Fast and Effective Single Pass Bayesian Learning

Nayyar A. Zaidi, Geoffrey I. Webb

Faculty of Information Technology, Monash University, Melbourne VIC 3800, Australia

15 April 2013

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.
- In addition, a desirable classifier should have:

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.
- In addition, a desirable classifier should have:
  - time complexity linear w.r.t to the no. of training examples,

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.
- In addition, a desirable classifier should have:
  - time complexity linear w.r.t to the no. of training examples,
  - directly handle multiple class problems,

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.
- In addition, a desirable classifier should have:
  - time complexity linear w.r.t to the no. of training examples,
  - directly handle multiple class problems,
  - directly handle missing values, and

Machine Learning from Big Data

- When data is too big to reside in RAM, machine learning have two options:
  - First, learn from a sample of data, thereby potentially losing information implicit in the data as a whole.
  - Second, process data out-of-core which results in expensive data-access, making single-pass algorithms extremely desirable.
- In addition, a desirable classifier should have:
  - time complexity linear w.r.t to the no. of training examples,
  - directly handle multiple class problems,
  - directly handle missing values, and
  - require minimal parameter tuning.

Bias and Variance for Classification

Bias and Variance for Classification

- Bias: Error due to the central tendency of the learner.

Bias and Variance for Classification

- Bias: Error due to the central tendency of the learner.
- Variance: Error due to the variability in response to sampling.

Bias and Variance for Classification

- Bias: Error due to the central tendency of the learner.
- Variance: Error due to the variability in response to sampling.

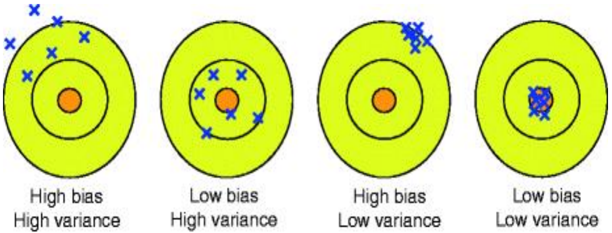


Figure: Image from Bias Variance Decomposition in 'Encyclopedia of Machine Learning', C. Sammut and G.I Webb, Editors 2010, Springer: New York.

Bias and Variance for Classification

- Bias: Error due to the central tendency of the learner.
- Variance: Error due to the variability in response to sampling.

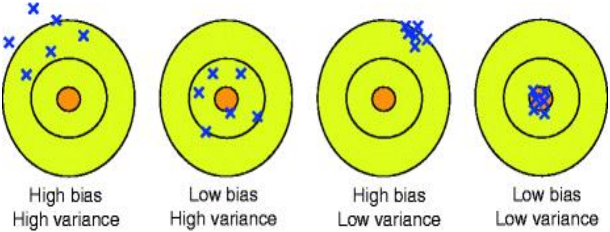


Figure: Image from Bias Variance Decomposition in 'Encyclopedia of Machine Learning', C. Sammut and G.I Webb, Editors 2010, Springer: New York.

- Since for big data, variance tends to decrease anyways as data quantity increases – *low bias algorithms are preferable*.

Averaged *n*-Dependence Estimators (AnDE)

- Averaged *n*-Dependence Estimators (AnDE) family of Bayesian learning algorithms provide efficient single pass learning with accuracy competitive to state-of-the-art in-core learning.

Averaged n-Dependence Estimators (AnDE)

- Averaged n-Dependence Estimators (AnDE) family of Bayesian learning algorithms provide efficient single pass learning with accuracy competitive to state-of-the-art in-core learning.
- $$\hat{\mathrm{P}}_{\mathrm{AnDE}}(y, \mathbf{x}) = \begin{cases} \dfrac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i \mid y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} & : \sum\limits_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) > 0 \\ \hat{\mathrm{P}}_{\mathrm{A(n\text{-}1)DE}}(y, \mathbf{x}) & : \text{otherwise} \end{cases}$$

Averaged n-Dependence Estimators (AnDE)

- Averaged n-Dependence Estimators (AnDE) family of Bayesian learning algorithms provide efficient single pass learning with accuracy competitive to state-of-the-art in-core learning.
- $$\hat{\mathrm{P}}_{\mathrm{AnDE}}(y, \mathbf{x}) = \begin{cases} \dfrac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i \mid y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} & : \displaystyle\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) > 0 \\ \hat{\mathrm{P}}_{\mathrm{A(n\text{-}1)DE}}(y, \mathbf{x}) & : \text{otherwise} \end{cases}$$
- In AnDE, n controls the bias-variance trade-off. Higher n leads to lower bias but higher variance.

