Anytime learning and classification for online applications

Geoff Webb

Zijian Zheng

Janice Boughton

Zhihai Wang

Ying Yang

Monash University,

Melbourne, Australia

http://www.csse.monash.edu.au/∼webb

MONASH University

Overview

- Many online applications require fast effective classification
  - user modeling
  - online assistants
  - recommender systems
  - spam & fraud detection
- Common solution uses
  - most efficient algorithm that delivers acceptable accuracy, eg naive Bayes, and
  - sufficient computational resources to deliver acceptable performance under peak loads
- Implies computational resources are idle in off-peak periods
- Current research investigates using any available idle resources to improve naive Bayes

Classification learning

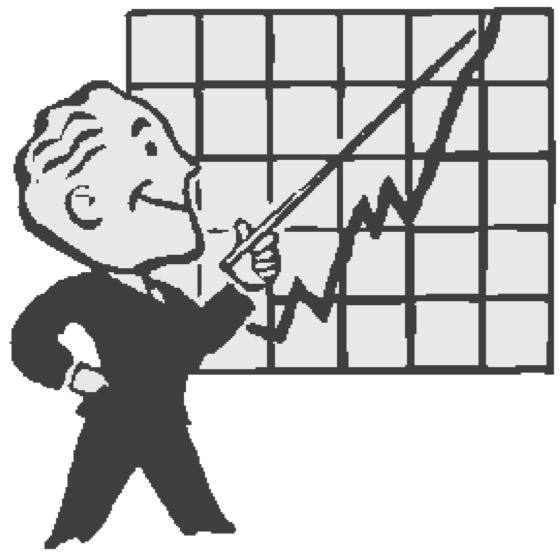
- Given a sample from XY want to select $y \in Y$ for new $\mathbf{x} = \langle x_1, \ldots, x_n \rangle \in X$
  - eg Xs = symptoms, Ys = diseases
- Error minimized by $argmax_y(P(y \mid \langle x_1, \ldots, x_n \rangle))$
  - but do not know probabilities
- Can estimate using
  - $P(W) \approx F(W)$
  - $P(W \mid Z) \approx \frac{F(W,Z)}{F(Z)}$
  - but usually too little data for accurate estimation for $P(\langle x_1, \ldots, x_n \rangle)$ or $P(y \mid \langle x_1, \ldots, x_n \rangle)$

Bayes' theorem

- $P(y \,|\, \mathbf{x}) = \frac{P(y)P(\mathbf{x} \,|\, y)}{P(\mathbf{x})}$
- $P(y \,|\, \mathbf{x}) \propto P(y)P(\mathbf{x} \,|\, y)$
- can estimate $P(y)$ from data so have replaced estimating $P(y \,|\, \mathbf{x})$ with estimating $P(\mathbf{x} \,|\, y)$
- Attribute independence assumption
  - $P(\langle x_1, \ldots, x_n \rangle \,|\, y) = \prod_{i=1}^{n} P(x_i \,|\, y)$
  - eg
    $$P(temp{=}high, pulse{=}high \,|\, ill) = P(temp{=}high \,|\, ill) \times P(pulse{=}high \,|\, ill)$$

Naive Bayesian Classification

- use Bayes theorem, attribute independence assumption, and estimation of probabilities from data to select most probable class for given x
- simple, efficient, and accurate
- direct theoretical foundation
- can provide probability estimates
- not necessarily Bayesian!



Attribute independence assumption

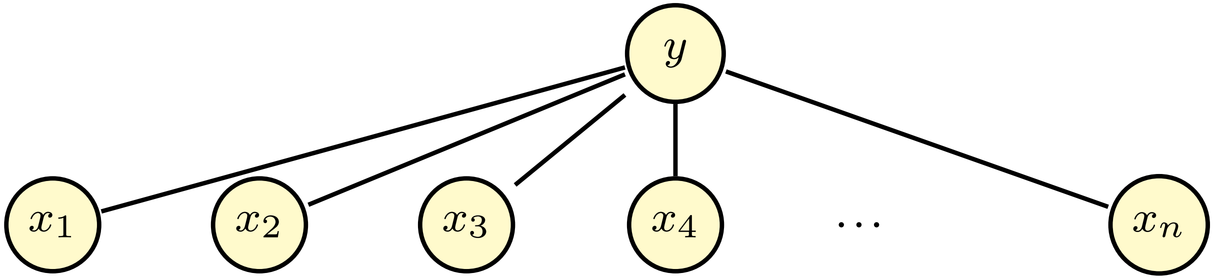
- Violations of the attribute independence assumption can increase expected error.
- Some violations do not matter (Domingos & Pazzani, 1996).
- Violations that matter are frequent
  - NB is often sub-optimal

