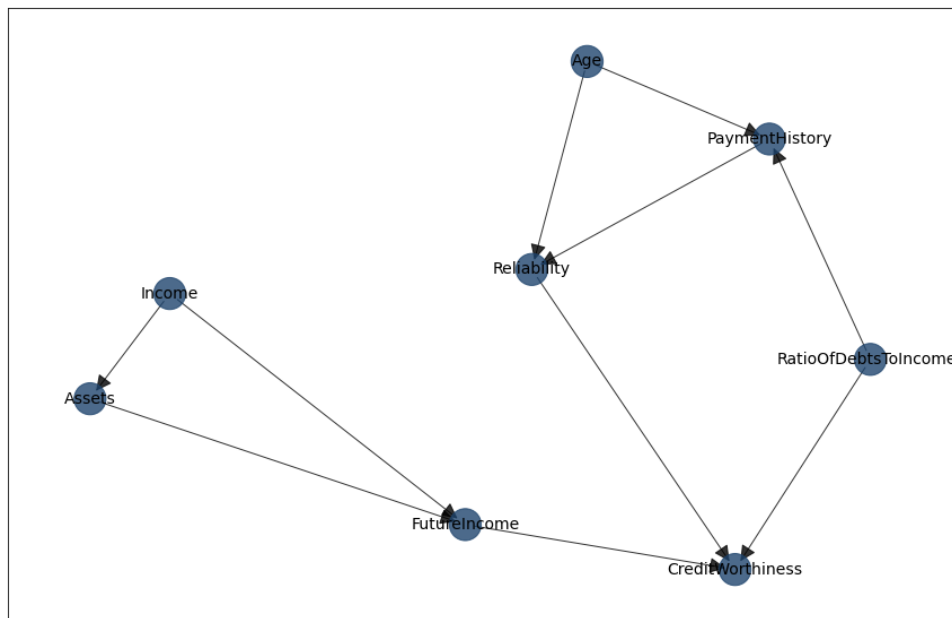
## Introduction to AI: Uncertainty 2

## Report 5

## Contents

Exercise:.....	1
Client is 20 years old with excellent payment history .....	3
Has high income, high assets and low ratio .....	3
Is 70 years old with medium assets and low income .....	3
Has high future income and low ratio .....	3
Is 22 with low income, low assets and excellent payment history .....	3
Collab Sources:.....	4
Feedback: .....	4

## Exercise:



```

vertices=['Income', 'PaymentHistory', 'Age', 'RatioOfDebtsToIncomes', 'Assets', 'Reliability', 'FutureIncome', 'CreditWorthiness']
edges=[
('PaymentHistory', 'Reliability'),
('Age', 'Reliability'),
('Age', 'PaymentHistory'),
('RatioOfDebtsToIncomes', 'PaymentHistory'),
('Income', 'Assets'),
('Assets', 'FutureIncome'),
('Income', 'FutureIncome'),
('Reliability', 'CreditWorthiness'),
('FutureIncome', 'CreditWorthiness'),
('RatioOfDebtsToIncomes', 'CreditWorthiness')]

DAG1 = bnlearn.make_DAG(edges1)
bnlearn.plot(DAG1, interactive = True)

```

```

cpt_Income = TabularCPD(variable='Income', variable_card=3,
                        values=[[0.5],
                                [0.2],
                                [0.3]],
                        evidence=[], evidence_card=[])

cpt_PaymentHistory = TabularCPD(variable='PaymentHistory', variable_card=3,
                                values=[[0.1,0.3,0.6,0.1,0.3,0.6],
                                        [0.3,0.5,0.2,0.3,0.5,0.2],
                                        [0.6,0.2,0.2,0.6,0.2,0.2]],
                                evidence=[ 'Age', 'RatioOfDebtsToIncomes'], evidence_card=[3,2]
                                )

cpt_Age = TabularCPD(variable='Age', variable_card=3,
                    values=[[0.1],
                            [0.3],
                            [0.6]],
                    evidence=[], evidence_card=[])

cpt_RatioOfDebtsToIncomes = TabularCPD(variable='RatioOfDebtsToIncomes', variable_card=2,
                                       values=[[0.9],
                                               [0.1]],
                                       evidence=[], evidence_card=[])

cpt_Assets = TabularCPD(variable='Assets', variable_card=3,
                        values=[[0.1,0.3,0.6],
                                [0.3,0.5,0.2],
                                [0.6,0.2,0.2]],
                        evidence=[ 'Income'], evidence_card=[3])

cpt_Reliability = TabularCPD(variable='Reliability', variable_card=2,
                             values=[[0.9,0.9,0.9,0.7,0.7,0.4,0.4,0.4],
                                     [0.1,0.1,0.1,0.3,0.3,0.3,0.6,0.6,0.6]],
                             evidence=[ 'PaymentHistory', 'Age'], evidence_card=[3,3]
                             )

cpt_FutureIncome = TabularCPD(variable='FutureIncome', variable_card=2,
                              values=[[0.8,0.6,0.6,0.6,0.6,0.6,0.4,0.2,0.2],
                                      [0.2,0.4,0.4,0.4,0.4,0.4,0.6,0.8,0.8]],
                              evidence=[ 'Income', 'Assets'], evidence_card=[3,3]
                              )

cpt_CreditWorthiness = TabularCPD(variable='CreditWorthiness', variable_card=2,
                                   values=[[0.9,0.7,0.5,0.7,0.5,0.3,0.3],
                                           [0.1,0.3,0.5,0.3,0.5,0.7,0.9]],
                                   evidence=[ 'FutureIncome', 'RatioOfDebtsToIncomes', 'Reliability'], evidence_card=[2,2,2]
                                   )

DAG1 = bnlearn.make_DAG(DAG1.CPD=[cpt_Income,cpt_PaymentHistory,cpt_Age,cpt_RatioOfDebtsToIncomes,cpt_Assets,cpt_Reliability,cpt_FutureIncome,cpt_CreditWorthiness])

