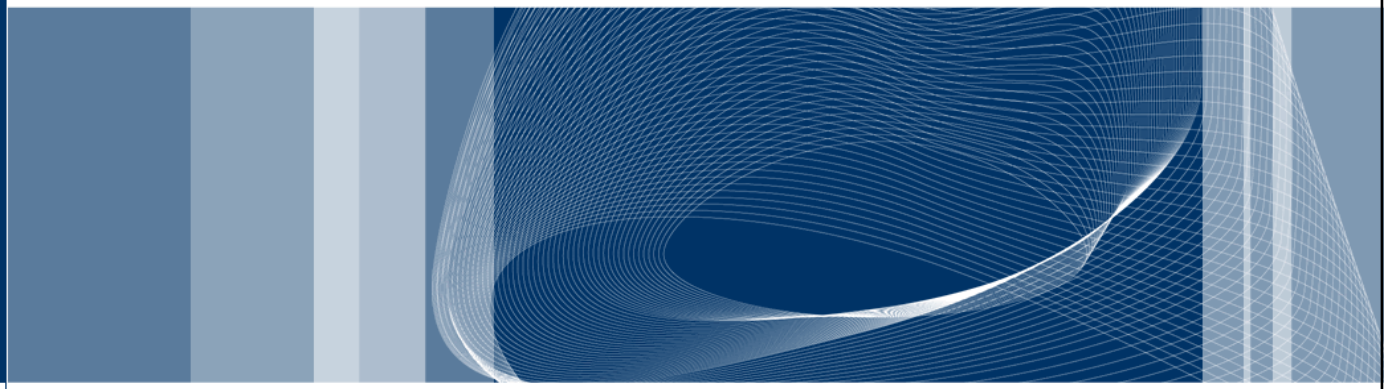
POLITECNICO DI MILANO



Naïve Bayes & Bayesian Networks

Data Mining and Text Mining

Prof. Pierluca Lanzi

Bayes Probability

- Conditional Probability:

$$P(C|A) = \frac{P(A \wedge C)}{P(A)} \qquad P(A|C) = \frac{P(A \wedge C)}{P(C)}$$

- Bayes theorem,

$$P(C|A) = \frac{P(A|C)P(C)}{P(A)}$$

- A priori probability of C, P(C), is the probability of event before evidence is seen
- A posteriori probability of C, P(C|A), is the probability of event after evidence is seen

Naïve Bayes Classifiers

- What's the probability of the class given an example?
- An example is represented as a tuple of attributes

$$\vec{x} = \langle x_1, \ldots, x_n \rangle$$

- Given the target y (identifying the class value for the instance) we are looking for the class with the highest probability for X

$$class = \arg\max_y P(y|\vec{x})$$

- Given the target y and the example x described by n attributes, Bayes Theorem says that

$$P(y|\vec{x}) = \frac{P(\vec{x}|y)P(y)}{P(\vec{x})}$$

- Naïve Bayes classifiers assume that attributes are statistically independent

- Thus, evidence splits into parts that are independent

$$P(y|\vec{x}) = \frac{P(x_1|y)\cdots P(x_n|y)P(y)}{P(\vec{x})}$$

Naïve Bayes for Classification

- Training
  - Count the frequency of tuples (x_i,y) for each attribute value x_i and each class value y in the dataset
  - Use the counts to compute estimates for the class probability P(y) and the conditional probability P(x_i|y)
- Testing
  - Given an example x, computes the most likely class as

$$
\begin{aligned}
class &= \arg\max_y P(y|\vec{x}) \\
&= \arg\max_y \frac{P(x_1|y)\cdots P(x_n|y)P(y)}{P(\vec{x})} \\
&= \arg\max_y P(x_1|y)\cdots P(x_n|y)P(y)
\end{aligned}
$$

- Two assumptions
  - Attributes are equally important
  - Attribute are statistically independent

- Statistically independent means
  - That knowing the value of one attribute x_j says nothing about the value of another x_i if the class y is known, that is, $P(x_i|x_j,y) = P(x_i|y)$
  - Independence assumption is almost never correct! But the scheme works well in practice

The Weather Dataset

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Computing Probabilities

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

What is the assigned class?

Using the Probabilities for Classification

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

$$\begin{aligned}
\text{Likelyhood of yes} &= P(Sunny|yes)P(Cool|yes)P(High|yes)P(True|yes)P(yes) \\
&= 0.0053 \\
\text{Likelyhood of no} &= P(Sunny|no)P(Cool|no)P(High|no)P(True|no)P(no) \\
&= 0.0206
\end{aligned}$$

- The sum of the two values should be one. Thus, we convert the two values into actual probabilities using normalization:

$$\begin{aligned}
P(yes|Sunny, Cool, High, True) &= 0.0053/(0.0053 + 0.0206) = 0.205 \\
P(no|Sunny, Cool, High, True) &= 0.0206/(0.0053 + 0.0206) = 0.795
\end{aligned}$$

The "Zero-Frequency Problem"

- What if an attribute value does not occur with every class value? (for instance, "Humidity = high" for class "yes")
- The corresponding probability will be zero,

$$P(Humidity = high|yes) = 0$$

- A posteriori probability will also be zero!
  (No matter how likely the other values are!)

$$P(yes|\langle \ldots,\ Humidity = high,\ \ldots \rangle) = 0$$

- Typical remedy is to add 1 to count for every (attribute value, class) pair
  - Process called *smoothing*.
  - Adding 1 is called a *Laplace estimator*
- Resulting probabilities will never be zero! It also stabilizes probability estimates

Modified Probability Estimates

- Sometimes, it is more appropriate to adda constant different from one; for example, we can add the same μ/3 to each count

$$\frac{2+\mu/3}{9+\mu} \qquad \frac{4+\mu/3}{9+\mu} \qquad \frac{3+\mu/3}{9+\mu}$$

Sunny Overcast Rainy

- μ here acts as a regularization (hyper)parameter
- If background knowledge is available on the frequency of certain values, we can add different weights to each count

$$\frac{2+\mu p_1}{9+\mu} \qquad \frac{4+\mu p_2}{9+\mu} \qquad \frac{3+\mu p_3}{9+\mu}$$

- ($p_1+p_2+p_3=1$) are prior estimates on probability of each values.

Example (1)

- Suppose we want to compute the probablities of a Naïve Bayes classifier using the Laplace estimator for the dataset:

location	size	pets	value
good	small	yes	high
good	big	no	high
good	big	no	high
bad	medium	no	medium
good	medium	only cats	medium
good	small	only cats	medium
bad	medium	yes	medium
bad	small	yes	low
bad	medium	yes	low
bad	small	no	low

Example (2)

- As done in the example involving the weather dataset first we count the occurrences and add one due to the Laplace estimator

Location				Size				Pets			
	Class high	Class medium	Class low		Class high	Class medium	Class low		Class high	Class medium	Class low
Good	3+1	2+1	0+1	Small	1+1	1+1	2+1	Yes	1+1	1+1	2+1
Bad	0+1	2+1	3+1	Medium	0+1	3+1	1+1	No	2+1	1+1	1+1
				Big	2+1	0+1	0+1	Only Cats	0+1	2+1	0+1

Example (3)

- Next, we can compute the frequencies

Location				Size				Pets			
	Class high	Class medium	Class low		Class high	Class medium	Class low		Class high	Class medium	Class low
Good	4/5	3/6	1/5	Small	2/6	2/7	3/6	Yes	2/6	2/7	3/6
Bad	1/5	3/6	4/5	Medium	1/6	4/7	2/6	No	3/6	2/7	2/6
				Big	3/6	1/7	1/6	Only Cats	1/6	3/7	1/6