Semi-naive Bayesian classification

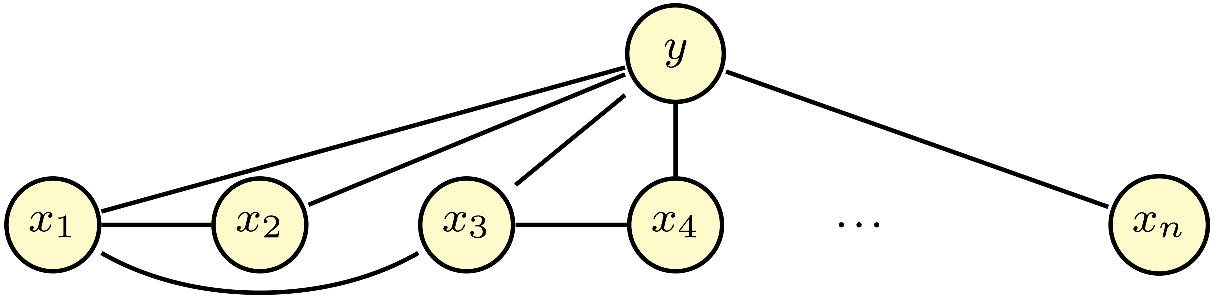
- Kononenko (1991) joins attributes
- Recursive Bayesian classifier (Langley, 1993)
- Selective naive Bayes (Langley & Sage, 1994)
- BSEJ (Pazzani, 1996)
- NBTree (Kohavi, 1996)
- Limited dependence Bayesian classifiers (Sahami, 1996)
- TAN (Friedman, Geiger & Goldszmidt, 1997)
- Adjusted probability NB (Webb & Pazzani, 1998)
- LBR [Lazy Bayesian Rules] (Zheng & Webb, 2000)
- Belief Net Classifiers (Greiner, Su, Shen & Zhou, 2005)
- PDAGs (Acid, de Campos & Castellano, 2005)
- TBMATAN (Cerquides & de Mantaras, 2005)

Markov net perspective

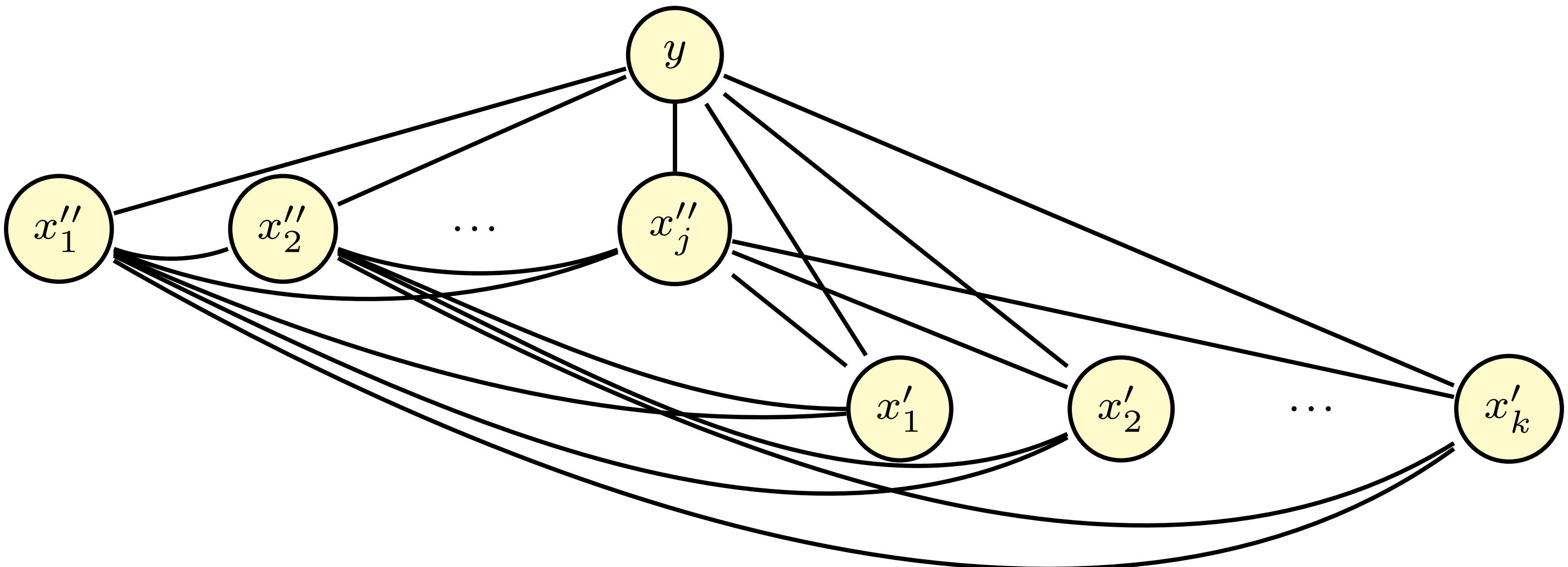
NB:



