

# SVM による誤差を含むクラス分類における 多種疑似量子アニーラの性能評価

水木直也<sup>1</sup> 森下誠<sup>1</sup> 河合直聰<sup>2</sup> 片桐孝洋<sup>2</sup> 棕木大地<sup>2</sup>  
星野哲也<sup>2</sup> 永井亨<sup>2</sup>

**概要**：近年、量子アニーリングが注目されており、その中でも疑似量子アニーラのクラウドサービスが各種展開されている。だが、疑似量子アニーラによるサポートベクターマシン（SVM）の評価は多くはない。そこで本報告では、4種類の疑似量子アニーラ（HITACHI CMOS アニーリングマシン、Fixstars Amplify Annealing Engine、NEC Vector Annealing サービス、Toshiba SQBM+）により、SVM による誤差を含むクラス分類を実装し、性能評価を行った。性能評価の結果、分類精度については古典コンピュータによる SVM に近い性能が得られる場合があった。一方で、Time to Solution(TTS)の観点の評価では、疑似量子アニーラによって大きな差が見られた。また、本実験のデータセットに限っては、古典が良い性能を示した。

**キーワード**：疑似量子アニーラ、SVM、2値分類問題、パラメータチューニング

## 1. はじめに

近年、量子コンピュータが注目されている。量子コンピュータは従来のコンピュータと異なる原理に基づいて動作し、特定の問題において従来のコンピュータの性能を上回ると期待されている。そのため、古典コンピュータとの性能評価の提示が強く求められている。

量子コンピュータの一種である量子アニーリングマシンは、アニーリングの原理を活用することで組合せ最適化問題の効率的な求解に活用できる。だが、量子アニーリングマシンの計算に使用する量子ビットは大量に用意するのが難しく、そのため大規模計算が困難となる。また、安定動作のためには一般的には-273 度程の低温環境が必要である。そのため、大規模な冷却装置が必要となるため、容易な設置を妨げる原因の 1 つとなっている。

このような背景の中、量子アニーリングマシンの動作を従来の古典的なコンピュータで模倣する、疑似量子アニーラが開発されている。疑似量子アニーラは、古典コンピュータの活用により、大規模計算や常温での動作を可能としている。また、古典コンピュータで用いる半導体技術を活用できるため多くの利点がある。現在、疑似量子アニーラを提供するクラウドサービスが複数展開されている。

一方、量子アニーラで対象となる応用問題を考えると、現在ニーズが高まっている機械学習関連の処理に対して、量子や疑似量子アニーラを活用して高精度な解を高速に求める要求が高まっていると言える。この機械学習アルゴリズムの主要なアルゴリズムの一つに、サポートベクターマシン（SVM）[1]がある。SVM は広くクラス分類に使われている汎用的な機械学習の手法である。それにもかかわらず、

量子と疑似量子アニーラを利用した SVM を活用した詳細な性能評価はあまりないといえる。

そこで本研究では、疑似量子アニーラのクラウドサービスを活用し、国内で主流となる複数の疑似量子アニーラの評価を目的とする。本研究で対象とする疑似量子アニーラは、HITACHI CMOS アニーリングマシン[2]、Fixstars Amplify Annealing Engine (Fixstars Amplify AE) [3]、NEC Vector Annealing サービス (NEC VA) [4]、Toshiba SQBM+[5] の 4 種類の疑似量子アニーラである。これらの疑似量子アニーラを用いて、SVM による誤差を含むクラス分類の問題を実装して、分類精度の評価を行う。また、組合せ問題の性能評価として用いられる Time to Solution (TTS)を評価基準に採用し、分類精度と求解速度の観点での性能評価を行う。

本報告の構成は、以下のとおりである。2 節で SVM の概要を、線形分離可能な問題と、線形分離不可能な問題から説明する。3 節では、今回使用する疑似量子アニーラの説明をする。4 節では、性能評価の詳細を示す。5 節で関連研究についてまとめる。最後に本報告のまとめを行う。

## 2. サポートベクターマシン（SVM）の概要

SVM とは、最も近い点までの距離が遠くなるように決定境界を決める分類手法である。アルゴリズム概要について、文献[6]をもとに、線形分離可能な場合と不可能な場合に分けて説明する。

### 2.1 SVM のアルゴリズム(線形分離可能)

$N$  次元空間上の集合を  $N-1$  次元の超平面で分離できること

<sup>1</sup> 名古屋大学 大学院情報学研究科  
Graduate School of Informatics, Nagoya University.  
<sup>2</sup> 名古屋大学 情報基盤センター  
Information Technology Center, Nagoya University.