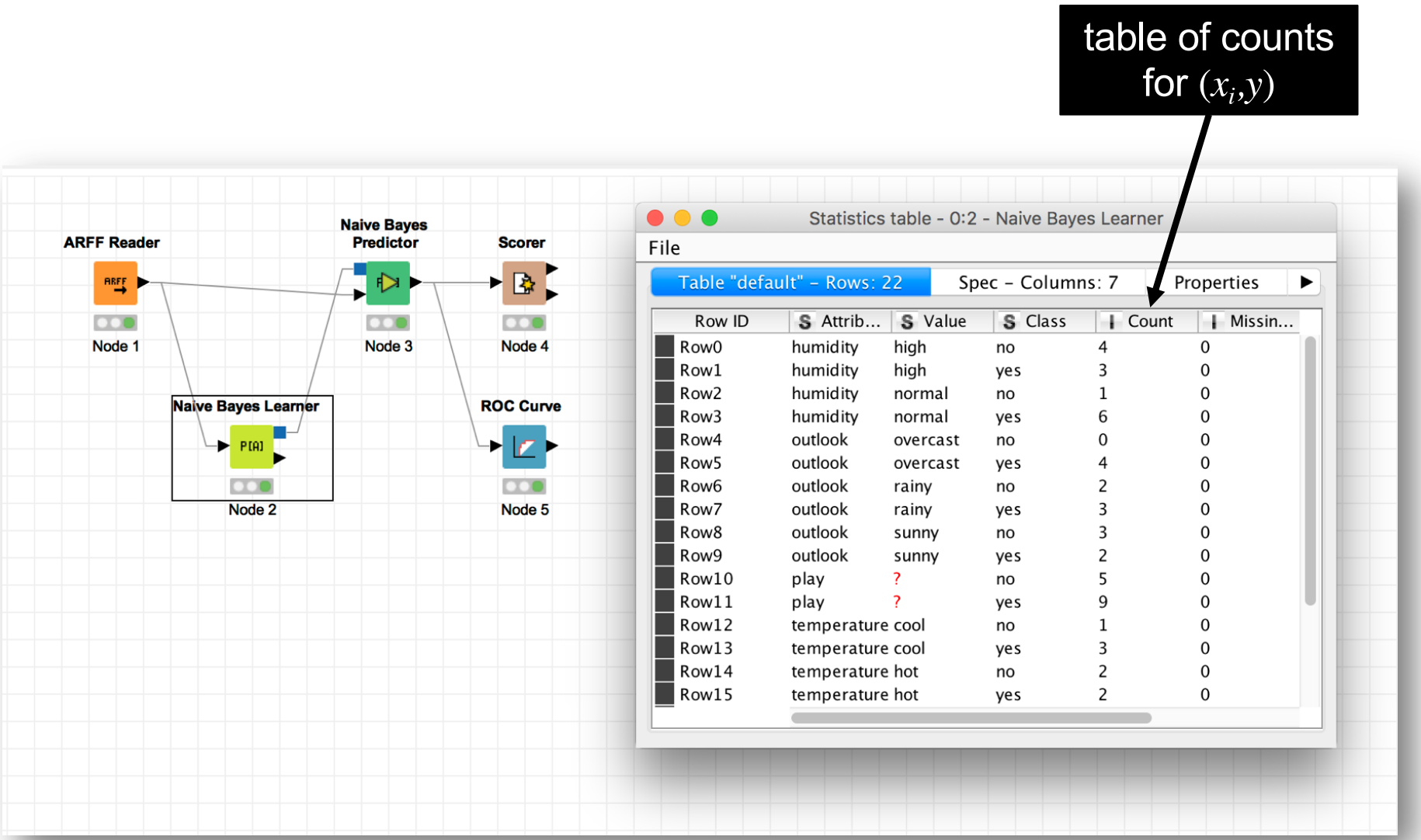
- Note that, Laplace estimator is not applied to the class, that is,
  - P(high) = 3/10
  - P(medium) = 4/10
  - P(low) = 3/10

How does Naïve Bayes deal with missing values?

- Training
  - Instance is not included in frequency count for attribute value-class combination
- Testing
  - The attribute will be omitted from calculation

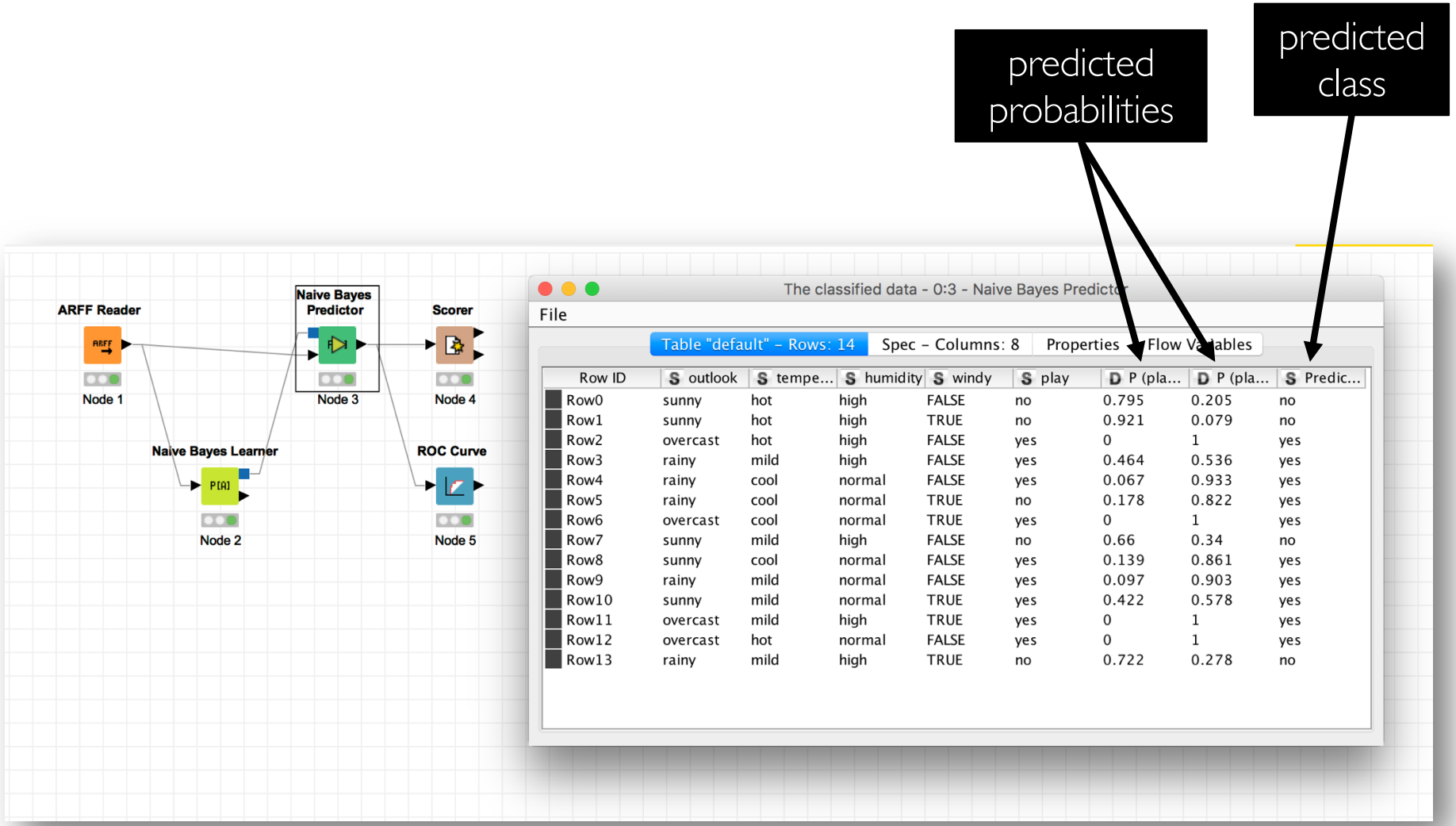
Outlook	Temperature	Humidity	Windy	Play
?	Cool	High	True	?

Likelihood of "yes" = 3/9 x 3/9 x 3/9 x 9/14 = 0.0238
Likelihood of "no" = 1/5 x 4/5 x 3/5 x 5/14 = 0.0343
P("yes" | ?, Cool, High, True) = 0.0238 / (0.0238 + 0.0343) = 41%
P("no" | ?, Cool, High, True) = 0.0343 / (0.0238 + 0.0343) = 59%



Row ID	Attrib...	Value	Class	Count	Missin...
Row0	humidity	high	no	4	0
Row1	humidity	high	yes	3	0
Row2	humidity	normal	no	1	0
Row3	humidity	normal	yes	6	0
Row4	outlook	overcast	no	0	0
Row5	outlook	overcast	yes	4	0
Row6	outlook	rainy	no	2	0
Row7	outlook	rainy	yes	3	0
Row8	outlook	sunny	no	3	0
Row9	outlook	sunny	yes	2	0
Row10	play	?	no	5	0
Row11	play	?	yes	9	0
Row12	temperature	cool	no	1	0
Row13	temperature	cool	yes	3	0
Row14	temperature	hot	no	2	0
Row15	temperature	hot	yes	2	0