TAN:



LBR:



An ensemble approach

- Objective
  - Maintain accuracy of LBR and TAN while lowering computation
- Computation results from
  - calculation of conditional probabilities
  - selection of interdependencies
- If allow at most class + k attribute interdependencies per attribute, probabilities can be estimated from an $k+2$ dimensional lookup table of joint frequencies
  - $P(x_i \,|\, y, x_j) \approx F[x_i, y, x_j]/F[x_j, y, x_j]$

AODE

- For efficiency, use 3d table, each attribute depends on class and one other attribute
  - in theory can accommodate any pair-wise attribute interdependencies
- For efficiency and to minimize variance, avoid model selection
  - use all interdependencies for which there is sufficient data for probability estimation
- Conflict: cannot represent multiple interdependencies if only one interdependency per attribute
- Solution: average all models that have a single attribute as parent to all others
- Qualification: restrict parents to frequent attribute values

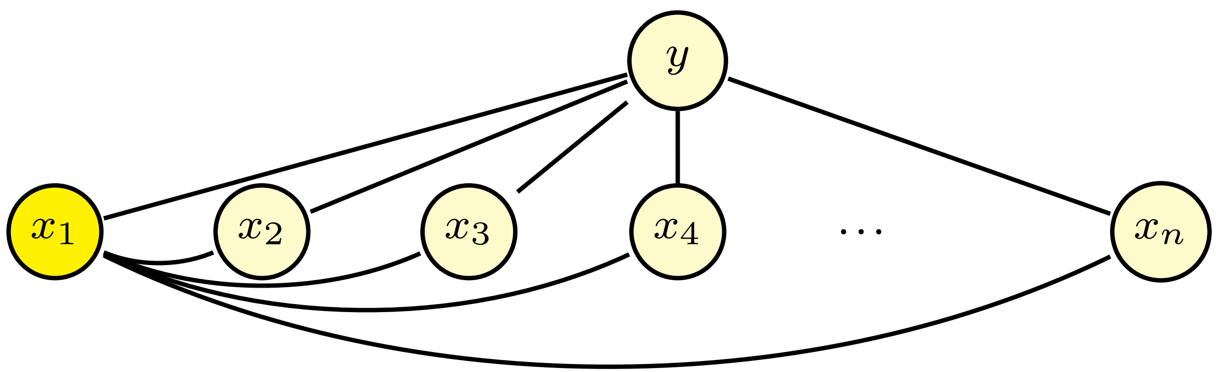
AODE (cont.)