を線形可能と定義する[6].

超平面の方程式は

$$W^T X + b = 0 \quad \cdots(1)$$

と表せる.  $X \in \mathbb{R}^n$ ,  $W \in \mathbb{R}^n$ ,  $b \in \mathbb{R}$ である.

$i$ 番目のデータ  $X_i$  がクラス 1 に属するとき 1, クラス 2 に属するとき -1 を返す  $t_i \in \mathbb{R}$  を導入すると, 条件式は

$$t_i(W^T X_i + b) > 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(2)$$

となる.

$N$  次元空間上の 1 点と超平面との距離  $d$  は

$$d = \frac{|w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b|}{\sqrt{w_1^2 + w_2^2 + \dots + w_n^2}} = \frac{|W^T X_i + b|}{\|W\|} \quad \cdots(3)$$

なので, 式(1)-式(3)から, マージン  $M$  を最大化するという条件は

$$\max_{w, b} M \frac{t_i(W^T X_i + b)}{\|W\|} \geq M \quad (i = 1, 2, 3, \dots, N) \quad \cdots(4)$$

と表せる. また,  $\|W\| = \frac{1}{M}$  となるように標準化すると

$$\max_{w, b} \frac{1}{\|W\|}, \quad t_i(W^T X_i + b) \geq 1 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(5)$$

となる. 計算量を減らすため以下のように変形する.

$$\min_{w, b} \frac{1}{2} \|W\|^2, \quad t_i(W^T X_i + b) - 1 \geq 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(6)$$

以上が線形分離可能な問題を SVM で解く際の大まかな方法である.

## 2.2 SVM のアルゴリズム(線形分離不可能)

線形分離ができない場合のアプローチは大きく「誤分類を許容する」と「高次元の座標に変換する」の 2 種類がある[6]. 基本的にはこの両方を活用する.

### 2.2.1 誤分類を許容する

ここで制約条件を, 式(7)とする.

$$\min_{w, b} \frac{1}{2} \|W\|^2, \quad t_i(W^T X_i + b) - 1 \geq 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(7)$$

線形分離不可能な場合, 式(7)の制約条件を満たせず, 学習できなくなる. そのため, 式(8)で表すスラック変数  $\varepsilon_i$  を導

入することを考える. このとき,

$$\varepsilon_i = \max \{0, 1 - t_i(W^T X_i + b)\} \quad \cdots(8)$$

となる. スラック変数  $\varepsilon_i$  を制約条件に加え, ある程度の誤分類を許容するようにする. そうすると,

$$t_i(W^T X_i + b) - 1 + \varepsilon_i \geq 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(9)$$

と変形できる.

ここで, スラック変数  $\varepsilon_i$  は, 以下の 3 つの性質を持つ.

1.  $0 \leq \varepsilon_i \leq 1$  のときマージン範囲内である.
2.  $1 < \varepsilon_i$  のとき誤分類が発生する.
3.  $\varepsilon_i$  が大きくなればなるほど誤分類の度合いが大きくなる.

以上より

$$\min_{w, b} \frac{1}{2} \|W\|^2, \quad t_i(W^T X_i + b) - 1 + \varepsilon_i \geq 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(10)$$

を, 誤分類を許容する関数に変換すると,

$$\min_{w, b} \frac{1}{2} \|W\|^2 + C \sum_{i=1}^n \varepsilon_i,$$

$$t_i(W^T X_i + b) - 1 + \varepsilon_i \geq 0 \quad (i = 1, 2, 3, \dots, N) \quad \cdots(11)$$

となる.