Averaged n-Dependence Estimators (AnDE)

- Averaged n-Dependence Estimators (AnDE) family of Bayesian learning algorithms provide efficient single pass learning with accuracy competitive to state-of-the-art in-core learning.
-

$$\hat{\mathrm{P}}_{\mathrm{AnDE}}(y, \mathbf{x}) = \begin{cases} \dfrac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i \mid y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} & : \displaystyle\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) > 0 \\ \hat{\mathrm{P}}_{\mathrm{A(n\text{-}1)DE}}(y, \mathbf{x}) & : \ \text{otherwise} \end{cases}$$

- In AnDE, n controls the bias-variance trade-off. Higher n leads to lower bias but higher variance.
- Unfortunately, large n has high time and space complexity especially as the dimensionality of data increases.

Averaged *n*-Dependence Estimators (AnDE)

- Averaged *n*-Dependence Estimators (AnDE) family of Bayesian learning algorithms provide efficient single pass learning with accuracy competitive to state-of-the-art in-core learning.
- $$\hat{\mathrm{P}}_{\mathrm{AnDE}}(y, \mathbf{x}) = \begin{cases} \dfrac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i | y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} & : \sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) > 0 \\ \hat{\mathrm{P}}_{\mathrm{A(n\text{-}1)DE}}(y, \mathbf{x}) & : \text{otherwise} \end{cases}$$
- In AnDE, *n* controls the bias-variance trade-off. Higher *n* leads to lower bias but higher variance.
- Unfortunately, large *n* has high time and space complexity especially as the dimensionality of data increases.
- How to reduce bias?

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\text{oedema}|\text{female}, \text{pregnant}) = \mathrm{P}(\text{oedema}|\text{pregnant})$

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.
- Simple correction for extreme form of violation of attribute independence assumption.

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.
- Simple correction for extreme form of violation of attribute independence assumption.
- Very effective in practice - reduce bias at small cost in variance.

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.
- Simple correction for extreme form of violation of attribute independence assumption.
- Very effective in practice - reduce bias at small cost in variance.
- For AnDE with $n \geq 1$, it uses statistics collected already - no learning overhead - reduces classification time.

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.
- Simple correction for extreme form of violation of attribute independence assumption.
- Very effective in practice - reduce bias at small cost in variance.
- For AnDE with $n \geq 1$, it uses statistics collected already - no learning overhead - reduces classification time.
- $\mathrm{P}(x_i \mid x_j) = 1$ iff $\#(x_j) = \#(x_i, x_j) > 100$

Subsumption Resolution (SR)

- If $\mathrm{P}(x_1|x_2) = 1.0$ then $\mathrm{P}(y|x_1, x_2) = \mathrm{P}(y|x2)$
- For example, $\mathrm{P}(\mathrm{oedema}|\mathrm{female}, \mathrm{pregnant}) = \mathrm{P}(\mathrm{oedema}|\mathrm{pregnant})$
- Subsumption resolution looks for subsuming attributes at classification time and ignores them.
- Simple correction for extreme form of violation of attribute independence assumption.
- Very effective in practice - reduce bias at small cost in variance.
- For AnDE with $n \geq 1$, it uses statistics collected already - no learning overhead - reduces classification time.
- $\mathrm{P}(x_i \mid x_j) = 1$ iff $\#(x_j) = \#(x_i, x_j) > 100$

Weighted AnDE (WAnDE)

- It has been shown that weighting sub-models can result in reducing the bias in AODE.

Weighted AnDE (WAnDE)

- It has been shown that weighting sub-models can result in reducing the bias in AODE.
- Different weighting schemes have been investigated. A popular one is WAODE due to its minimal computational overhead.