Example of Naïve Bayes using KNIME



Row ID	outlook	tempe...	humidity	windy	play	P (pla...	P (pla...	Predic...
Row0	sunny	hot	high	FALSE	no	0.795	0.205	no
Row1	sunny	hot	high	TRUE	no	0.921	0.079	no
Row2	overcast	hot	high	FALSE	yes	0	1	yes
Row3	rainy	mild	high	FALSE	yes	0.464	0.536	yes
Row4	rainy	cool	normal	FALSE	yes	0.067	0.933	yes
Row5	rainy	cool	normal	TRUE	no	0.178	0.822	yes
Row6	overcast	cool	normal	TRUE	yes	0	1	yes
Row7	sunny	mild	high	FALSE	no	0.66	0.34	no
Row8	sunny	cool	normal	FALSE	yes	0.139	0.861	yes
Row9	rainy	mild	normal	FALSE	yes	0.097	0.903	yes
Row10	sunny	mild	normal	TRUE	yes	0.422	0.578	yes
Row11	overcast	mild	high	TRUE	yes	0	1	yes
Row12	overcast	hot	normal	FALSE	yes	0	1	yes
Row13	rainy	mild	high	TRUE	no	0.722	0.278	no

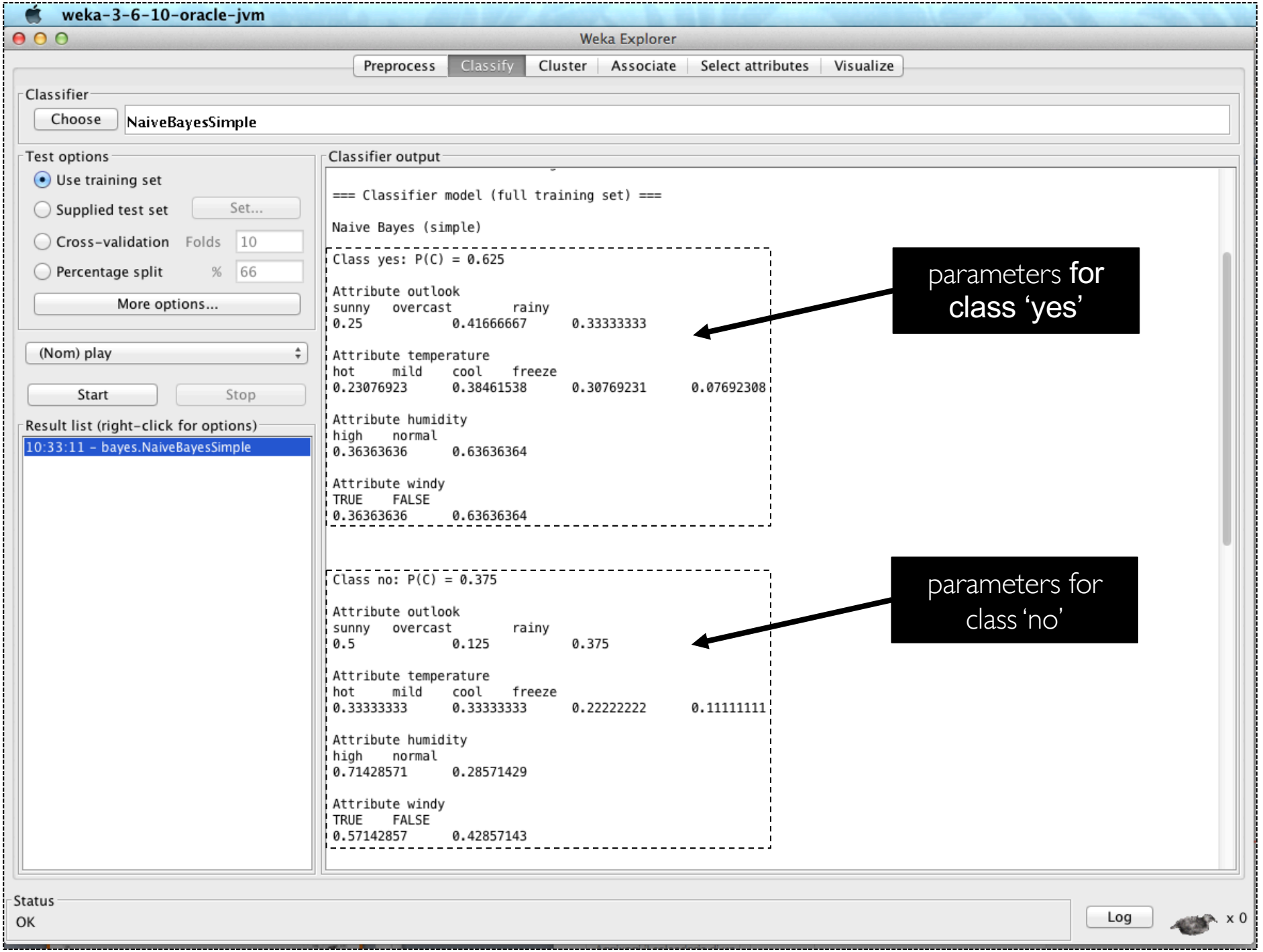
Example of Naïve Bayes using KNIME

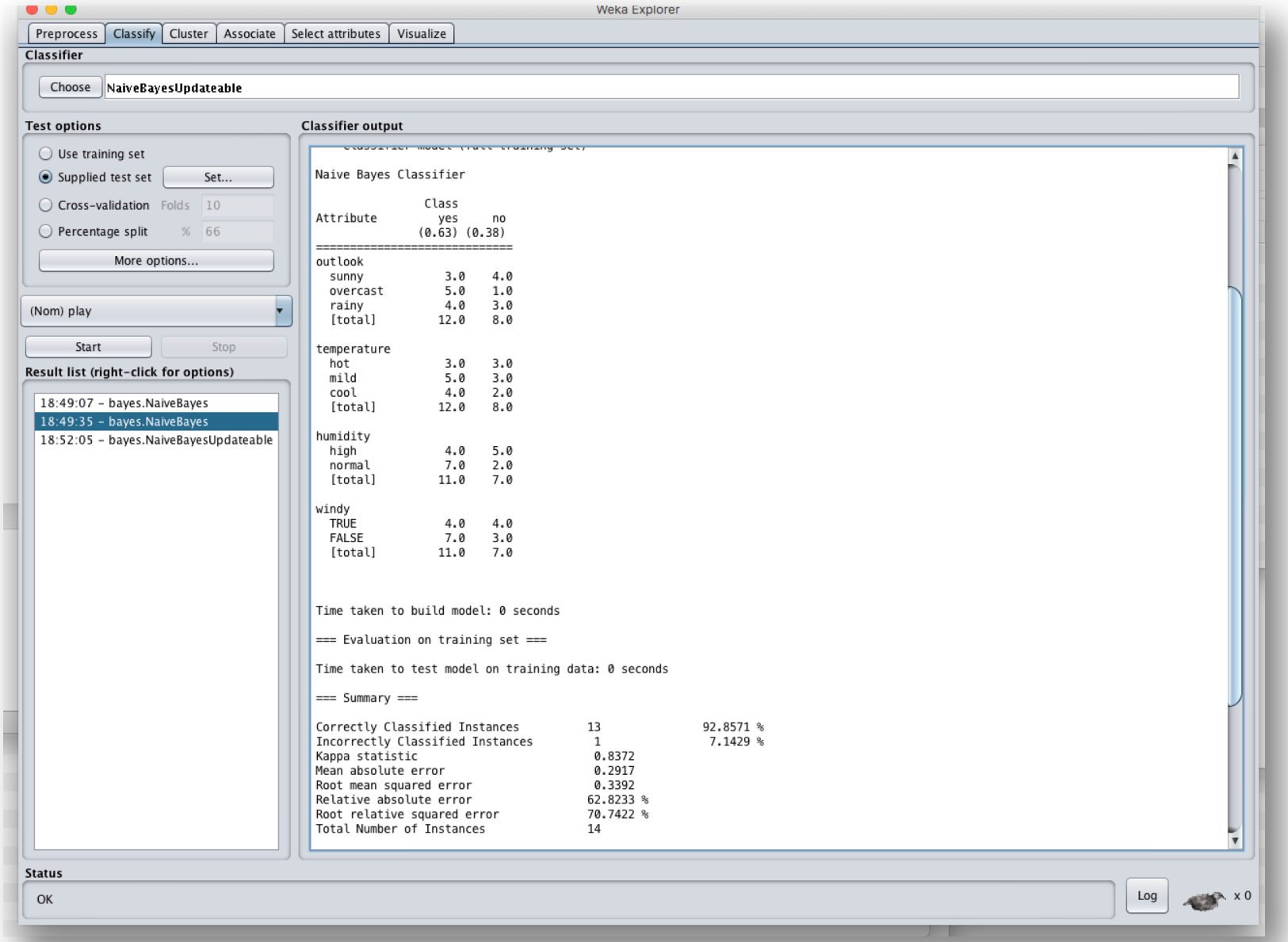
Another Version of the Weather Data


```
@relation weather.symbolic

@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool, freeze}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
...
```


weka-3-6-10-oracle-jvm
Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayesSimple
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) play
Start
Stop
Result list (right-click for options)
10:33:11 - bayes.NaiveBayesSimple
Classifier output
=== Classifier model (full training set) ===
Naive Bayes (simple)
Class yes: P(C) = 0.625
Attribute outlook
sunny overcast rainy
0.25 0.41666667 0.33333333
Attribute temperature
hot mild cool freeze
0.23076923 0.38461538 0.30769231 0.07692308
Attribute humidity
high normal
0.36363636 0.63636364
Attribute windy
TRUE FALSE
0.36363636 0.63636364
Class no: P(C) = 0.375
Attribute outlook
sunny overcast rainy
0.5 0.125 0.375
Attribute temperature
hot mild cool freeze
0.33333333 0.33333333 0.22222222 0.11111111
Attribute humidity
high normal
0.71428571 0.28571429
Attribute windy
TRUE FALSE
0.57142857 0.42857143
parameters for class 'yes'
parameters for class 'no'
Status
OK
Log
x 0