このとき, 係数  $C$  が大きいほど誤分類の最小化関数に対する影響が大きくなる. そのため,  $C$  が大きいことは過学習寄りになっていることを示す. 一方,  $C$  が小さいことは未学習であり, 汎化寄りになっていることを示す.

### 2.2.3 高次元の座標に変換する

線形分離不可能な問題を, 写像  $\varphi$  により線形分離可能な高次元の座標に変換した後, 逆変換することで解を求めるが, 射影した後の内積計算のコストが大きいので写像  $\varphi$  は直接定義しない.

本研究では

$$K(X_i, X_j) = \exp \left( -\frac{\|X_i - X_j\|^2}{2\sigma^2} \right) = \exp(-\gamma \|X_i - X_j\|^2) \quad \cdots(12)$$

で定義される RBF カーネル[7]を使用した. ここで式(12)の  $\gamma$  は, 「1 点の学習データが識別面に影響を与える範囲」を表すパラメータである.  $\gamma$  が大きくなるほど 1 点の影響範囲が小さい=曲率が大きな識別面となる.

### 3. 疑似量子アニーラ

疑似量子アニーラとは、量子アニーリングマシンの動作を従来の古典的なコンピュータで模倣するものである。疑似量子アニーラは、組合せ最適化問題の求解に使用できる。

疑似量子アニーラは、量子アニーラとは異なり、量子性を活用していない。しかし、常温での動作が可能、大規模計算に適用が可能など、多くの利点を持っている。そのため実用化が進んでおり、疑似量子アニーラを提供するクラウドサービスが複数展開されている[2][3][4][5]。

表1は、本研究で使用した疑似量子アニーラと、先行研究[8]に用いられた量子アニーラであるD-Wave 2000Qのスペックのまとめである。また、本研究では、Fixstars Amplify SDK[9]という、Fixstars社が提供するイジングマシン向けソフトウェア開発キットを利用して、各疑似量子アニーラを使用している。

表1 各アニーラのスペックのまとめ

装置名	装置形式	最大ビット数	全結合換算ビット数	結合グラフ
D-Wave 2000Q	量子回路	2,048	64	キメラグラフ
CMOS (GPU版, float)	GPU	262,144	512	キンググラフ
Amplify AE	GPU	262,144	131,072	全結合グラフ
NEC VA Engine[10]	Vector Engine	300,000	300,000	全結合グラフ
SQBM+	GPU	10,000,000	10,000,000	全結合グラフ

表1から、D-Wave 2000Qは量子回路を使用しており量子性を持つが、それ以外の疑似量子アニーラはGPU(Graphics Processing Unit)などの古典ハードウェアを使用しているため量子特性を活用していない。

なお、Fixstars Amplify SDKを通じて使用可能なCMOSアニーリングマシンには、GPU版32bit(int), GPU版32bit(float), ASIC版4bitがある[11]。本研究では、GPU版32bit(float)を使用している。

また、CMOSアニーリングマシンの全結合換算ビット数が512で、本研究で使用した疑似量子アニーラの中で一番小さいことが分かる。後述する、疑似量子アニーラにおけるSVMのハイパーパラメータKを、今回2または3としている。Kは、各点を表現するために使用する量子ビットの数であるため、K=3のとき、CMOSアニーラで扱うことの出来る訓練データの数は、170が限界となる。

今回、訓練データの数を100とした。この理由は、このCMOSアニーリングマシンにおけるハードウェア制約に基づいて決定されている点に注意する。

### 4. 性能評価

#### 4.1 問題設定

本性能評価では以下の図1、2のように2次元平面上に乱数を生成し、2種類の関数を基準に2クラスにラベル付けを行うことで、線形分離可能な問題と線形分離不可能な問題を設定して実験を行った。

図1、2の境界面の各関数はそれぞれ、 $y=x$ 、および $(x-50)^2+(y-50)^2=1600$ と設定した。その際、0%から20%の1%刻みの割合で誤ラベルを混入させて、検証用データを作成した。