$$P(y \mid \langle x_1, \ldots, x_n \rangle) = \frac{P(y, \langle x_1, \ldots, x_n \rangle)}{P(\langle x_1, \ldots, x_n \rangle)}$$

$$\begin{aligned} P(y, \langle x_1, \ldots, x_n \rangle) &= P(y, x_i) P(\langle x_1, \ldots, x_n \rangle \mid y, x_i) \\ &= \frac{\sum_{i:|x_i|>k} P(y, x_i) P(\langle x_1, \ldots, x_n \rangle \mid y, x_i)}{|\{i : |x_i| > k\}|} \end{aligned}$$

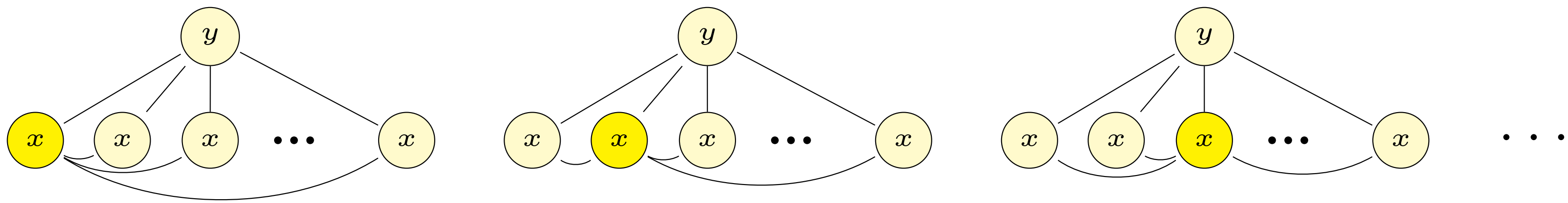
$$P(\langle x_1, \ldots, x_n \rangle \mid y, x_i) \approx \prod_{j=1}^{n} P(x_j | y, x_i)$$

Markov net:



AODE interpretations

- Bayesian average over all dual parent models
  - uniform prior
- Ensemble of all dual parent models



Complexity

alg.	train time	train space	class time	class space
NB	$O(ni)$	$O(nvc)$	$O(nc)$	$O(nvc)$
AODE	$O(n^2i)$	$O((nv)^2c)$	$O(n^2c)$	$O((nv)^2c)$
TAN	$O(n^3ci)$	$O((nv)^2c + ni)$	$O(nc)$	$O(nv^2c)$
LBR	$O(ni)$	$O(ni)$	$O(n^3ci)$	$O(ni + nvc)$

n = no. of attributes
v = ave. no. attribute values
c = no. classes
i = no. training instances

Evaluation

- 37 data sets from UCI repository
  - data used in previous related research
  - minus pioneer for which we could not complete computation
- Algorithms implemented in Weka
- NB, AODE, TAN, LBR, J48, boosted J48
- MDL discretisation for NB, AODE, TAN and LBR
- Laplace estimate
- 10-fold cross-validation

Error

Mean error:

AODE	NB	TAN	LBR	J48	Boosted J48
0.209	0.223	0.214	0.212	0.229	0.206

Geometric mean error ratio:

NB	TAN	LBR	J48	Boosted J48
1.104	1.038	1.030	1.187	1.006

Win–draw–loss table with 2-tail *p*:

NB	TAN	LBR	J48	Boosted J48
21-6-10	22-2-13	18-3-16	23-0-14	20-0-17
0.0354	0.0877	0.4321	0.0939	0.3714

Compute time

- Mean training time in seconds

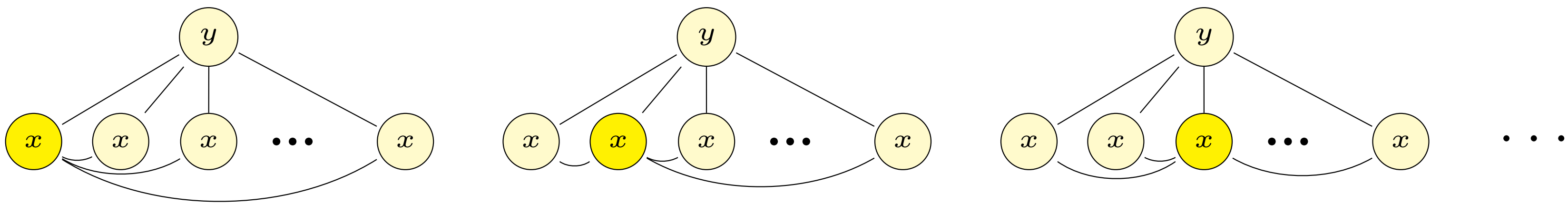
AODE	NB	TAN	LBR	J48	Boosted J48
3.8	3.4	516.9	4.2	26.6	390.4