Example of Naïve Bayes with Laplace Estimator

Numeric Attributes in Naïve Bayes

- So far, we applied Naïve Bayes to categorical data. What if some (or all) of the attributes are numeric? Two options
- Discretize the data to make it binary or discrete
  -
- Compute a probability density for each class
  - Assume parametric form for distribution and estimate its parameters. E.g., assume attribute values for class follow Gaussian distribution
  - Directly estimate probability density from the data. E.g., use kernel smoothing to estimate density of values along axis for class

Numeric Attributes

- We assume that the attributes have a normal or Gaussian probability distribution (given the class)
- The probability density function for the normal distribution is defined by two parameters, the mean and the standard deviation
- Sample mean,

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- Standard deviation,

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2}$$

- Then the density function f(x) is

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Statistics for Weather data with numeric temperature

Outlook			Temperature		Humidity		Windy			Play	
	Yes	*No*	*Yes*	*No*	*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	64, 68,	65, 71,	65, 70,	70, 85,	False	6	2	9	5
Overcast	4	0	69, 70,	72, 80,	70, 75,	90, 91,	True	3	3		
Rainy	3	2	72, …	85, …	80, …	95, …					
Sunny	2/9	3/5	μ =73	μ =75	μ=79	μ=86	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	σ =6.2	σ =7.9	σ=10.2	σ=9.7	True	3/9	3/5		
Rainy	3/9	2/5									

$$f(temperature = 66|yes) = \frac{1}{\sqrt{2\pi}6.2} e^{-\frac{(66-73)^2}{2\times 6.2^2}} = 0.0340$$

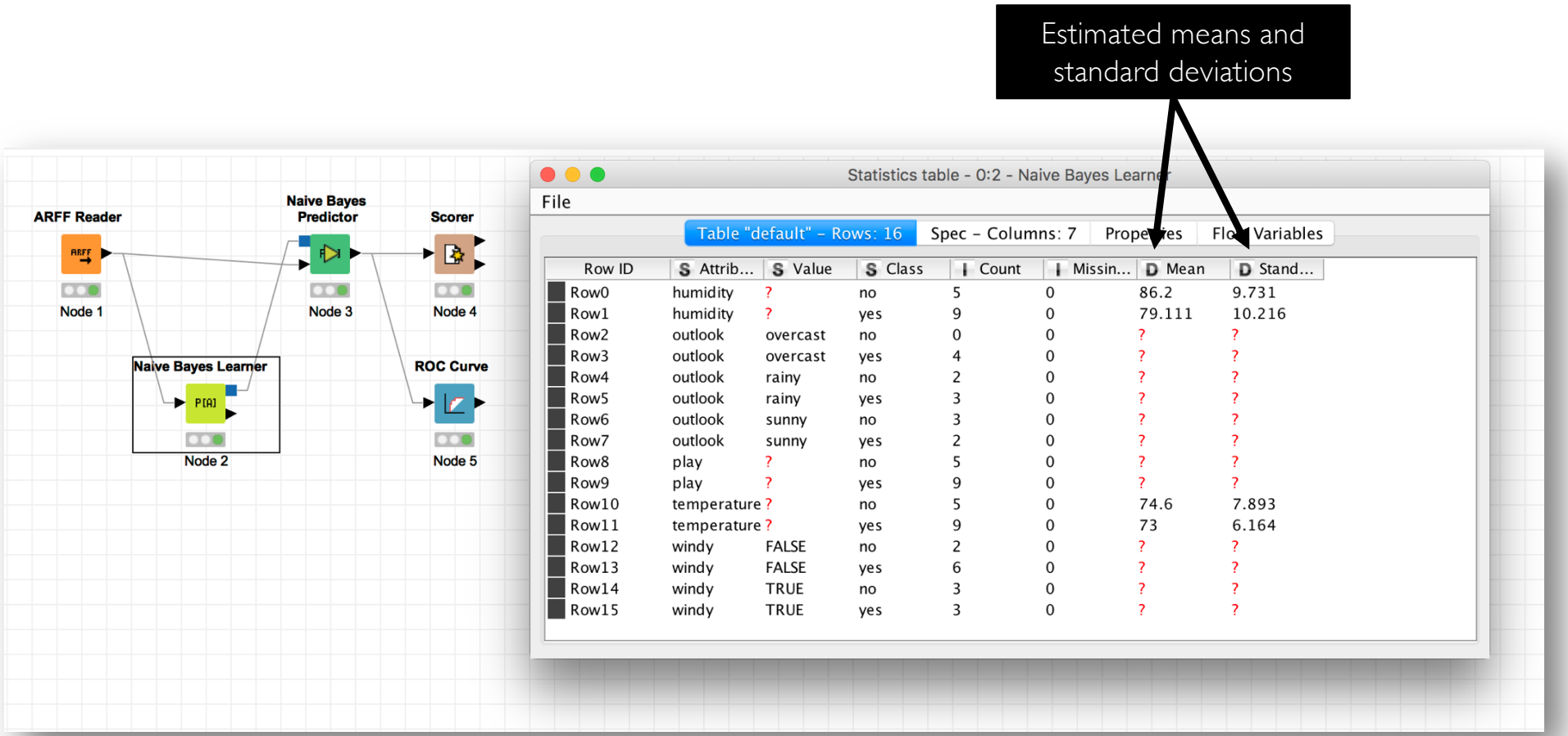
Classifying a new day

- A new day,

Outlook	Temperature	Humidity	Windy	Play
Sunny	66	90	true	?

