Performance Evaluation of Support Vector Machines with Quantum-inspired Annealers

Ryoga Fukuhara† Makoto Morishita† Takahiro Katagiri‡

Masatoshi Kawai‡ Tetsuya Hoshino‡ Toru Nagai‡

Graduate School of Informatics, Nagoya University† Information Technology Center, Nagoya University‡

1 Introduction

Quantum computers, drawing considerable attention due to their capacity for simultaneous parallel computations stemming from their quantum nature, are poised to emerge as the next-generation high-speed computing systems, boasting vastly superior computational capabilities compared to conventional, or classical, computers. This expectation arises from the prediction that Moore's Law will eventually plateau, rendering substantial speed improvements in classical computers unattainable. Meanwhile, the demands for data processing continue to surge, intensifying the need for high-performance computing solutions, thus elevating expectations for novel computer paradigms. Consequently, there is a global acceleration in the development of quantum computers. Conversely, when it comes to addressing combinatorial optimization problems that involve finding optimal combinations of variables to enhance a specific metric among multiple options within various constraints, there is a diversification of quantum annealing methods, semiconductor annealing machines, and other quantum-related hardware. Notably, semiconductor annealing machines have garnered attention as non-von Neumann computers capable of performing annealing processes to rapidly derive optimal solutions for combinatorial optimization problems at room temperature. Nevertheless, the range of practical applications for these machines is not yet extensive. As a result, in this study, we undertook the implementation of SVM (Support Vector Machine) [1], a well-established machine learning algorithm, on a CMOS annealing machine, which falls under the category of quantum-inspired annealers.

2 Performance Evaluation

In this performance evaluation, we address the task of generating random two-dimensional numbers and categorizing them into two classes based on whether they exceed or fall below a specified function threshold. We create three distinct problem types by altering the set of functions: one that is linearly separable and two that are not. Table 1 presents the accuracy of each problem as evaluated in both classical and CMOS annealers.

Table 1: Summary of accuracies (without error)

Problem kinds		Linearly separatable	Linearly non-separatable (1)	Linearly non-separatable (2)
Classical	Accuracy	98.4%	95.6%	98.8%
CMOS annealing	Accuracy Specify Parameters $(B, K, \gamma,\ \xi)$	**94.4%** (10, 3, 0.01, 0)	**94.3%** (10, 2, 0.01,100)	**96.6%** (10, 3, 100, 10)

3 Conclusion

In this study, we conducted an evaluation of Support Vector Machines (SVM) within a quantum-inspired annealer environment. Utilizing the CMOS annealer developed by Hitachi as a quantum-inspired annealer. We formulated artificial problems encompassing two categories of binary classification problems: those that are linearly separable and those that are linearly non-separatable. We subsequently evaluated the performance of SVM within this context. The results of our evaluation conducted using the Amplify, a quantum-inspired annealer environment in the cloud. Furthermore, upon conducting performance evaluation in a classical computing environment, we found that the accuracy rates matched those achieved in the quantum-inspired annealer environment. However, our evaluation also highlighted the crucial role of adjusting SVM-specific hyperparameters during execution within a quantum-inspired annealer, as this adjustment had a significant impact on solution accuracy. Consequently, autotuning [2] this hyperparameter adjustment process is imperative, given that manual adjustment incurs human and time costs.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number JP19H05662.

REFERENCES

[1] V. N. Vapnik, “The Nature of Statistical Learning Theory. Springer”, New York, 1995.

[2] T. Katagiri and D. Takahashi, “Japanese Autotuning Research: Autotuning Languages and FFT”, Proc. of the IEEE, 106(11), pp. 2056-2067, 2018.