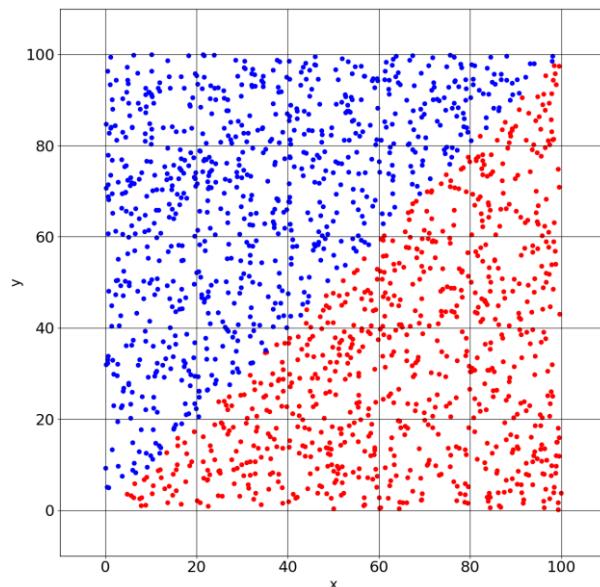


図1 線形分離可能な問題

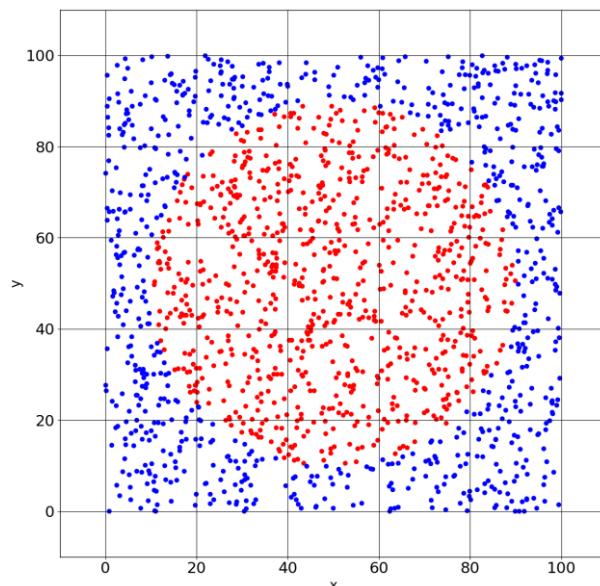


図2 線形分離不可能な問題

ここで誤ラベルを 10%混入させたデータは、図 3, 4 のようになる。

計 1600 個の乱数を生成し、ホールドアウト法に基づき、100 個を訓練データ（図 5, 6）、1000 個を検証データ（図 7, 8）、500 個をテストデータ（図 9, 10）としてクラス分類を行う。

また、SVM のハイパーパラメータの設定については、先行研究[12]を参考に、グリッドサーチによるハイパーパラメータチューニングを行い、その上で、4.2 節で説明する範囲においてチューニングを行う。

今回、データの作成および誤差の混入は、Python プログラムを用いてランダムに行った。そのため、特定の誤ラベル混入率において、極端に分類精度が低くなったり高くなったりする可能性がある。このような偏りを軽減するため、各誤ラベル混入率ごとに 5 セットのデータを作成し、その平均をとった。