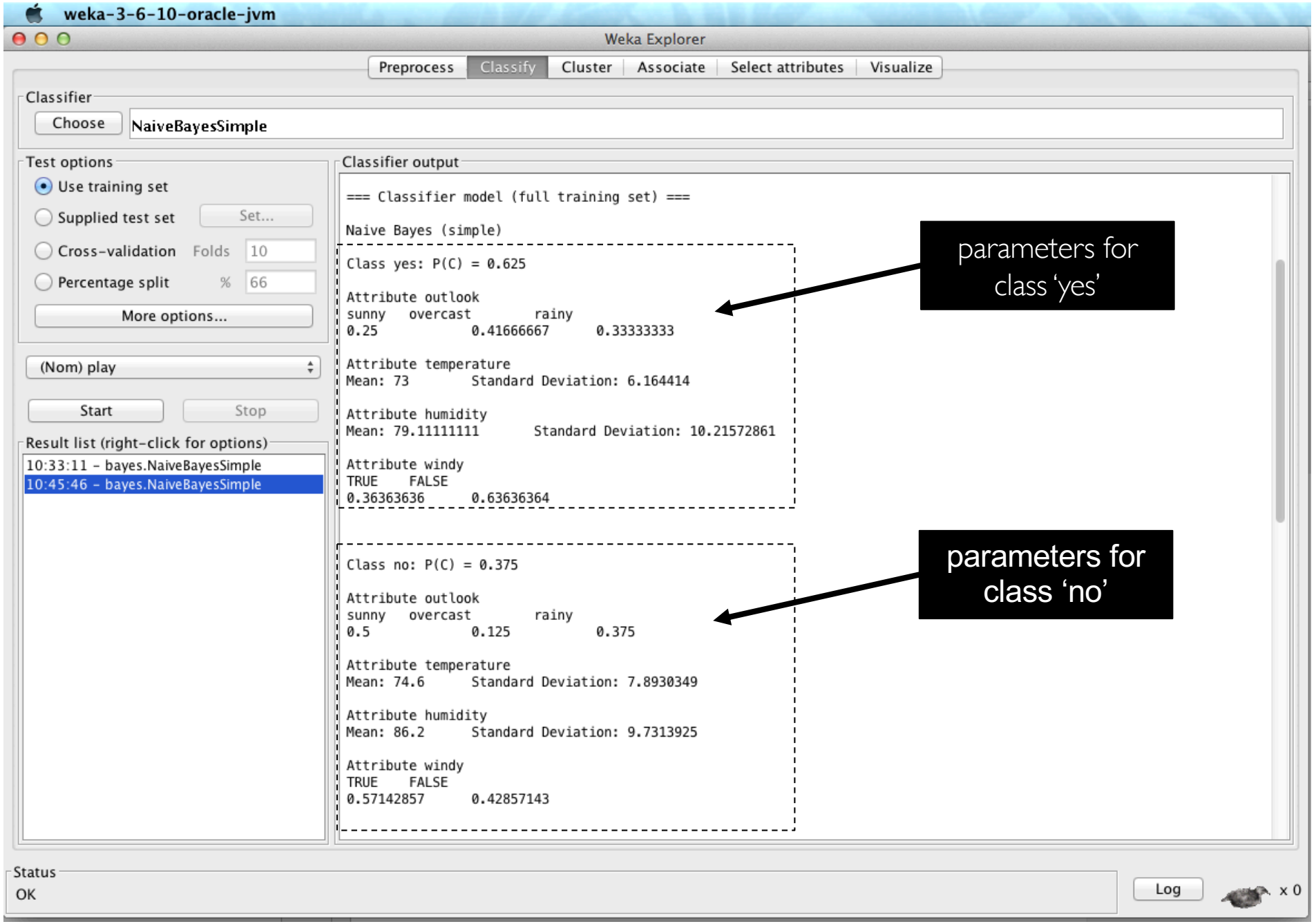
- Missing values during training are not included in calculation of mean and standard deviation

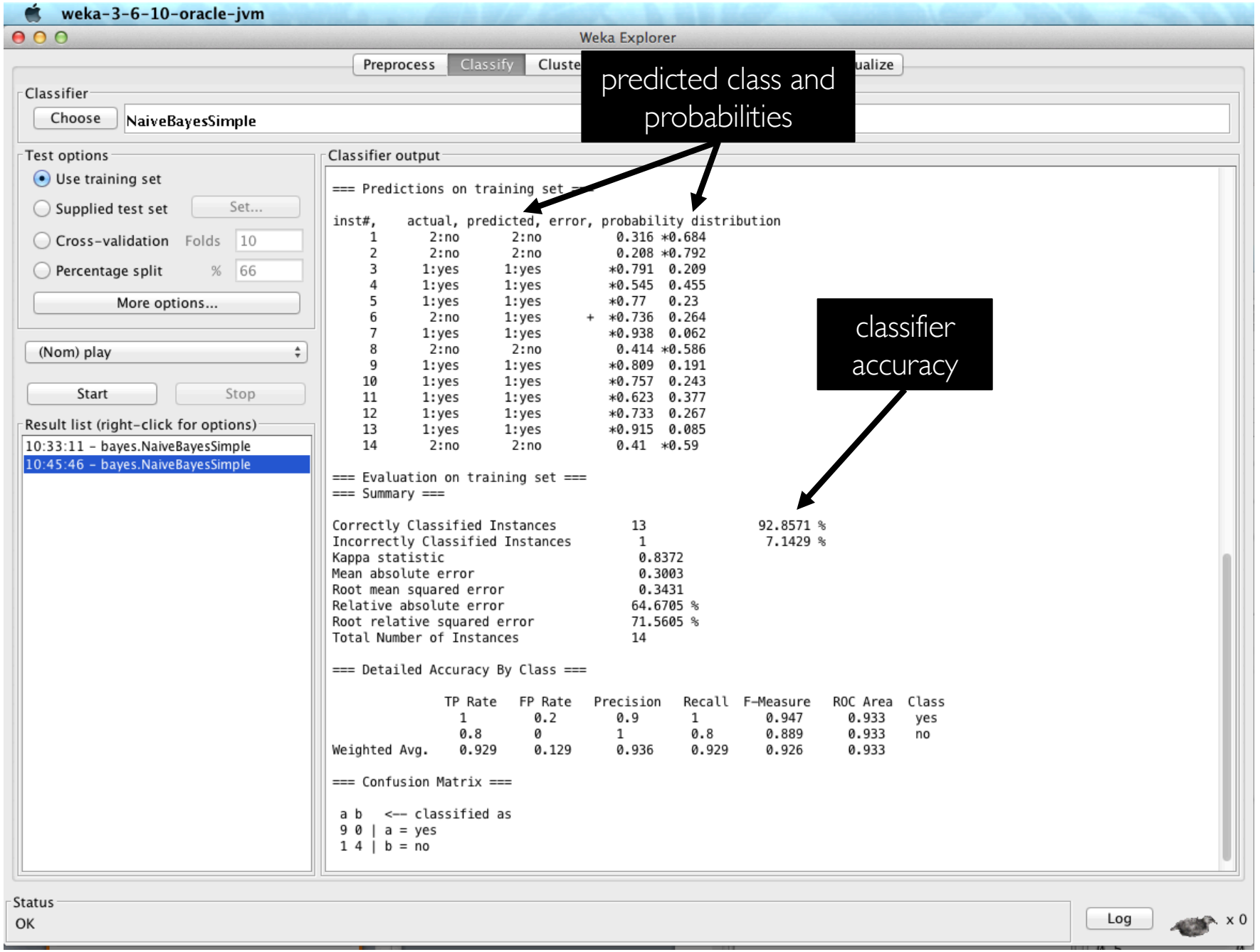
Likelihood of "yes" = 2/9 × 0.0340 × 0.0221 × 3/9 × 9/14 = 0.000036
Likelihood of "no" = 3/5 × 0.0291 × 0.0380 × 3/5 × 5/14 = 0.000136
P("yes" | Sunny, 66, 90, True) = 0.000036 / (0.000036 + 0. 000136) = 20.9%
P("no" | Sunny, 66, 90, True)) = 0.000136 / (0.000036 + 0. 000136) = 79.1%



Row ID	Attrib...	Value	Class	Count	Missin...	Mean	Stand...
Row0	humidity	?	no	5	0	86.2	9.731
Row1	humidity	?	yes	9	0	79.111	10.216
Row2	outlook	overcast	no	0	0	?	?
Row3	outlook	overcast	yes	4	0	?	?
Row4	outlook	rainy	no	2	0	?	?
Row5	outlook	rainy	yes	3	0	?	?
Row6	outlook	sunny	no	3	0	?	?
Row7	outlook	sunny	yes	2	0	?	?
Row8	play	?	no	5	0	?	?
Row9	play	?	yes	9	0	?	?
Row10	temperature	?	no	5	0	74.6	7.893
Row11	temperature	?	yes	9	0	73	6.164
Row12	windy	FALSE	no	2	0	?	?
Row13	windy	FALSE	yes	6	0	?	?
Row14	windy	TRUE	no	3	0	?	?
Row15	windy	TRUE	yes	3	0	?	?

Example of Naïve Bayes with nominal and continuous attributes using KNIME


weka-3-6-10-oracle-jvm
Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayesSimple
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) play
Start
Stop
Result list (right-click for options)
10:33:11 - bayes.NaiveBayesSimple
10:45:46 - bayes.NaiveBayesSimple
Classifier output
=== Classifier model (full training set) ===
Naive Bayes (simple)
Class yes: P(C) = 0.625
Attribute outlook
sunny overcast rainy
0.25 0.41666667 0.33333333
Attribute temperature
Mean: 73 Standard Deviation: 6.164414
Attribute humidity
Mean: 79.11111111 Standard Deviation: 10.21572861
Attribute windy
TRUE FALSE
0.36363636 0.63636364
Class no: P(C) = 0.375
Attribute outlook
sunny overcast rainy
0.5 0.125 0.375
Attribute temperature
Mean: 74.6 Standard Deviation: 7.8930349
Attribute humidity
Mean: 86.2 Standard Deviation: 9.7313925
Attribute windy
TRUE FALSE
0.57142857 0.42857143
parameters for class 'yes'
parameters for class 'no'
Status
OK
Log
x 0


