







### 5. The Complexity of the Algorithm

The independence assumption, Naive Bayes classifiers can quickly learn to use high dimensional features with limited training data compared to more sophisticated methods.

Let  $n$  is number of training examples,  $v^*$  is dimensionality of the features and  $k$  is number of classes.

All it needs to do is computing the frequency of every feature value  $v^*$  for each class, but space complexity of training is  $O(k.v^*.n)$  since you need to store the data which also takes time.

The computational complexity efficiency of Naive Bayes lies in the fact that the runtime complexity of Naive Bayes classifier is  $O(k.v^*.n)$ .

### 6. Applications of LDAB Algorithm

#### 6.1. Computational

$$p(Y) = \frac{3}{4}, p(N) = \frac{1}{4}$$

#### • TRAINING

e1:x1	2	1	2	2	2	2	
e2:x2	2	2	2	1	2	2	
e3:x3	2	2	2	2	2	2	$d =  V  = 6$
Total	6	5	6	5	6	6	$\Rightarrow N_Y = 34$
$\Rightarrow \lambda_Y$	7/40	6/40	7/40	6/40	7/40	7/40	$40 = N_Y +  V $

Class Y

e4:x4	0	0	1	0	2	1	$\Rightarrow N_N = 4$
$\Rightarrow \lambda_N$	1/10	1/10	2/10	1/10	3/10	2/10	$10 = N_N +  V $

Class N

#### • TEST

e5:x5 = [1, 1, 2, 1, 2, 1]

$$p(Y|e5) \propto p(Y) \cdot \prod_{i=1}^d p(x_i|Y) = \frac{3}{4} \times \frac{7}{40} \times \frac{6}{40} \times \left(\frac{7}{40}\right)^2 \times \frac{6}{40} \times \left(\frac{7}{40}\right)^2 \times \frac{7}{40} \approx 4.8 \times 10^{-7}$$

$$p(N|e5) \propto p(N) \cdot \prod_{i=1}^d p(x_i|N) = \frac{1}{4} \times \frac{1}{10} \times \frac{1}{10} \times \left(\frac{2}{10}\right)^2 \times \frac{1}{10} \times \left(\frac{3}{10}\right)^2 \times \frac{2}{10} \approx 1.8 \times 10^{-7}$$

$$p(x5|Y) > p(x5|N) \Rightarrow e5 \in Y$$

Test data

#### 6.2. Probability

$$p(Y|e5) = \frac{4.8 \times 10^{-7}}{4.8 \times 10^{-7} + 1.8 \times 10^{-7}} \approx 0.7273$$

$$\Rightarrow p(N|e5) = 1 - p(Y|e5) \approx 0.2727 \text{ and } e5 \in Y.$$

#### 6.3. Using the Naïve Bayes model in Python

Supervised learning lets us make predictions based on the data that we see and thus apply generalisations

```
>>> from sklearn.naive_bayes import MultinomialNB
>>> import numpy as np
>>> e1 = [2, 1, 2, 2, 2, 2] #input – training data
```

```
>>> e2 = [2, 2, 2, 1, 2, 2]
>>> e3 = [2, 2, 2, 2, 2, 2]
>>> e4 = [0, 0, 1, 0, 2, 1]
>>> training_data = np.array([e1, e2, e3, e4])
>>> result_data = np.array(['Y', 'Y', 'Y', 'N'])
>>> e5 = np.array([[1, 1, 2, 1, 2, 1]]) #test data
>>> ml = MultinomialNB(alpha=1) #call MultinomialNB
>>> ml.fit(training_data, result_data) #process model
MultinomialNB(alpha=1, class_prior=None, fit_prior=True)
>>> print('Probability of e5:', ml.predict_proba(e5))
Probability of e5: [[0.27079929 0.72920071]]
#output - new set of data and evaluate the accuracy
>>> print('Predicting class of e5:', str(ml.predict(e5)[0]))
Predicting class of e5: Y
```

## 7. Conclusion

In this article, we build a model for learning data analytics from applying the Naïve Bayes probability formula with the purpose of modeling from real problems that can be applied accurately and effectively. Next, we propose to build an algorithm for learning data analytics and data needs to be tested correctly. Finally, a specific example is presented in detail to illustrate the LDAB algorithm as well as its complexity calculation.

## Acknowledgements

The author would like to thank the Ministry of Technology - Higher Education and President, Van Lang University for financial support to this research.

