jBNC: Quick Start

Jarek Sacha

January 4, 2003

1 Introduction

Bayesian Network Classifier Toolbox – jBNC is a library of Java classes for training and testing Bayesian network classifiers. Current version of jBNC also contains command line tools for training and testing classifiers implemented in the toolkit.

Classifiers implemented by jBNC have been shown to perform well in a variety of artificial intelligence, machine learning, and data mining applications. Detailed description of the implemented classifiers and their benchmark tests can be found in [9].

More information and the most recent version of jBNC can be found at `http://jbnc.sourceforge.net`. This document was created for jBNC v.1.0.

Bayesian network classifiers implemented by jBNC

- Naive Bayes [4]
- TAN - tree augmented naive Bayes [6]
- FAN - forest augmented naive Bayes
- STAN - selective tree augmented naive Bayes
- STAND - selective tree augmented naive Bayes with node discarding
- SFAN - selective forest augmented naive Bayes
- STAND - selective forest augmented naive Bayes with node discarding

Network quality measures implemented by jBNC

- HGC - Heckerman-Geiger-Chickering
- SB - Standard Bayesian
- LC - Local criterion
- LOO - Leave-One-Out cross validation
- $CV_{n,t}$ - n -fold t-times Cross Validation

2 Setup

1. Download the jBNC toolkit distribution from `http://jbnc.sorceforge.net`
2. Unpack content of `jBNC_full_v.1.0.zip` into a directory of your choice, we will call it *`jbnc_home`*. Make sure when unpacking that directory structure in the archive is preserved.
3. Add `jBNC_all.jar` to your system `CLASSPATH` variable. Alternatively you can setup `CLASSPATH` variable just before you run command line tools, for instance using command:

   ```
   set CLASSPATH=jbnc_home/jBNC.jar
   ```

3 Command Line Tools

jBNC has three utilities that can be accessed from a command line: `Classifier`, `CrossVal`, and `DatasetInfo`. General syntax for running command line utilities is following:

```
java jbnc.command options
```

Typing command without options, for instance:

```
java jbnc.Classifier
```

prints options summary.

3.1 `Classifier` and `CrossVal` Utilities

`Classifier` utility is used to learn and test a Bayesian network classifier, it uses a single learning dataset and a single testing dataset. `CrossVal` utility performs learning and cross-validation test of a Bayesian network classifier, it assumes that cross validation datasets are already generated and attempts to read them using given file stem. The file name format is *filestem-repetition-fold*. For instance for file stem `vote` the file names could be `vote-0-0.*`, `vote-0-1.*`, etc. Examples of cross-validation datasets can be found in subdirectory `data/bench_cv5`. The example datasets were generated using the `GenCVFiles` utility from MLC++ library [7].

Both, `Classifier` and `CrossVal` utilities accept the same command line options:

-a *algor_name* Bayesian network classifier algorithm choices: `naive` (for naive Bayes), `TAN`, `FAN`, `STAN`, `STAND`, `SFAN`, `SFAND`.

-c *class_name* Name of the class variable. The default value is `class`.

-d Print debugging information.

-f *filestem* Load test data in C4.5 format (`.names` + `.data` + `.test`). For `Classifier` the files will be: *filestem*`.names` – file with specification of attributes, *filestem*`.data` – file with training cases, and *filestem*`.test` – file with test cases. For `CrossVal` the file names will be *filestem*`-?-?.names`, *filestem*`-?-?.data`, and *filestem*`-?-?.test`.

-l *table-name* Log result to database table *table-name*. It is assumed that results are logged to a database with ODBC name `jtest`.

-n *filename* Save constructed network(s) to *filename*`.bif`. File is saved in BIF 0.15 format.

-q *quality-measure* Bayesian network quality measure choices: HGC – Heckerman-Geiger-Chickering, SB –Standard Bayesian, LC – Local criterion, LOO – leave-one-out cross validation, CV10 – ten-fold cross validation, CV1010 – ten-fold ten-times cross validation.

