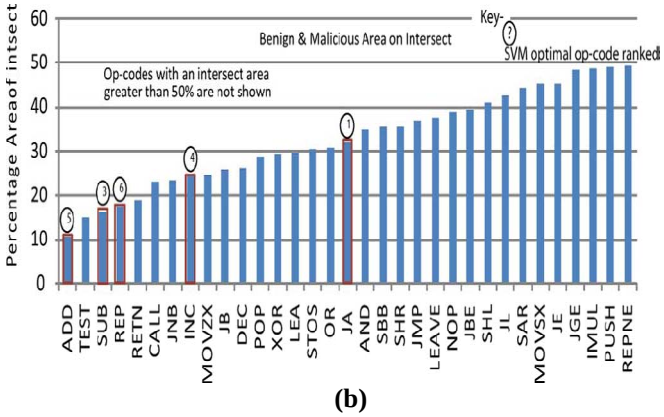

(b)

Fig. 4. (a) Ideal characteristics. (b) Benign and malicious area of interest.

difference between the population density and variance of malicious and benign software and therefore it would be highly speculative to infer any meaning to *add* and *sub*.

6. Conclusion

This paper, proposes the use of SVM as a means of identifying malware. It shows that malware, that is packed/encrypted, can be detected using SVMs and by using the opcodes chosen by the SVM as a benchmark, determined a prefilter stage using eigenvectors that can reduce the feature set and therefore reduce the training effort. The results presented in this paper exposed three key points. Firstly, the identification of a high population opcode: *mov* that is not only is a poor indicator of benign/malicious software, but inhibits the ability to correctly classify software when used with other opcodes such as *ja, adc, sub, inc, add* and *rep*. Secondly, a subset of opcodes can be used to detect malware. However, the SVM analysis demonstrates that *ja, adc* and *sub* are strong indicators of malware as they are four times more likely to be used in the correct classification of malware than the next most significant opcodes *(inc)*. Several opcodes have been identified as potential indicators of malware, which provides the basis for an improvement in detection techniques beyond current state of the art [22]. Finally, using the 'eigenvector' prefilter, irrelevant features are safely removed by dataset.

References

[1] A. Lakhotia, E. U. Kumar, and M. Venable, "A method for detecting obfuscated calls in malicious binaries," *IEEE Trans. Software Eng.*, vol. 31, no. 11, pp. 955–968, Nov. 2005.

[2] D. Bilar, "Opcodes as predictor for malware," *Int. J. Electron. Security Digital Forensics*, vol. 1, no. 2, pp. 156–168, 2007.

[3] D. Bilar, "Callgraph properties of executables and generative mechanisms," *AI Commun., Special Issue on Network Anal. in Natural Sci. and Eng.*, vol. 20, no. 4, pp. 231–243, 2007. [4] I. Santos, Y. K. Penya, J. Devesa, and P. G. Garcia, "N-grams-based file signatures for malware detection," *S3Lab, Deusto Technological Found.*, 2009[Online].

[4] Available: pgbg@technologico.deusto.es

[5] R. Sekar, M. Bendre, D. Bollineni, and Bollineni, R. Needham and M. Abadi, Eds., "A fast automaton-based method for detecting anomalous program behaviors," in *Proc. 2001 IEEE Symp. Security and Privacy, IEEE Comput. Soc.*, Los Alamitos, CA, USA, 2001, pp. 144–155.

[6] W. L. K. Wang, S. Stolfo, and B. Herzog, "Fileprints: Identifying file types by n-gram analysis," in *Proc. 6th IEEE Inform. Assurance Workshop*, Jun. 2005, pp. : 64–71.

[7] I. Santos, F. Brezo, J. Nieves, Y. K. Penya, B. Sanz, C. Laorden, and P. G. Bringas, "Opcode-sequence-based malware detection," in *Proc. 2nd Int. Symp. Eng. Secure Software and Syst. (ESSoS)*, Pisa, Italy, Feb. 3–4, 2010, vol. LNCS 5965, pp. 35–43.

[8] I. Santos, F. Brezo, B. Sanz, C. Laorden, and Y. P. G. Bringas, "Using opcode sequences in single-class learning to detect unknown malware," *IET Inform. Security*, vol. 5, no. 4, pp. 220–227, 2011.

[9] I. Santos, F. Brezo, X. Ugarte-Pedrero, and Y. P. G. Bringas, "Opcode sequences as representation of executables for data-mining-based unknown malware detection," *Inform. Sci.*, 2011 [Online]. Available: http://dx.doi.org/10.1016/j.ins.2011.08.020

[10] A. Shabtai, R. Moskovitch, C. Feher, S. Dolev, and Y. Elovici, "Detecting unknown malicious code by applying classification techniques on opcode patterns," *Security Informatics*, vol. 1, pp. 1–22, 2012.

[11] R. Moskovitch, C. Feher, N. Tzachar, E. Berger,M.Gitelman, S.Dolev, and Y. Elovici, "Unknown malcode detection using opcode representation," in *Proc. 1st Eur Conf. Intell. and Security Informatics (EuroISI08)*, 2008, pp. 204–215.

[12] Y. Song, M. Locasto, and A. Stavro, "On the infeasibility of modeling polymorphic shellcode," in *Proc. ACM Conf. Computer and Commun. Security*, 2007, pp. 541–551.

[13] C. Kolbitsch, P. M. Comparetti, C. Kruegel, E. Kirda, X. Zhou, and X. Wang, "Effective and efficient malware detection at the end host," in *Proc. 18th Usenix Security Symp.*, 2009, pp. 351–366.

[14] P. Ferrie, The ultimate anti debugge reference May 2011 [Online]. Available: http://pferrie.host22.com/papers/antidebug.pdf

[15] X. Chen, "Towards an understanding of anti-virtualization and anti debugging behavior in modern malware," *ICDSN Proc.*, pp. 177–186, 2008.

[16] B. E. Bernhard, G. M. Isabelle, and V. N. Vladimir, H. Haussler, Ed., "A training algorithm for optimal margin classifiers," in *Proc. 5th Ann. ACM Workshop on COLT ACM Press*, Pittsburgh, PA, USA, 1992, pp. 144–152.

[17] C. Ko, M. Ruschitzka, and K. Levitt, "Execution monitoring of security- critical programs in distributed systems: A specification-based approach," in *Proc. 1997 IEEE Symp. Security and Privacy*, Oakland, CA, USA, May 1997, p. 175-1 87.

[18] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, A ractical Guide to Support Vector Classification, Department of Computer Science National Taiwan University, Taipei, Taiwan, Apr. 15, 2010 [Online]. Available: http://www.csie.ntu.edu.tw/

[19] R. Vanderbei, *Linear Programming: Foundations and Extensions Pub*. New York, NY, USA: Springer, 2000,

ISBN: 0792373421.

[20] Matlab Statistics Toolbox Oct. 2011 [Online]. Available: http://www.mathworks.co.uk/help/toolbox/stats/

Author Profile

Mr. Dasu Vaman Ravi Prasad is working as associate professor in Computer Science Engineering from CVSR College of Engineering from Anurag Group of Institutions Venkatapur (V), Ghatkesar(M), Ranga Reddy District, Hyderabad-500088, Telangana State.

Mr. Pagidimarri Venu received the B.Tech degree in Computer Science of Engineering from JNTU Anantapur in 2011 and now pursuing M.Tech. degree in Computer science from CVSR College of Engineering in JNTU Hyderabad