```

Client is 20 years old with excellent payment history

```
q1= bnlearn.inference.fit(DAG1, variables=['CreditWorthiness'], evidence={'Age':0, 'PaymentHistory':0})

[bnlearn] >Variable Elimination..
[bnlearn] >Data is stored in [query.df]
+-----+-----+
| | CreditWorthiness | p |
+-----+-----+
| 0 | 0 | 0.61936 |
+-----+-----+
| 1 | 1 | 0.38064 |
+-----+-----+
```

Has high income, high assets and low ratio

```
q2= bnlearn.inference.fit(DAG1, variables=['CreditWorthiness'], evidence={'Income':0, 'Assets':0, 'RatioOfDebtsToIncomes':1})

[bnlearn] >Variable Elimination..
[bnlearn] >Data is stored in [query.df]
+-----+-----+
| | CreditWorthiness | p |
+-----+-----+
| 0 | 0 | 0.49744 |
+-----+-----+
| 1 | 1 | 0.50256 |
+-----+-----+
```

Is 70 years old with medium assets and low income

```
q3= bnlearn.inference.fit(DAG1, variables=['CreditWorthiness'], evidence={'Age':2, 'Assets':1, 'Income':2})

[bnlearn] >Variable Elimination..
[bnlearn] >Data is stored in [query.df]
+-----+-----+
| | CreditWorthiness | p |
+-----+-----+
| 0 | 0 | 0.49912 |
+-----+-----+
| 1 | 1 | 0.50088 |
+-----+-----+
```

Has high future income and low ratio

```
q4= bnlearn.inference.fit(DAG1, variables=['CreditWorthiness'], evidence={'FutureIncome':0, 'RatioOfDebtsToIncomes':1})

[bnlearn] >Variable Elimination..
[bnlearn] >Data is stored in [query.df]
+-----+-----+
| | CreditWorthiness | p |
+-----+-----+
| 0 | 0 | 0.5624 |
+-----+-----+
| 1 | 1 | 0.4376 |
+-----+-----+
```

Is 22 with low income, low assets and excellent payment history

```
q5= bnlearn.inference.fit(DAG1, variables=['CreditWorthiness'], evidence={'Age':1, 'Income':2, 'Assets':2, 'PaymentHistory':0})

[bnlearn] >Variable Elimination..
[bnlearn] >Data is stored in [query.df]
+-----+-----+
| | CreditWorthiness | p |
+-----+-----+
| 0 | 0 | 0.555782 |
+-----+-----+
| 1 | 1 | 0.444218 |
+-----+-----+
```

Asked about last part with no reply: attempt at naive bayes testing and pruning

```
DAG1 = bnlearn.make_DAG(DAG1,CPD=[cpt_Income,
                                cpt_PaymentHistory,cpt_Age,
                                cpt_RatioOfDebtsToIncomes,cpt_Assets,
                                cpt_Reliability,cpt_FutureIncome,
                                cpt_CreditWorthiness],checkmodel=True)
bnlearn.print_CPD(DAG1, checkmodel=True)
bnlearn.plot(DAG1)

model = bnlearn.structure_learning.fit(df, methodtype='naivebayes', root_node="CreditWorthiness")
model = bnlearn.independence_test(model, df, prune=False)
bnlearn.plot(model)
model = bnlearn.independence_test(model, df, prune=True)
bnlearn.plot(model)
```

## Collab Sources:

Original Collab:

<https://colab.research.google.com/drive/1Kb6b9aeryr5WBuXteHtikuSHISTMULKz#scrollTo=wH6Fjpr3-iw>

Link to my own collab:

<https://colab.research.google.com/drive/13DezZDFhFtLfaWrmHQL24OadwFT4pvZM?authuser=1#scrollTo=wH6Fjpr3-iw>

## Feedback:

Thank you for the time extension, the variable credit worthiness was not explained in terms of 0 1, the ratio of debts to income was given twice (see below, line 3, line 8).

```
income - 3 levels: 0:high, 1:medium, 2:low
assets - 3 levels: 0:high, 1:medium, 2:low
ratio of debts to income - 2 levels: 0:high, 1:low
payment history - 3 levels: 0:excellent, 1:acceptable, 2:unacceptable
age - 3 levels: 0:16-21, 1:22-64, 2:over65
reliability - 2 levels: 0:reliable, 1:unreliable
future income - 2 levels: 0:promising, 1:not_promising
ratio of debts to income - 2 levels: 0:low, 1:high
```