Hugin: a Bayesian Network based decision tool

Gianluca Corrado and Andrea Passerini

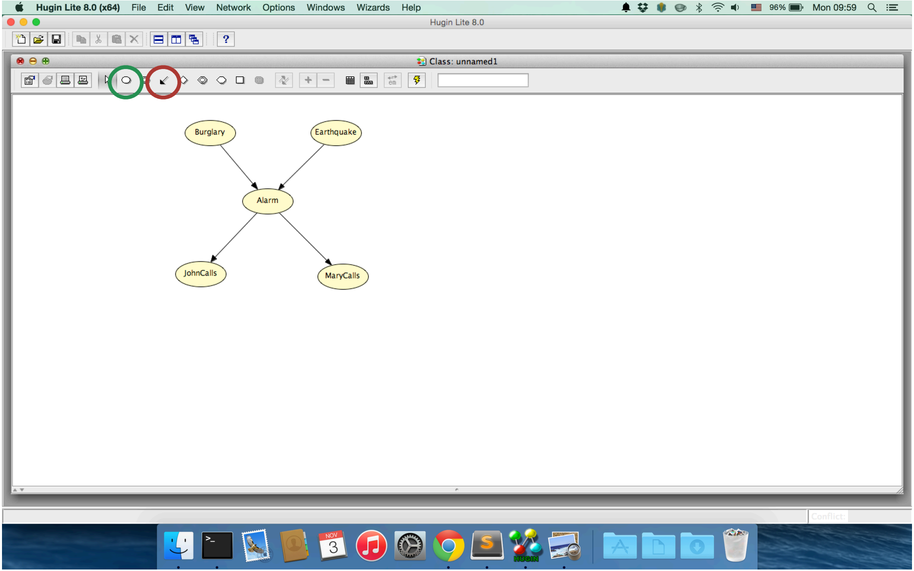
gianluca.corrado@unitn.it
passerini@disi.unitn.it

Machine Learning

Downloading and Installing

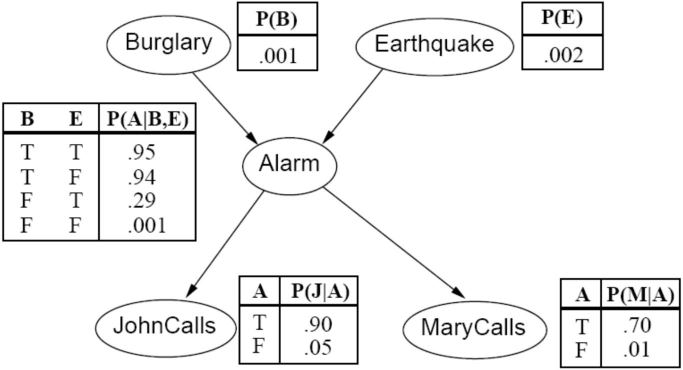
- FREE HuginLite
- The free trial version is limited to handle max. 50 states and learn from max. 500 cases
- It is prohibited to use the free Hugin Lite for any other purpose than the demonstration of capabilities and proof of concept
- `http://www.hugin.com/productsservices/demo/hugin-lite`

Defining Nodes and Links



Defining the States

- By clicking on a state holding the CRTL key
- Insert the probability value associated to each state for all the nodes.



P(B)
.001

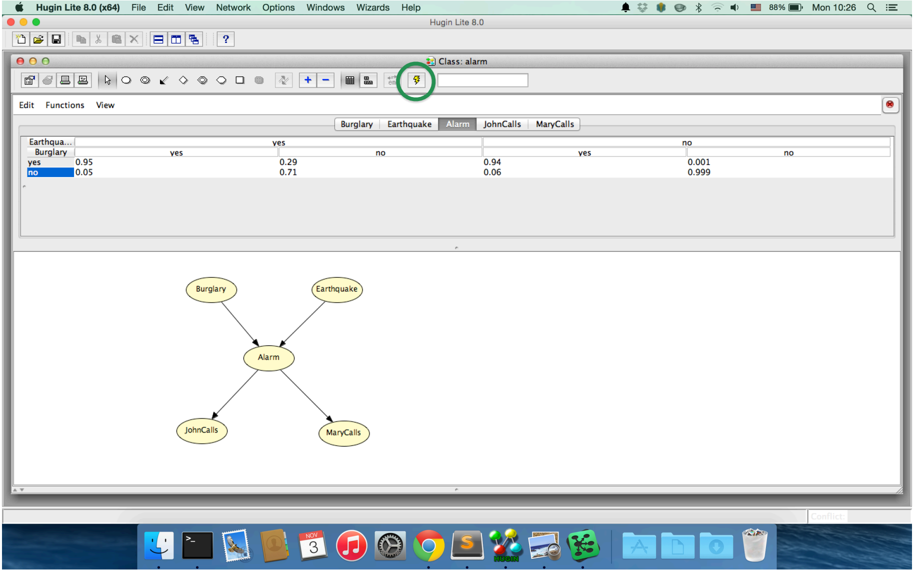
P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