-s *number* Number of smoothing priors to test; has to be an integer greater or equal to zero.

-t Print execution time in milliseconds.

For example, to test FAN classifier using LOO quality measure on dataset `vote` use the following command:

```
java jbnc.Classifier -a FAN -q LOO -f data/bench/vote
```

The output should look like something like this:

```
Classifier tester
File stem:  data/orig/vote
Algorithm:  FAN
Quality measure:  Leave-one-out cross validation
+
Error = 5.926% +- 2.04% (94.074%) [127/8/135]
```

Number after `-+` is an estimate of the standard deviation according to binomial model. The number in parenthesis is accuracy (100%-error), the numbers in square brackets are: number of correct classifications, number of false classifications, and total number of cases in test set, respectively.

3.2 `DatasetInfo` Utility

The `DatasetInfo` utility is used to print information about a dataset. It takes a single attribute: the file name stem. It assumes that the file is saved in C4.5 format. `DatasetInfo` prints information about number of classes and number of attributes in the dataset (defined in file *filestem*`.names`). It also prints frequency of each of the classes in train and test datasets. For instance, command

```
java jbnc.DatasetInfo data/orig/vote
```

would produce output

```
DatasetInfo
Filestem:  db/orig/vote
Number of classes = 2
Number of attributes = 16
File '../db/vote.all' has 435 cases.
  democrat :  267 [61.38%]
  republican :  168 [38.62%]
File '../db/vote.data' has 300 cases.
  democrat :  184 [61.33%]
  republican :  116 [38.67%]
File '../db/vote.test' has 135 cases.
  democrat :  83 [61.48%]
  republican :  52 [38.52%]
```

4 jBNC API

jBNC can be readily used as a library. JavaDoc API documentation is included with the distribution, it is also available on-line at `http://jbnc.sourceforge.net/docs/api/index.html`. Source code for command line utilities can be used as coding examples.

5 Benchmark Datasets

The jBNC distribution contains directory `data` with datasets used for benchmarking of classifier implemented by jBNC [9]. The benchmark datasets are in the C4.5 format [8]. Most of the datasets are from the University of California at Irvine Machine Learning Repository `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Directory `data/orig` contains datasets in the original format. Data sets with continuous attributes were discretized using the minimal entropy heuristic [3, 5]. Benchmarking for most datasets was based on 5-fold cross-validations. Those dataset are marked by CV-5 in Table 1 and are located in `data/bench_cv5`. Datasets that were discretized but used without cross-validation are in `data/bench_disc`.

Generation of cross validation partition, used during experiments, and discretization of continuous attributes (the minimal entropy heuristic) had been performed using MLC++ package [7]. Note, each of the training partitions was discretized separately, then the discovered scheme was used to discretize a corresponding test partition.

5.1 The Datasets

Here is a brief description of the datasets used for benchmarking; more information can be found in [1].

australian Australian credit approval data.

breast Breast cancer databases from the University of Wisconsin Hospitals, Madison.

chess Chess End-Game – King+Rook versus King+Pawn on a7 (usually abbreviated KRKPA7). The pawn on a7 means it is one square away from queening. It is the King+Rook's side (white) to move.

cleve Cleveland heart disease database. Eight attributes are symbolic, six numeric. There are two classes: healthy (buff) or with heart-disease (sick). The attributes are: age, sex, chest pain type (angina, abnang, notang, asympt), resting blood pressure, serum cholesterol in mg/dl, fasting blood sugar $<$ 120 mg/dl (true or false), resting ECG (norm, abnormal, hyper), max heart rate, exercise induced angina (true or false), oldpeak = ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment (up, flat, down), number of major vessels (0-3) colored by fluoroscopy, thallium (normal, fixed defect, reversible defect).

corral An artificial dataset designed to show that decision trees might pick a really bad attribute for the root. The target concept is $(a_1$ **xor** $a_2)$ **or** $(a_3$ **xor** $a_4)$, attribute A_5 is correlated to the class variable and attribute A_6 is irrelevant.