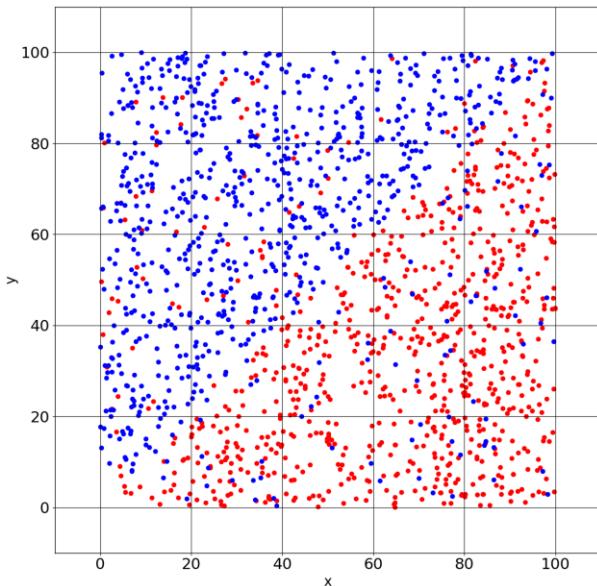


図 3 線形分離可能な問題（10%の誤ラベル）

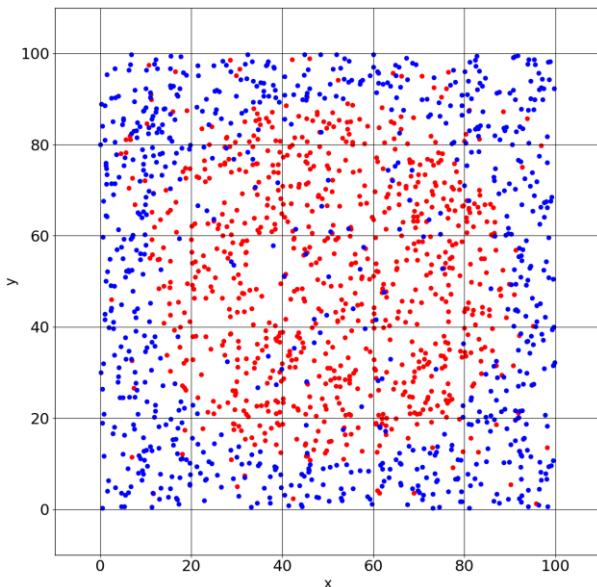


図 4 線形分離不可能な問題（10%の誤ラベル）

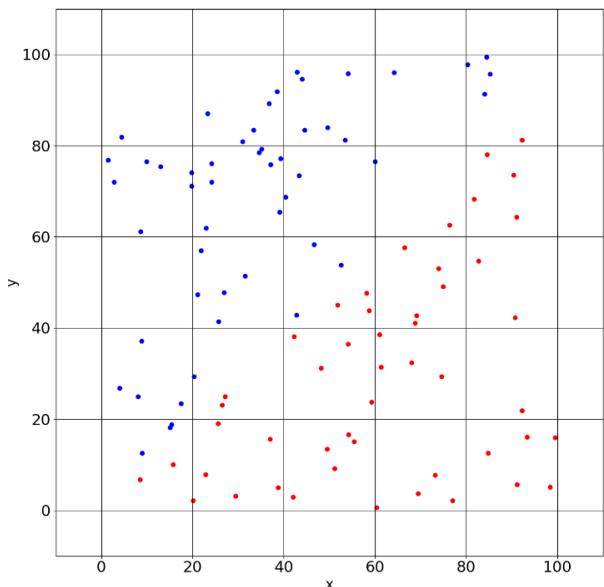


図 5 線形分離可能な問題の訓練データ

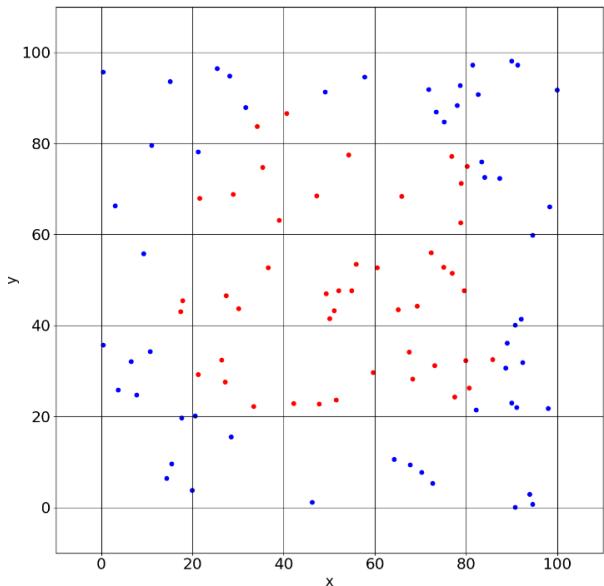


図 6 線形分離不可能な問題の訓練データ

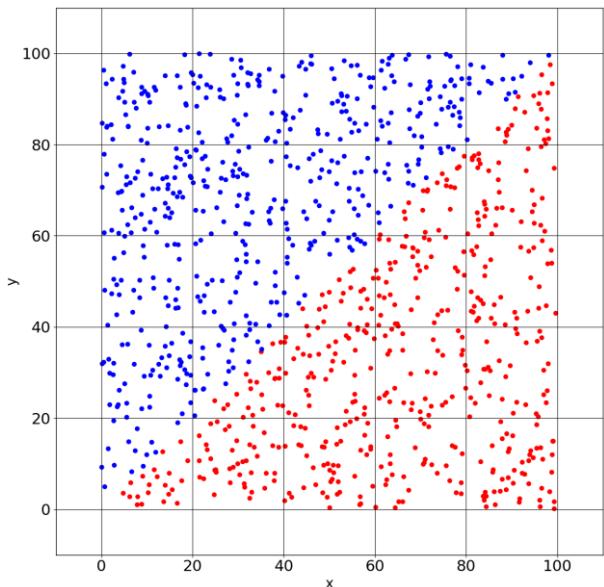


図 7 線形分離可能な問題の検証データ

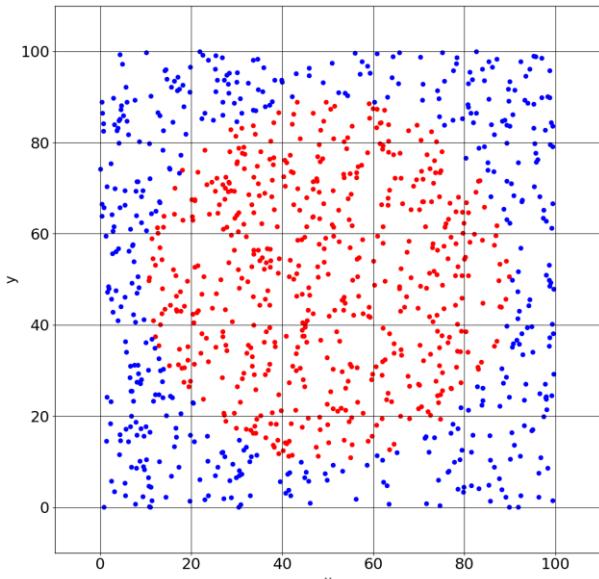


図 8 線形分離不可能な問題の検証データ

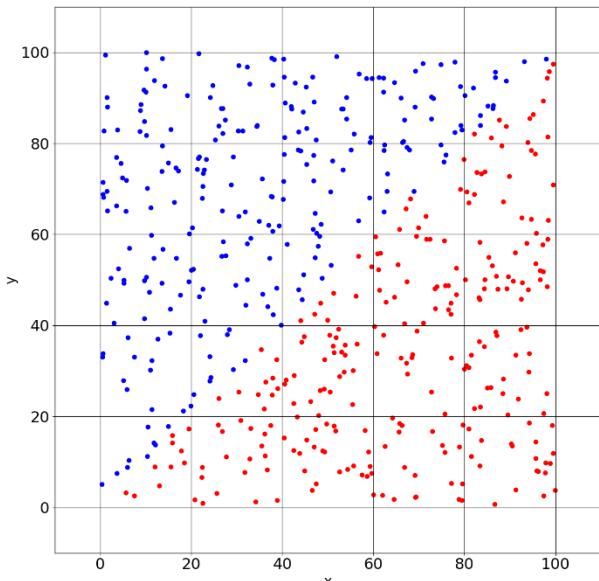


図 9 線形分離可能な問題のテストデータ

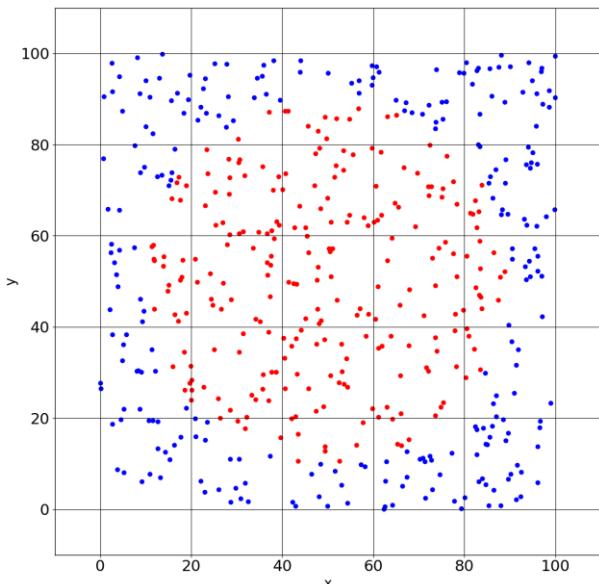


図 10 線形分離不可能な問題のテストデータ

#### 4.2 性能評価の手順

本評価では、疑似量子アニーラで利用するハミルトニアンについて、文献[8]を参考に、以下のハミルトニアンを用いる。

$$E = \frac{1}{2} \sum_{n,m,k,j} a_{K_{n+k}} a_{K_{m+j}} B^{k+j} t_n t_m k(x_n, x_m) -$$

$$\sum_{n,k} B^k a_{K_{n+k}} + \frac{1}{2} \xi (\sum_{n,k} B^k a_{K_{n+k}} t_n)^2 \quad \cdots(14)$$