## References

- [1] Webb, G.I., Boughton, J., Wang, Z (2005). Not so naive Bayes: Aggregating onedependence estimators. *Machine Learning*, 5–24.
- [2] Witten, I.H., Frank, E (2000). *Data mining: practical machine learning tools and techniques with Java implementations*. San Francisco, CA: Morgan Kaufmann.
- [3] A. S. Thanuja Nishadi (2019). *Text Analysis: Naïve Bayes Algorithm using Python JupyterLab*, International Journal of Scientific and Research Publications, Issue 11.
- [4] Jake VanderPlas (2014). *Frequentism and Bayesianism: A Python-driven Primer*.
- [5] Bhargava, K. Tara Phani (2018). *Analysis and Design of Visualization of Educational Institution Database using Power BI Tool*.
- [6] Abdous, M., He, W., & Yen, C (2012). Using data mining for predicting relationships between online question theme and final grade. *Educational Technology & Society*, 15(3), 77-88.
- [7] Brandon, K (2009). *Investing in Education: The American Graduation Initiative*. Office of Social Innovation and Civic Participation.
- [8] Campbell, J. P., DeBlois, P. B., & Oblinger (2007). *Academic analytics: A new tool for a new era*. *EDUCAUSE Review*, 42(4), 40-57.
- [9] Dietz-Uhler, B., Hurn, J.E., & Hurn (2012). *Making use of data in an LMS to predict student performance: A learning analytics investigation*. Unpublished manuscript.
- [10] Dringus, L. P (2012). Learning analytics considered harmful. *Journal of Asynchronous Learning Networks*, 16(3), 87-100.
- [11] Dyckhoff, A. L., Zielke, D., Bultmann, M., Chatti, M. A., & Schroeder (2012). Design and implementation of a learning analytics toolkit for teachers. *Educational Technology & Society*, 15(3), 58-76.
- [12] Brain, D., Webb, G.I. (2002). The need for low bias algorithms in classification learning from large data sets. In: *Proc. 16th European Conf. Principles of Data Mining and Knowledge Discovery (PKDD2002)*, Berlin:Springer-Verlag, 62–73.
- [13] Dziuban, C., Moskal, P., Cavanaugh, T., & Watts, A (2012). Analytics that inform the university: Using data you already have. *Journal of Asynchronous Learning Networks*, 16(3), 21-38.
- [14] Greller, W. & Drachslrer (2012). *Translating learning into numbers: A generic framework for learning analytics*.
- [15] Ice, P., Diaz, S., Swan, K., Burgess, M., Sharkey, M., Sherrill, J., Huston, D., & Okimoto, H (2012). The PAR framework proof of concept: Initial findings from a multi-institutional analysis of federated postsecondary data. *Journal of Asynchronous Learning Networks*, 16(3), 63-86.
- [16] Long, P. & Siemens, G (2011). *Penetrating the fog: Analytics in Learning and Education*.
- [17] Webb, G.I. (2001). Candidate elimination criteria for lazy Bayesian rules. In: *Proc. Fourteenth Australian Joint Conf. Artificial Intelligence*. Volume 2256., Berlin:Springer, 545–556.
- [18] Xie, Z., Hsu, W., Liu, Z., Lee, M.L.(2002). Snnb: A selective neighborhood based naive Bayes for lazy learning. In: *Advances in Knowledge Discovery and Data Mining, Proc. Pacific-Asia Conference*, Berlin:Springer, 104–114.
- [19] Zhang, N. L (2004). Hierarchical latent class models for cluster analysis. *Journal of Machine Learning Research* 5:697–723.

- [20] Fayyad, U.M., Irani, K.B. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In: Proc. 13th Int. Joint Conf. Artificial Intelligence (IJCAI-93), Morgan Kaufmann, 1022–1029.
- [21] Jones, S. J (2012). Technology review: The possibilities of learning analytics to improve learner centered decision making. The Community College Enterprise, Spring, 89-92.
- [22] Zheng, Z., and Webb, G. I (2000). Lazy learning of bayesian rules. Journal of Machine Learning 41(1):53–84.
- [23] Koller, D (2013). Probabilistic Graphical Models. Coursera Stanford – Lectures. Retrieved December 6, from <https://class.coursera.org/pgm/lecture>
- [24] Koller, D., & Friedman, N. (2009). Probabilistic graphical models: Principals and techniques MIT Press. ISBN 978-0262013192
- [25] Koller, D., Friedman, N., Getoor, L., & Taskar, B. (2007). Graphical models in a nutshell. In L. Getoor, & B. Taskar (Eds.), An introduction to statistical relational learning () MIT Press.
- [26] Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers.
- [27] Jurafsky, Dan and Manning, Christopher (2014). Natural Language Processing. Coursera Stanford - Lectures. Retrieved April 5, from <https://class.coursera.org/nlp/lecture/28>
- [28] Naive Bayes for classifying Text (2014). Retrieved April 5, from <http://www.cs.nyu.edu/faculty/davise/ai/BayesText.html>
- [29] Chow, C.K. & C.N. Liu (1968). Approximating discrete probability distributions with dependence trees. IEEE Trans. on Info. Theory, 14, 462- 467.

## Authors Profile



**Dr. Tran Ngoc Viet.** Born in 1973 in Thanh Khe District, Da Nang City, Vietnam. He graduated from Maths\_IT faculty of Da Nang university of science. He got master of science at university of Danang and hold Ph.D Degree in 2017 at Danang university of technology. His main major: Applicable mathematics in transport, parallel and distributed process, data science, machine learning, graph theory and distributed programming.



**Dr. Hoang Le Minh** was born in 1957 in Ha Noi, Vietnam. He got Ph.D Degree in 1984 at Matxcova university of science. He is currently a lecturer at the Faculty of Information Technology at Van Lang University. His main major: Graph theory, data science, machine learning, big data analytics, data pre-processing and analysis, distributed programming.



**Le Cong Hieu** was born in 1979. He graduated from information technology faculty of Ho Chi Minh university of science. He has a master's degree in computer science, graduated in 2014 of Hue university. He is currently a lecturer at the Faculty of Information Technology at Van Lang University. His main major: discrete mathematics, graph theory, machine learning and distributed programming.



**Tong Hung Anh** was born in 1964 in Hue, Vietnam. He has a master's degree in computer networking, graduated in 2011 university Paris VI. He is currently a lecturer at the Faculty of Information Technology at Van Lang university. His main major: Python for machine learning, distributed programming, mathematics and statistics for data science, data pre-processing and analysis.