weka-3-6-10-oracle-jvm
Weka Explorer
Preprocess
Classify
Cluste
ualize
predicted class and probabilities
Classifier
Choose
NaiveBayesSimple
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
Classifier output
=== Predictions on training set ===
inst#, actual, predicted, error, probability distribution
1 2:no 2:no 0.316 *0.684
2 2:no 2:no 0.208 *0.792
3 1:yes 1:yes *0.791 0.209
4 1:yes 1:yes *0.545 0.455
5 1:yes 1:yes *0.77 0.23
6 2:no 1:yes + *0.736 0.264
7 1:yes 1:yes *0.938 0.062
8 2:no 2:no 0.414 *0.586
9 1:yes 1:yes *0.809 0.191
10 1:yes 1:yes *0.757 0.243
11 1:yes 1:yes *0.623 0.377
12 1:yes 1:yes *0.733 0.267
13 1:yes 1:yes *0.915 0.085
14 2:no 2:no 0.41 *0.59
classifier accuracy
=== Evaluation on training set ===
=== Summary ===
Correctly Classified Instances 13 92.8571 %
Incorrectly Classified Instances 1 7.1429 %
Kappa statistic 0.8372
Mean absolute error 0.3003
Root mean squared error 0.3431
Relative absolute error 64.6705 %
Root relative squared error 71.5605 %
Total Number of Instances 14
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure ROC Area Class
1 0.2 0.9 1 0.947 0.933 yes
0.8 0 1 0.8 0.889 0.933 no
Weighted Avg. 0.929 0.129 0.936 0.929 0.926 0.933
=== Confusion Matrix ===
a b <-- classified as
9 0 | a = yes
1 4 | b = no
(Nom) play
Start
Stop
Result list (right-click for options)
10:33:11 - bayes.NaiveBayesSimple
10:45:46 - bayes.NaiveBayesSimple
Status
OK
Log
x 0

Discussion

- Naïve Bayes works surprisingly well, even if independence assumption is clearly violated
- Why? Because classification doesn't require accurate probability estimates as long as maximum probability is assigned to correct class
- However,
  - Adding too many redundant attributes will cause problems
    e.g. adding identical attributes will make classifier worse and worse
  - Also, many numeric attributes are not normally distributed so that Gaussian assumption may be poor and need substituting

Bayesian Belief Networks

Bayesian Belief Networks

- The conditional independence assumption,
  - makes computation possible
  - yields optimal classifiers when satisfied
  - but is seldom satisfied in practice, as attributes (variables) are often correlated
- Bayesian Belief Networks (BBN) allows us to specify which pair of attributes are conditionally independent
- They provide a graphical representation of probabilistic relationships among a set of random variables

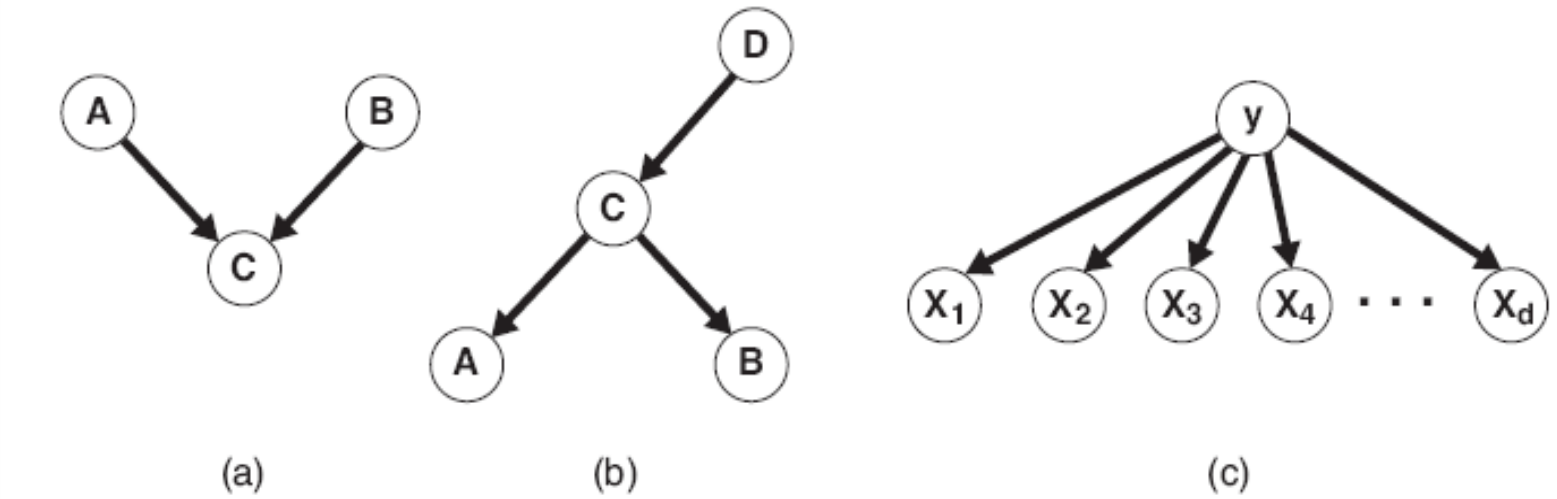
Bayesian Belief Networks

- Describe the probability distribution governing a set of variables by specifying
  - Conditional independence assumptions that apply on subsets of the variables
  - A set of conditional probabilities

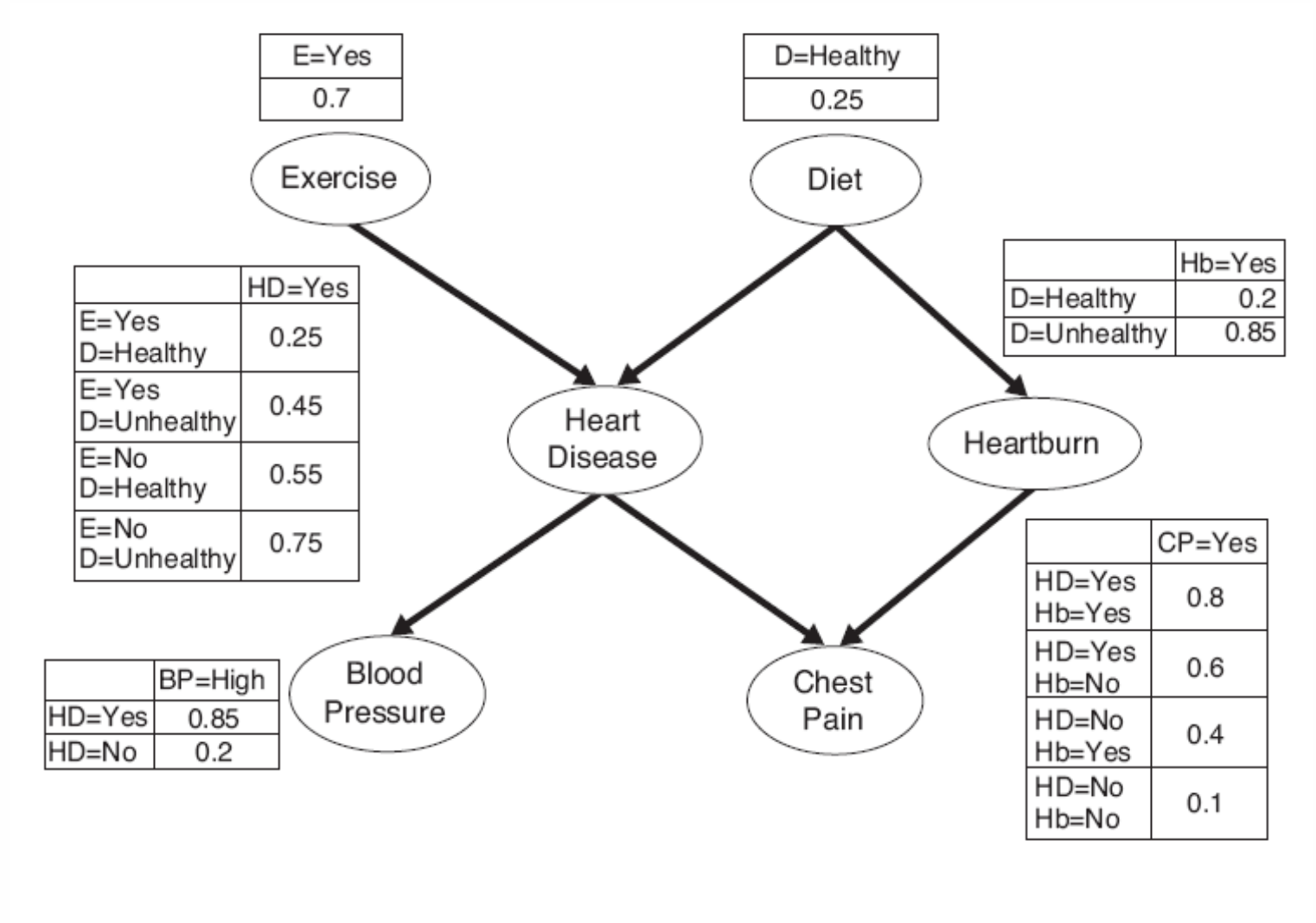
- Two key elements
  - A direct acyclic graph, encoding the dependence relationships among variables
  - A probability table associating each node to its immediate parents node