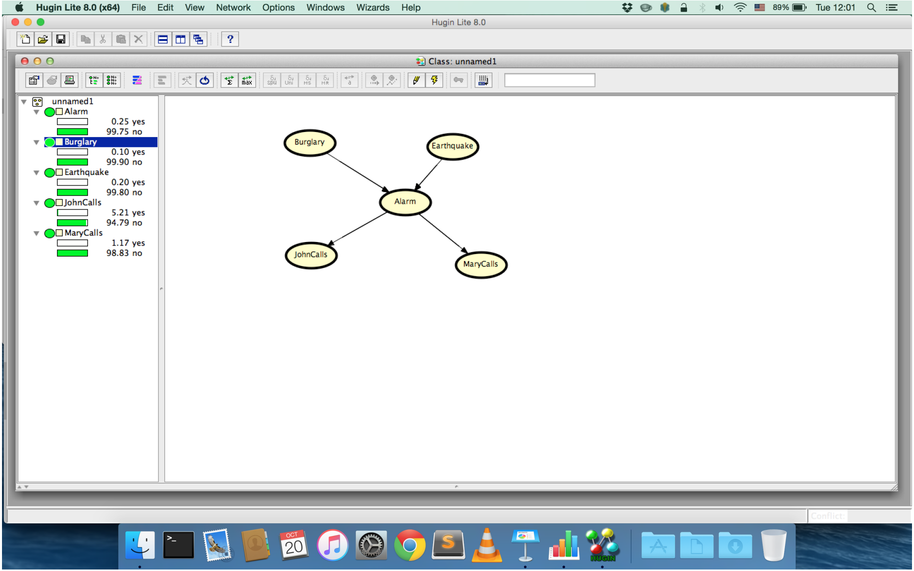
A	P(M|A)
T	.70
F	.01

Compiling the Network



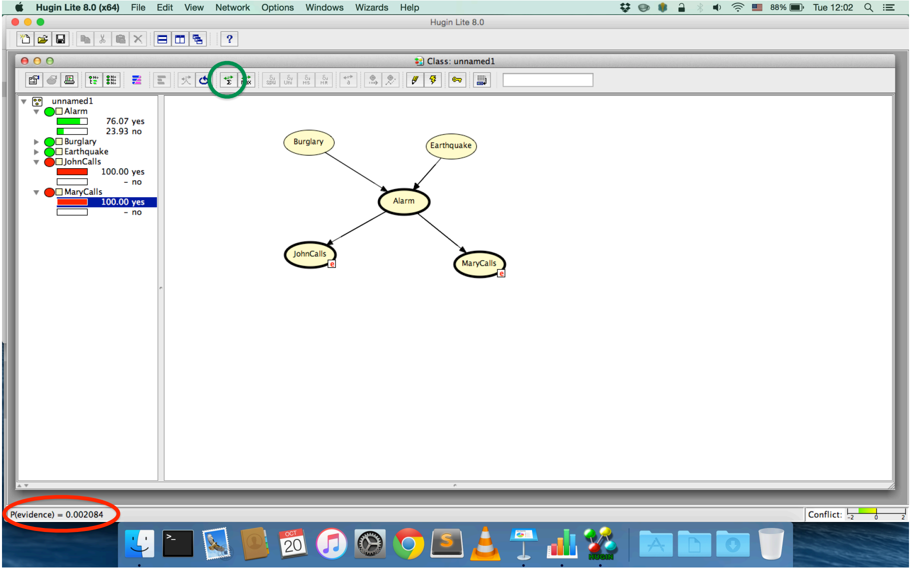
Earthqua...	yes		no	
Burglary	yes	no	yes	no
yes	0.95	0.29	0.94	0.001
no	0.05	0.71	0.06	0.999

Running the Network



```
unnamed1
  Alarm
      0.25 yes
     99.75 no
  Burglary
      0.10 yes
     99.90 no
  Earthquake
      0.20 yes
     99.80 no
  JohnCalls
      5.21 yes
     94.79 no
  MaryCalls
      1.17 yes
     98.83 no
```

P(evidence)



```
unnamed1
  Alarm
     76.07 yes
     23.93 no
  Burglary
  Earthquake
  JohnCalls
    100.00 yes
         – no
  MaryCalls
    100.00 yes
         – no
```

P(evidence) = 0.002084

Computing the probability of a combination of states

- We want to compute $P(alarm = "yes", johncalls = "yes" | burglary = "yes")$
- Exploting that $P(A, B) = P(A|B)P(B)$

$$P(alarm = "yes", johncalls = "yes" | burglary = "yes") = \\ = \frac{P(alarm = "yes", johncalls = "yes", burglary = "yes")}{P(burglary = "yes")}$$

$$P(alarm = "yes", johncalls = "yes" | burglary = "yes") = \\ = \frac{0.000846}{0.001} = 0.846$$

Learning from Data

- Select Wizards, Learning Wizard
- Load the training file (small_asia.dat)
- In structure constraints import model information from ChestClinic.net
- Select a learning algorithm
- Give to each state a prior of 1
- RUN the learning algorithm
- Compile the learned network

Analysis Wizard

- Select Wizards, Analysis Wizard
- Sample 100 new examples according to the learned network
- Check them in Data Source
- Analyze the quality of the generated data in Data Accuracy
- Clear the Data Source and Load the test file (test_asia_small.dat)
- Analyze the performance of classification of the learned network

Assignment

- Consider the data file iris.dat
- the file contains 4 continuous features
- and one label (*type*)
- Random sample a train (100 examples) and a test (the remaining) sets (balanced w.r.t. *type*)
- Repeat the sampling 3 times (generate 3 training sets and 3 test sets)
- Learn the Bayesian networks (using NPC)
- Test the learned Bayesian networks
- Write a short report (2-3 pages) summarizing the methodology used and the results obtained.

Assignment

- After completing the assignment submit it via email
- Send an email to gianluca.corrado@unitn.it (cc: passerini@disi.unitn.it)
- Subject: HuginSubmit2016
- Attachment: id_name_surname.zip containing:
    - the script used to sample the data (named sampler.ext)
    - the train and test sets (named train_X.dat and test_X.dat respectively)
    - the learned networks (named npc_X.net)
    - the report (named report.pdf)
    - X is the reference to the sampled train and test sets.

NOTE

- No group work
- This assignment is mandatory in order to take the oral exam