- Mean testing time in seconds

AODE	NB	TAN	LBR	J48	Boosted J48
1.1	0.2	0.1	15456.1	0.1	0.6

Further features

- Incremental
- Parallelizable
- Anytime classification



Anytime classification

- Assume computational budget
  - separate training and classification time budgets
  - both time and space
  - both contract and anytime components
- Need small improvement steps
- Want monotonicity
- Want performance at least as good as naive Bayes when learning terminated after equivalent computation

Anytime AODE

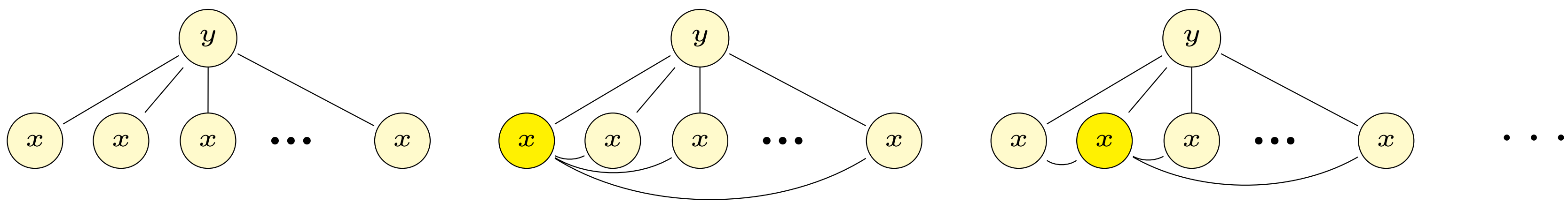
- Compute as many SPODEs as time allows
- Return average of all SPODEs computed

Start with naive Bayes!

- Undesirable to start with a single SPODE, as high variance often leads to lower accuracy than naive Bayes
- Solution, use naive Bayes then a sequence of SPODEs

$$P(y \mid \langle x_1, \ldots, x_n \rangle) = \frac{P(y, \langle x_1, \ldots, x_n \rangle)}{P(\langle x_1, \ldots, x_n \rangle)}$$

$$\begin{aligned}
P(y, \langle x_1, \ldots, x_n \rangle) &= P(y)P(\langle x_1, \ldots, x_n \rangle \mid y) \\
&= P(y, x_i)P(\langle x_1, \ldots, x_n \rangle \mid y, x_i) \\
&= \frac{P(y)P(\langle x_1, \ldots, x_n \rangle \mid y) + \sum_{i=1}^{n} P(y, x_i)P(\langle x_1, \ldots, x_n \rangle \mid y, x_i)}{n+1}
\end{aligned}$$