Weighted AnDE (WAnDE)

- It has been shown that weighting sub-models can result in reducing the bias in AODE.
- Different weighting schemes have been investigated. A popular one is WAODE due to its minimal computational overhead.
- $$\hat{\mathrm{P}}_{\mathrm{WAnDE}}(y, \mathbf{x}) = \begin{cases} \dfrac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) w_s \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i | y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} \\ \hat{\mathrm{P}}_{\mathrm{WA(n\text{-}1)DE}}(y, \mathbf{x}) \end{cases}$$

Weighted AnDE (WAnDE)

- It has been shown that weighting sub-models can result in reducing the bias in AODE.
- Different weighting schemes have been investigated. A popular one is WAODE due to its minimal computational overhead.
- $$\hat{\mathrm{P}}_{\mathrm{WAnDE}}(y, \mathbf{x}) = \begin{cases} \frac{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s) w_s \hat{\mathrm{P}}(y, x_s) \prod_{i=1}^{a} \hat{\mathrm{P}}(x_i | y, x_s)}{\sum_{s \in \binom{\mathcal{A}}{n}} \delta(x_s)} \\ \hat{\mathrm{P}}_{\mathrm{WA(n\text{-}1)DE}}(y, \mathbf{x}) \end{cases}$$
- $w_s = \mathrm{MI}(s, Y) = \sum_{y \in Y} \sum_{x_s \in X_s} \mathrm{P}(x_s, y) \log \frac{\mathrm{P}(x_s, y)}{\mathrm{P}(x_s)\mathrm{P}(y)}$

Complexity Analysis

- Complexity at training time: $\mathrm{O}(t\binom{m}{n+1})$, and classification time: $\mathrm{O}(km\binom{m}{n})$, t is the no. of training examples.

Complexity Analysis

- Complexity at training time: $\mathrm{O}(t\binom{m}{n+1})$, and classification time: $\mathrm{O}(km\binom{m}{n})$, t is the no. of training examples.
- Subsumption resolution requires no additional training time. At classification time it requires $\binom{m}{2}$ comparisons to identify any subsumed attribute values.

Complexity Analysis

- Complexity at training time: $\mathrm{O}(t\binom{m}{n+1})$, and classification time: $\mathrm{O}(km\binom{m}{n})$, t is the no. of training examples.
- Subsumption resolution requires no additional training time. At classification time it requires $\binom{m}{2}$ comparisons to identify any subsumed attribute values.
- WAnDE requires the calculation of weights at the training time, $\mathrm{O}(k\binom{m}{n})$. The classification time impact is negligible.

Experimental Details

- Each algorithm is tested on each data set using 20 rounds of 2-fold cross validation. Probability estimates were smoothed using m-estimation with $m = 1$.

Experimental Details

- Each algorithm is tested on each data set using 20 rounds of 2-fold cross validation. Probability estimates were smoothed using m-estimation with $m = 1$.
- Win-draw-loss results are presented. Standard binomial sign test, assuming that wins and losses are equiprobable, is applied to these records. Difference is significant if the outcome of a two-tailed binomial sign test is less than 0.05.

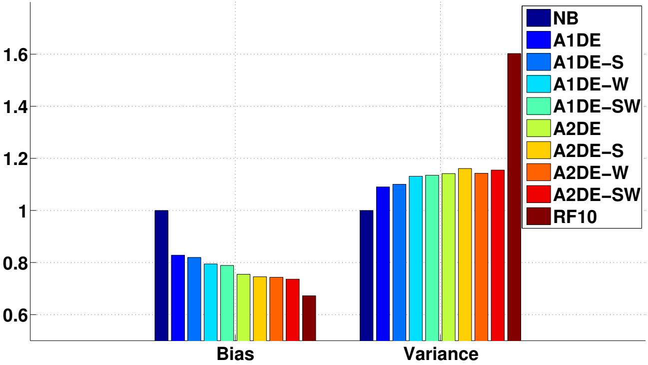
Experimental Details

- Each algorithm is tested on each data set using 20 rounds of 2-fold cross validation. Probability estimates were smoothed using m-estimation with $m = 1$.
- Win-draw-loss results are presented. Standard binomial sign test, assuming that wins and losses are equiprobable, is applied to these records. Difference is significant if the outcome of a two-tailed binomial sign test is less than 0.05.
- The data sets are divided into four categories. First, consisting of all 71 data sets. Second, large data sets with number of instances $> 10,000$. Third, medium data sets with number of instances > 1000 and $< 10,000$. Fourth, small data sets with number of instances < 1000.