式(14)では、 $B, K, \gamma, \xi$ の4つのパラメータが出現しており、それらを疑似量子アニーラによる SVM のハイパーパラメータとしている。ここで、 $B$ は符号化の基底、 $K$ は符号化のためのバイナリ変数の数、 $\gamma$ はRBFカーネル関数中の係数、 $\xi$ はエネルギー関数中の第二制約の係数である。

$B, K, \gamma, \xi$ の設定値は、文献[12]を参考に、表2の数値を使用し、グリッドサーチによるハイパーパラメータチューニングを行った。

表 2 疑似量子アニーラによる SVM のハイパーパラメータ設定範囲

パラメータ	設定範囲
$B$	2, 10
$K$	2, 3
$\gamma$	0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000
$\xi$	0, 10, 100

また、古典コンピュータによる SVM のハイパーパラメータとその探索範囲は、表3のようにした。このように設定したのは、疑似量子アニーラによる SVM と古典コンピュータによる SVM のハイパーパラメータ探索点の数（トライアル数）を等しくするためである ( $2 \times 2 \times 8 \times 3 = 16 \times 6 = 96$ )。

表 3 古典コンピュータによる SVM のハイパーパラメータ設定範囲

パラメータ	設定範囲
$C$	0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000, 5000
$\gamma$	0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05

具体的な性能評価の手順は、以下の通りである。