crx Credit card applications data.

diabetes Pima Indians diabetes database.

flare Classification of solar flares.

german German credit database.

glass Glass identification database.

glass2 Variant of the glass identification database with two classes and corresponding cases removed.

heart Another heart disease database. It has the structure similar to `cleve` database, the same classes and attributes.

hepatitis Survival of hepatitis patients.

iris This is perhaps the best known database to be found in the pattern recognition literature. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other.

letter The objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli.

lymphography Classification of lymphography data.

mofn-3-7-10 Artificial dataset: 10 bits; 3 out of 7 should be on; remaining three are irrelevant (A_1, A_2,A_{10}).

pima Pima Indians diabetes database from the National Institute of Diabetes and Digestive and Kidney Diseases.

satimage Landsat Satellite data: multi-spectral values of pixels in 3x3 neighborhoods in a satellite image, and the classification associated with the central pixel in each neighborhood.

segment Image segmentation database. The instances were drawn randomly from a database of 7 outdoor images. The images were hand-segmented to create a classification for every pixel.

shuttle-small The shuttle dataset contains 9 attributes all of which are numerical. Approximately 80% of the data belongs to class 1.

soybean-large Soybean disease databases.

vehicle Vehicle silhouettes: 3D objects within a 2D image by application of an ensemble of shape feature extractors to the 2D silhouettes of the objects.

vote Voting records drawn from the Congressional Quarterly Almanac, 98th Congress, 2nd session 1984, Volume XL: Congressional Quarterly Inc. Washington, D.C., 1985.

waveform-21 Artificial dataset from waveform generator. Three classes of waveforms. Each class is generated from a combination of 2 or 3 “base” waves. All 21 attributes include noise.

References

[1] C. Blake, E. Keogh, and C. Merz. UCI repository of machine learning databases, 1998. http://www.ics.uci.edu/~mlearn/MLRepository.html.

[2] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth & Brooks, Monterey, CA, 1984.

[3] J. Catlett. On changing continuous attributes into ordered discrete attributes. In Y. Kodratoff, editor, *Proceedings of the European Working Session on Learning*, pages 164–178, Berlin, Germany, 1991. Springer-Verlag.

[4] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973.

[5] U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 1022–1027. Morgan Kaufmann, 1993.

[6] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29 (2/3):131–163, November 1997.

[7] R. Kohavi, D. Sommerfield, and J. Dougherty. Data mining using MLC++, a machine learning library in C++. *International Journal of Artificial Intelligence Tools*, 6(4):537–566, 1997.

[8] J. R. Quinlan. *C4.5: Programming for Machine Learning*. Morgan Kaufmann, 1993.

[9] J. P. Sacha. *New Synthesis of Bayesian Network Classifiers and Cardiac SPECT Image Interpretation*. Ph.D. Dissertation, The University of Toledo, Dec. 1999.

Table 1: UCI Machine Learning Repository datasets used for testing.

Dataset	# Attributes	# Classes	# Cases	
			Train	Test
australian	14	2	690	CV-5
breast	10	2	683	CV-5
chess	36	2	2,130	1,066
cleve	13	2	296	CV-5
corral	6	2	128	CV-5
crx	15	2	653	CV-5
diabetes	8	2	768	CV-5
flare	10	2	1,066	CV-5
german	20	2	1,000	CV-5
glass	9	7	214	CV-5
glass2	9	2	163	CV-5
heart	13	2	270	CV-5
hepatitis	19	2	80	CV-5
iris	4	3	150	CV-5
letter	16	26	15,000	5,000
lymphography	18	4	148	CV-5
mofn-3-7-10	10	2	300	1,024
pima	8	2	768	CV-5
satimage	36	6	4,435	2,000
segment	19	7	1,540	770
shuttle-small	9	7	3,866	1,934
soybean-large	35	19	562	CV-5
vehicle	18	4	846	CV-5
vote	16	2	435	CV-5
waveform-21	21	3	300	4,700