Examples of Bayesian Belief Networks

- Variable A and B are independent, but each one of them has an influence on the variable A
- A is conditionally independent of both B and D, given C
- The configuration of the typical naïve Bayes classifier



An Example of Bayesian Belief Network



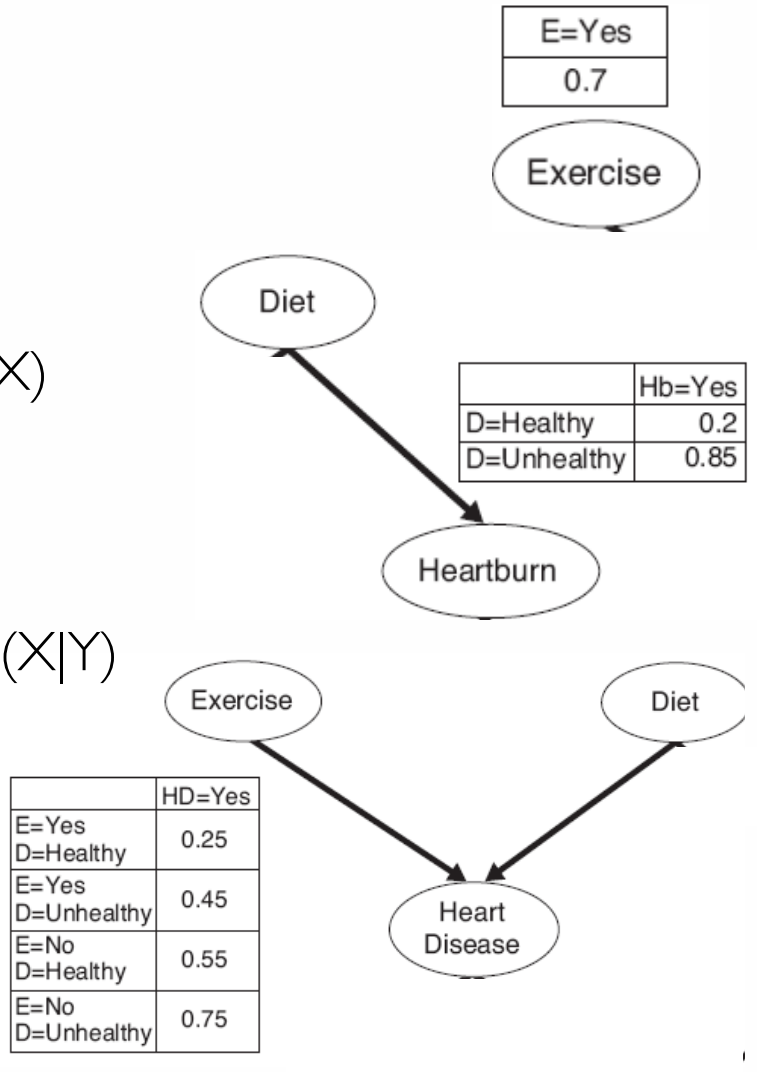
Probability Tables

- Each node is associated with probability table

- If a node X does not have any parents, then the table contains only the prior probability P(X)

- If a node X has only one any parent Y, then the table contains the conditional probability P(X|Y)

- If a node X has multiple parents (Y1, …, Yk) table contains the conditional probability P(X|Y1…Yk)

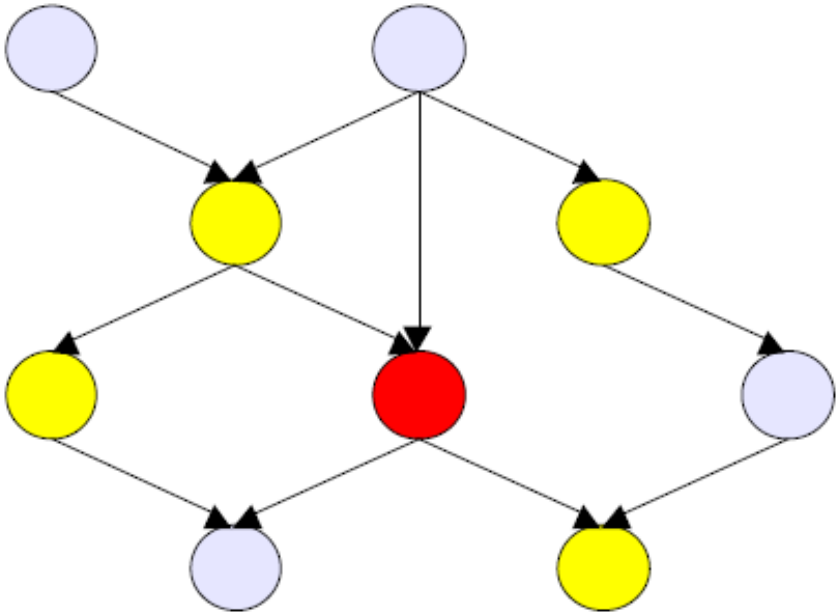


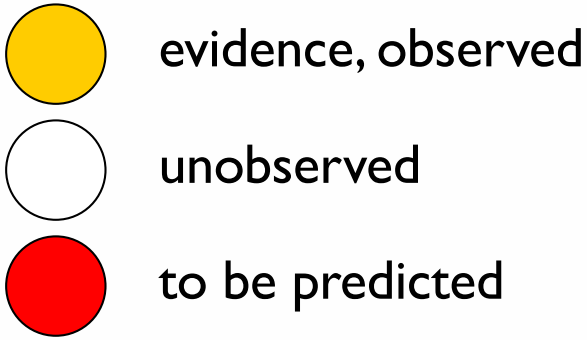
Probability Tables

- The network topology imposes conditions regarding the variable conditional independence
- Each node is associated with a probability table
  - If a node X does not have any parents, then the table contains only the prior probability P(X)
  - If a node X has only one any parent Y, then the table contains only the conditional probability P(X|Y)
  - If a node X has multiple parents, Y_1, …, Y_k the the table contains the conditional probability P(X|Y_1 … Y_k)