- ① グリッドサーチにより、訓練データから SVM 分類器を作成する。その際、各アニーラの実行時間を測定する。
- ② 作成した分類器を検証データに適用し、分類精度とTTSを算出する。

- ③ ②における分類精度, TTS における上位 10 個の分類器を, テストデータに適用し, 分類精度と TTS を算出する.

古典コンピュータおよび各 4 種の疑似量子アニーラについて, 以上の①～③の手順により, 線形分離可能, 線形分離不可能な問題のそれぞれに対して Time to Solution (TTS) の上位 10 平均と, 分類精度の上位 10 平均に基づき性能評価を行った. 分類精度は, 表 4 に示す混同行列に基づいた計算式(15)のように, TTS は式(16)のように定義した.

表 4 混同行列

	予測 Positive	予測 Negative
正解 Positive	TP (True Positive)	FN (False Negative)
正解 Negative	FP (False Positive)	TN (True Negative)

$$\text{分類精度} = \frac{TP+TN}{TP+FP+TN+FN} \quad \cdots (15)$$

$$TTS = \frac{\log(1-0.8)}{\log(1-p)} T \quad (T = \text{実行時間}, p = \text{分類精度}) \quad \cdots (16)$$

実行環境は, Windows10, Intel Core i9-13900KF, メモリ 32GB 搭載のパソコンで Ubuntu 22.04.3 LTS (WSL2) を使用した. Python はバージョン 3.10.12, Fixstars Amplify SDK はバージョン 1.3.1 を使用した.