MONASH University

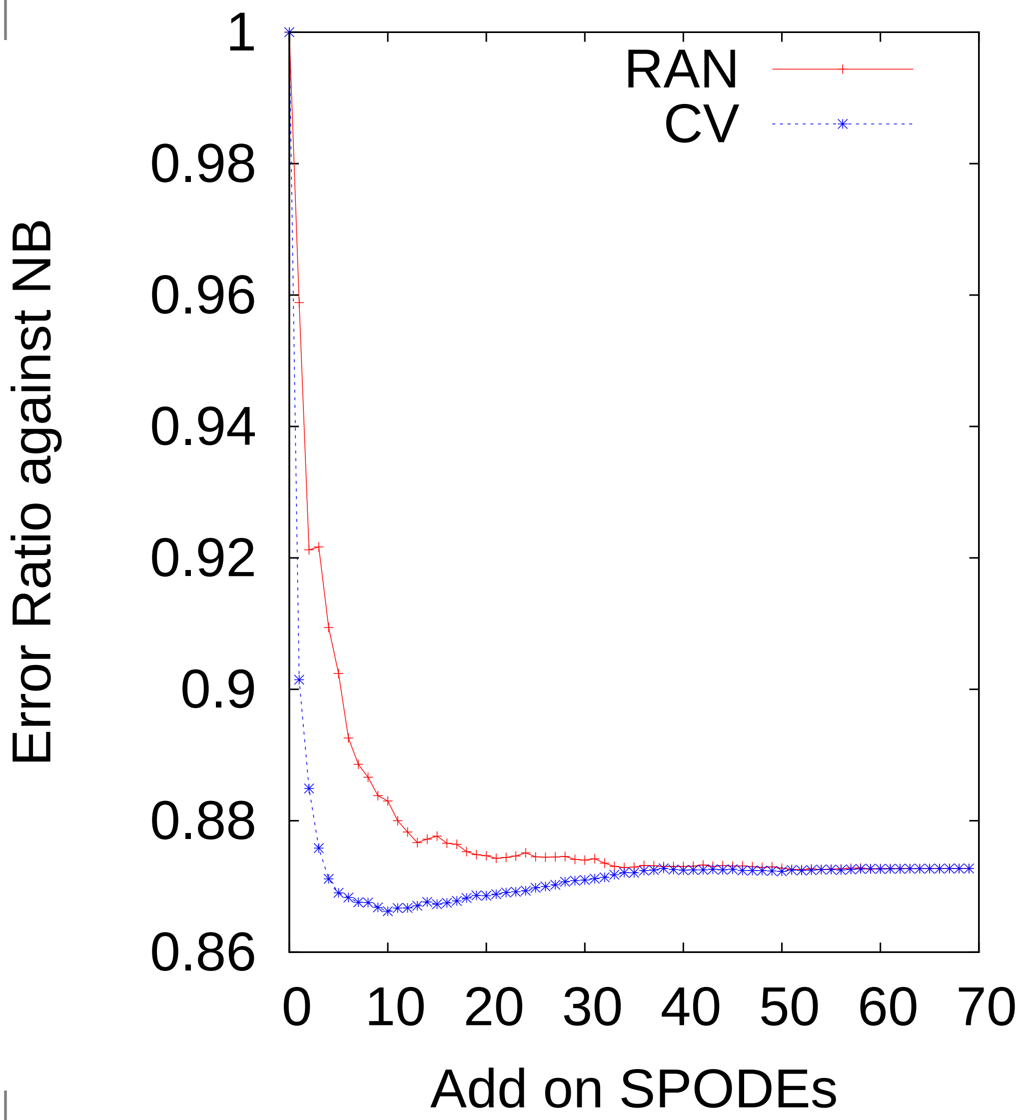
Ordering

- It is credible that some SPODEs will be more effective than others
- It would be desirable to include them first
- CV evaluates each SPODE on the training data
  - leave-one-out cross-validation
  - order from most to least effective

Experients

- Datasets
  - abalone, adult, ae, anneal, audio, autos, balance-scale, bands, breast-cancer-wisconsin, bupa, chess, cleveland-heart-disease, cmc, cedit-assessment, dmplexer, echocardiogram, german, glass, heart, hepatitis, horse-colic, house, hungarian, hypo, ionosphere, iris, kr-vs-kp, labor-neg, led, letter, lung-cancer, lymphography, mfeat-mor, mush, new-thyroid, optdigits, page-blocks, pendigits, phoneme, pid, post-operative, promoters, primary-tumor, satellite, soybean-large, segment, sick, sign, sonar, splice-junction, syncon, thyroid, tic-tac-toe, vehicle, volcanoes, vowel-context, waveform-5000, wine, yeast, zoo

Comparison of CV and Random order



- Mean across all datasets of results standardised against NB
- Random is monotonic
- CV selects better SPODEs first
- Need stopping criterion!

Conclusions

- During off-peak periods many online classification systems will fail to fully utilise available computational resources
- Popular naive Bayes can be augmented by ensemble of SPODEs
- Utilise otherwise idle computational resources to improve classification accuracy
- Supports incremental and parallel as well as anytime classification