Experimental Details

- Each algorithm is tested on each data set using 20 rounds of 2-fold cross validation. Probability estimates were smoothed using m-estimation with $m = 1$.
- Win-draw-loss results are presented. Standard binomial sign test, assuming that wins and losses are equiprobable, is applied to these records. Difference is significant if the outcome of a two-tailed binomial sign test is less than 0.05.
- The data sets are divided into four categories. First, consisting of all 71 data sets. Second, large data sets with number of instances $> 10,000$. Third, medium data sets with number of instances > 1000 and $< 10,000$. Fourth, small data sets with number of instances < 1000.
- Numeric attributes are discretized using MDL discretization for all compared techniques except Random Forest.

Bias and Variance Analysis



0-1 Loss

All Data Sets

	NB	A1DE	A1DE-S	A1DE-W	A1DE-SW	A2DE	A2DE-S	A2DE-W	A2DE-SW
A1DE	**53/4/14**								
A1DE-S	**51/4/16**	**27/31/13**							
A1DE-W	**50/2/19**	35/8/28	29/8/34						
A1DE-SW	**48/3/20**	38/6/27	32/10/29	20/42/9					
A2DE	**54/3/14**	**50/4/17**	**48/4/19**	**45/8/18**	**41/10/20**				
A2DE-S	**49/3/19**	**46/3/22**	**45/4/22**	**44/5/22**	**43/5/23**	23/34/14			
A2DE-W	**48/2/21**	**46/3/22**	**45/4/22**	**47/6/18**	**46/6/19**	36/8/27	35/9/27		
A2DE-SW	**47/2/22**	**45/2/24**	**42/3/26**	**45/7/19**	**44/6/21**	37/9/25	36/11/24	21/34/16	
RF10	40/1/30	28/2/41	26/5/40	**24/2/45**	**24/2/45**	**22/3/46**	**20/4/47**	**17/3/51**	**17/3/51**

Large Data Sets

	NB	A1DE	A1DE-S	A1DE-W	A1DE-SW	A2DE	A2DE-S	A2DE-W	A2DE-SW
A1DE	**12/0/0**								
A1DE-S	**12/0/0**	7/4/1							
A1DE-W	**12/0/0**	**9/2/1**	7/1/4						
A1DE-SW	**12/0/0**	**10/1/1**	8/2/2	5/6/1					
A2DE	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	**11/0/1**				
A2DE-S	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	**7/5/0**			
A2DE-W	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	9/1/2	5/1/6		
A2DE-SW	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	**12/0/0**	9/1/2	8/1/3	**6/6/0**	
RF10	**12/0/0**	9/0/3	9/0/3	9/0/3	9/0/3	7/1/4	6/1/5	5/1/6	5/1/6

0-1 Loss (Contd)