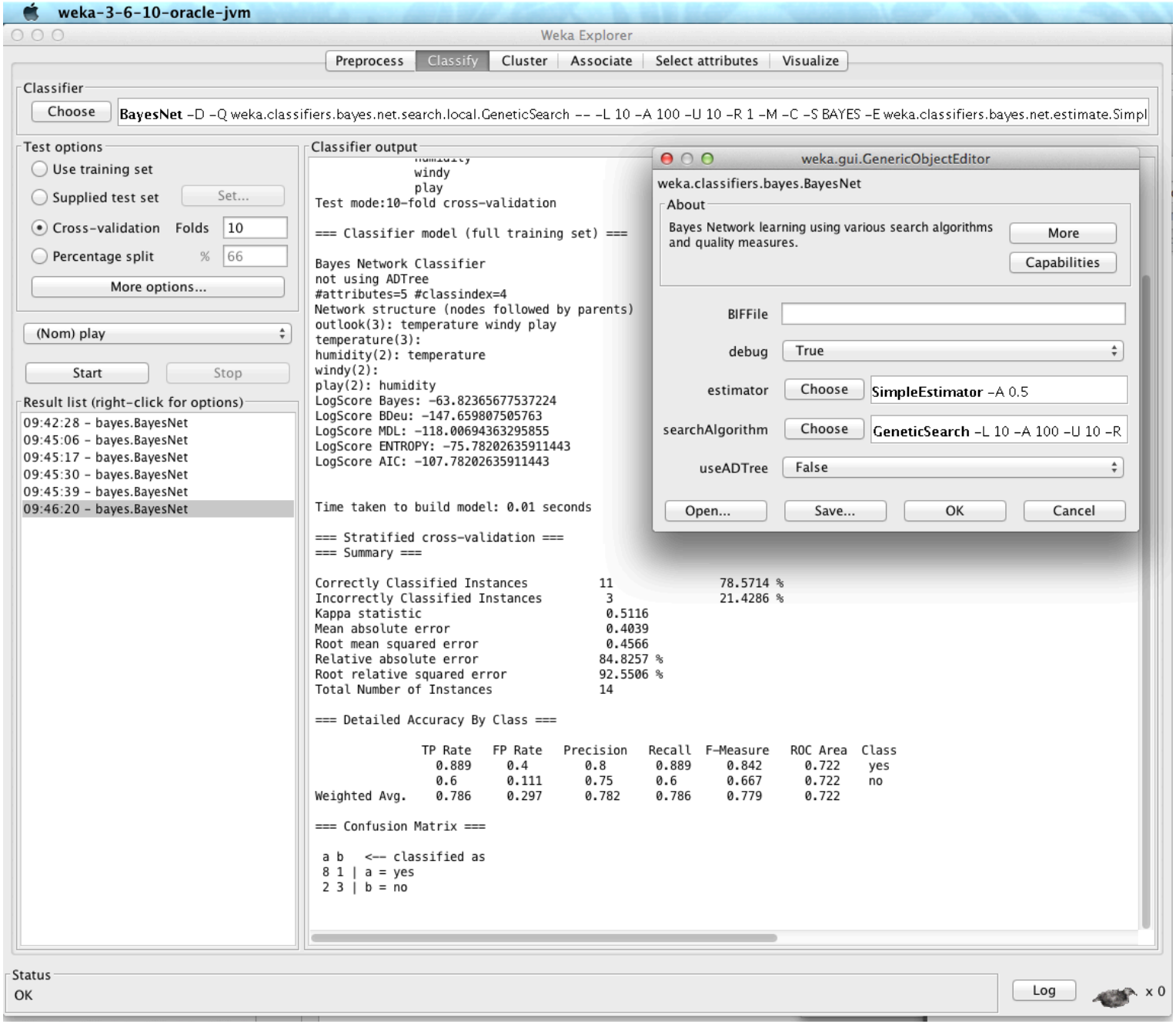
Bayesian Belief Networks

- Values of variables in Network can be known or unknown.
- Idea is to estimate probabilities over unknown variables given known values

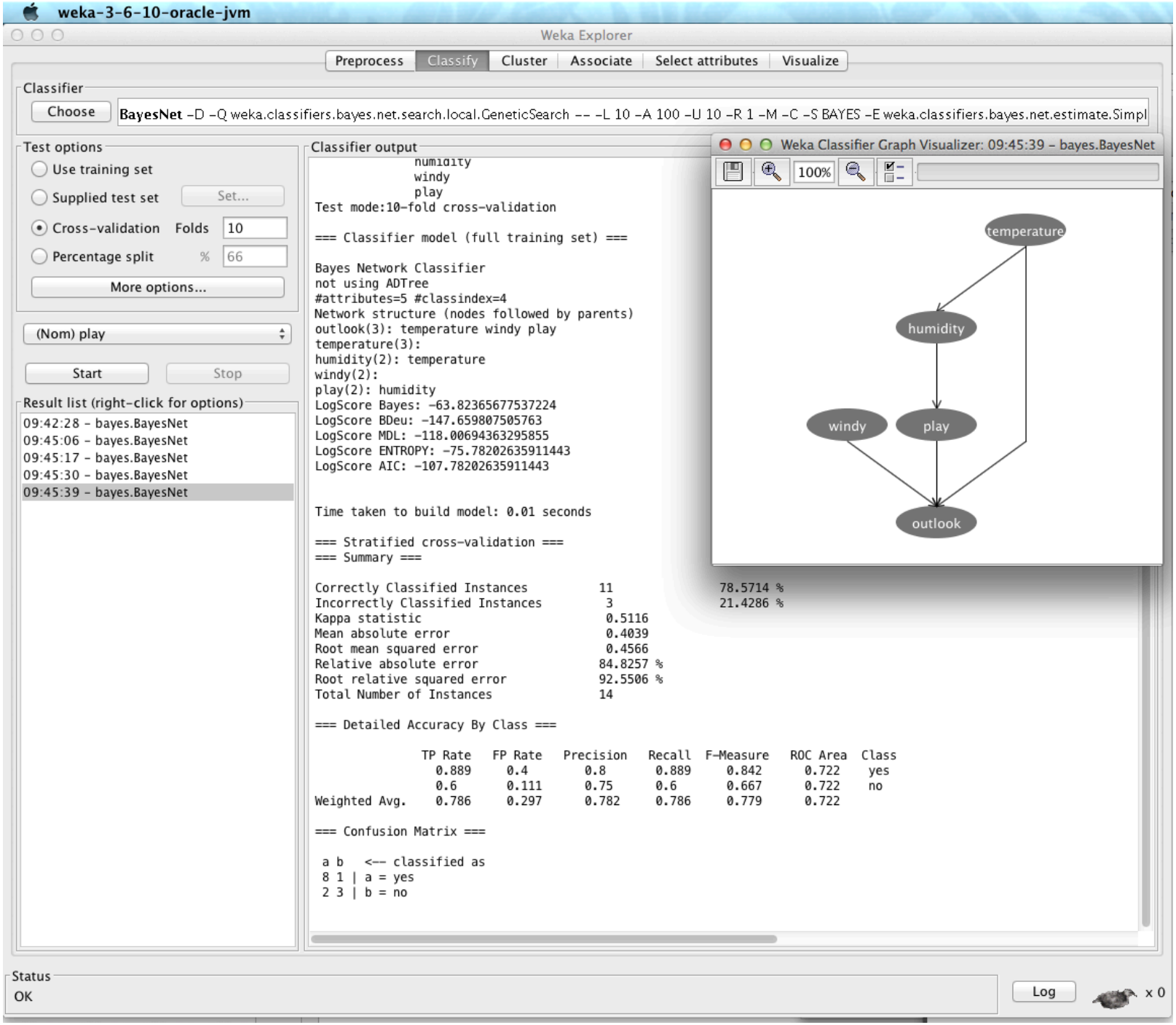




- In general the inference is NP-complete but there are approximating methods, e.g. Monte Carlo



http://www.cs.waikato.ac.nz/~remco/weka.bn.pdf


weka-3-6-10-oracle-jvm
Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
BayesNet -D -Q weka.classifiers.bayes.net.search.local.GeneticSearch -- -L 10 -A 100 -U 10 -R 1 -M -C -S BAYES -E weka.classifiers.bayes.net.estimate.Simpl
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) play
Start
Stop
Result list (right-click for options)
09:42:28 - bayes.BayesNet
09:45:06 - bayes.BayesNet
09:45:17 - bayes.BayesNet
09:45:30 - bayes.BayesNet
09:45:39 - bayes.BayesNet
Classifier output
humidity
windy
play
Test mode:10-fold cross-validation
=== Classifier model (full training set) ===
Bayes Network Classifier
not using ADTree
#attributes=5 #classindex=4
Network structure (nodes followed by parents)
outlook(3): temperature windy play
temperature(3):
humidity(2): temperature
windy(2):
play(2): humidity
LogScore Bayes: -63.82365677537224
LogScore BDeu: -147.659807505763
LogScore MDL: -118.00694363295855
LogScore ENTROPY: -75.78202635911443
LogScore AIC: -107.78202635911443
Time taken to build model: 0.01 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 11 78.5714 %
Incorrectly Classified Instances 3 21.4286 %
Kappa statistic 0.5116
Mean absolute error 0.4039
Root mean squared error 0.4566
Relative absolute error 84.8257 %
Root relative squared error 92.5506 %
Total Number of Instances 14
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure ROC Area Class
0.889 0.4 0.8 0.889 0.842 0.722 yes
0.6 0.111 0.75 0.6 0.667 0.722 no
Weighted Avg. 0.786 0.297 0.782 0.786 0.779 0.722
=== Confusion Matrix ===
a b <-- classified as
8 1 | a = yes
2 3 | b = no
Weka Classifier Graph Visualizer: 09:45:39 - bayes.BayesNet
100%
temperature
humidity
windy
play
outlook
Status
OK
Log
x 0

Learning in Bayesian Belief Networks

- Several cases of learning Bayesian belief networks
  - Given both network structure and all the variables is easy
  - Given network structure but only some variables
  - When the network structure is not known in advance

Bayesian Belief Networks

- Bayesian Networks
  - Encode causal dependencies among variables
  - Well suited to dealing with incomplete data
  - Robust to overfitting
  - When used to build classifier, they extend Naïve Bayes by removing conditional independence assumption
- Other benefits
  - Can be used to capturing prior knowledge from domain experts
  - Graphical model describing relationship between variables is often useful for understanding / visualizing data
- Building the network is time consuming!
  - Involves searching huge space of possible conditional independence relationships
  - But adding variables to an existing network is straightforward