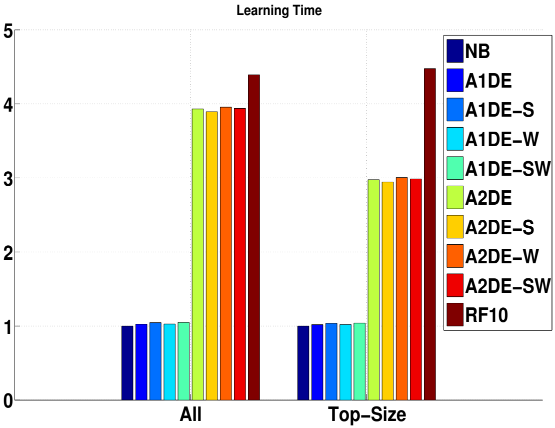
Medium Data Sets

	NB	A1DE	A1DE-S	A1DE-W	A1DE-SW	A2DE	A2DE-S	A2DE-W	A2DE-SW
A1DE	**18/1/0**								
A1DE-S	**19/0/0**	7/5/7							
A1DE-W	**19/0/0**	13/1/5	10/3/6						
A1DE-SW	**18/1/0**	12/1/6	10/4/5	5/8/6					
A2DE	**19/0/0**	**17/0/2**	**15/1/3**	11/1/7	11/1/7				
A2DE-S	**19/0/0**	**16/0/3**	**14/1/4**	12/1/6	12/1/6	6/9/4			
A2DE-W	**19/0/0**	**17/0/2**	**16/2/1**	**15/2/2**	**14/2/3**	**13/3/3**	**13/3/3**		
A2DE-SW	**19/0/0**	**16/0/3**	**14/1/4**	**14/2/3**	**14/2/3**	11/4/4	11/5/3	5/7/7	
RF10	**15/0/4**	10/0/9	8/3/8	6/1/12	6/1/12	6/1/12	5/2/12	**4/1/14**	**4/1/14**

Small Data Sets

	NB	A1DE	A1DE-S	A1DE-W	A1DE-SW	A2DE	A2DE-S	A2DE-W	A2DE-SW
A1DE	23/3/14								
A1DE-S	20/4/16	13/22/5							
A1DE-W	19/2/19	13/5/22	12/4/24						
A1DE-SW	18/2/20	16/4/20	14/4/22	**10/28/2**					
A2DE	23/3/14	21/4/15	21/3/16	22/7/11	19/9/12				
A2DE-S	18/3/19	18/3/19	19/3/18	20/4/16	19/4/17	10/20/10			
A2DE-W	17/2/21	17/3/20	17/2/21	20/4/16	20/4/16	14/4/22	17/5/18		
A2DE-SW	16/2/22	17/2/21	16/2/22	19/5/16	18/4/18	17/4/19	17/5/18	10/21/9	
RF10	13/1/26	**9/2/29**	**9/2/29**	**9/1/30**	**9/1/30**	**9/1/30**	**9/1/30**	**8/1/31**	**8/1/31**

Averaged Learning Time



Conclusion

- Both SR and weighting are just as effective at reducing A2DE's bias as it is at reducing A1DE's.

Conclusion

- Both SR and weighting are just as effective at reducing A2DE's bias as it is at reducing A1DE's.
- There is strong synergy between the two techniques and that they operate in tandem to reduce the bias of both A1DE and A2DE more effectively than does either in isolation.

Conclusion

- Both SR and weighting are just as effective at reducing A2DE's bias as it is at reducing A1DE's.
- There is strong synergy between the two techniques and that they operate in tandem to reduce the bias of both A1DE and A2DE more effectively than does either in isolation.
- We compared A2DE with MI-weighting and subsumption resolution against the state-of-the-art in-core learning algorithm Random Forest.

Conclusion

- Both SR and weighting are just as effective at reducing A2DE's bias as it is at reducing A1DE's.
- There is strong synergy between the two techniques and that they operate in tandem to reduce the bias of both A1DE and A2DE more effectively than does either in isolation.
- We compared A2DE with MI-weighting and subsumption resolution against the state-of-the-art in-core learning algorithm Random Forest.
- Using only single-pass learning, A2DE with MI-weighting and subsumption resolution achieves accuracy that is very competitive with the state-of-the-art in in-core learning, making it a desirable algorithm for learning from very large data.

Conclusion

- Both SR and weighting are just as effective at reducing A2DE's bias as it is at reducing A1DE's.
- There is strong synergy between the two techniques and that they operate in tandem to reduce the bias of both A1DE and A2DE more effectively than does either in isolation.
- We compared A2DE with MI-weighting and subsumption resolution against the state-of-the-art in-core learning algorithm Random Forest.
- Using only single-pass learning, A2DE with MI-weighting and subsumption resolution achieves accuracy that is very competitive with the state-of-the-art in in-core learning, making it a desirable algorithm for learning from very large data.
- Code is available as weka package online.