各疑似量子アニーラの主なクライアントパラメータ設定は, 以下の表 5-8 のようにした.

表 5 CMOS アニーリングマシンの  
クライアントパラメータ

クライアントパラメータ	設定値
温度変化ステップ数	10
温度ステップ当たりの長さ	100
初期温度	100
最終温度	0.02

表 6 Fixstars Amplify AE のクライアントパラメータ

クライアントパラメータ	設定値
タイムアウト(ms)	1000

表 7 NEC VA のクライアントパラメータ

クライアントパラメータ	設定値
アニーリングの sweep 数	500
アニーリングのモード	Speed
タイムアウト(ms)	10000

表 8 Toshiba SQBM+ のクライアントパラメータ

クライアントパラメータ	設定値
タイムアウト(ms)	1000

### 4.3 実験結果

図 11 および図 12 には, 線形分離可能な問題と線形分離不可能な問題における, 誤ラベル混入率と分類精度の関係を示したグラフを示す. 横軸が誤ラベル混入率, 縦軸が分類精度となる. ここで分類精度は, 各誤差に対して手順③で得られた分類精度の平均値を算出し, それを 5 つのデータセット全体で平均した値である.

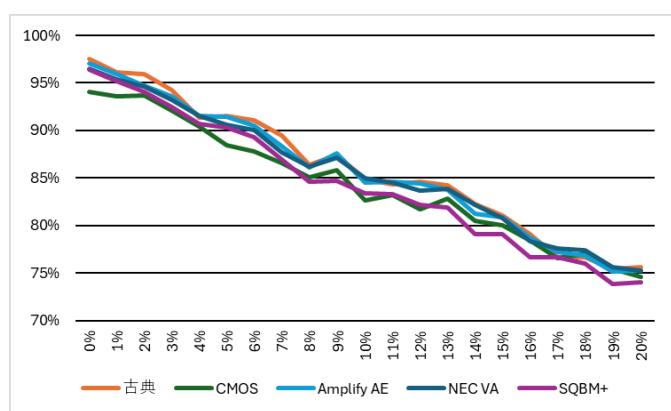


図 11 線形分離可能な問題における,  
誤ラベル混入率と分類精度

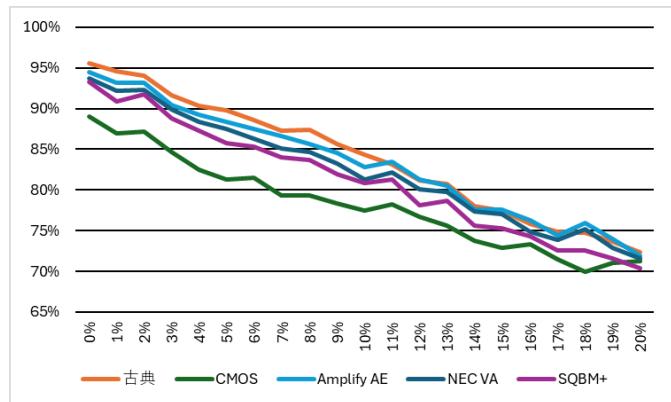


図 12 線形分離不可能な問題における,  
誤ラベル混入率と分類精度

また, 図 13 および図 14 に, 線形分離可能な問題と線形分離不可能な問題における, 誤ラベル混入率と TTS の関係を示したグラフを示す. 横軸が誤ラベル混入率, 縦軸が TTS となる. ここで TTS は, 各誤差に対して手順③で得られた TTS の平均値を算出し, それを 5 つのデータセット全体で平均した値である. なお, 疑似量子アニーラによって TTS に大きなばらつきが見られたため, 縦軸に対数スケールを採用した.

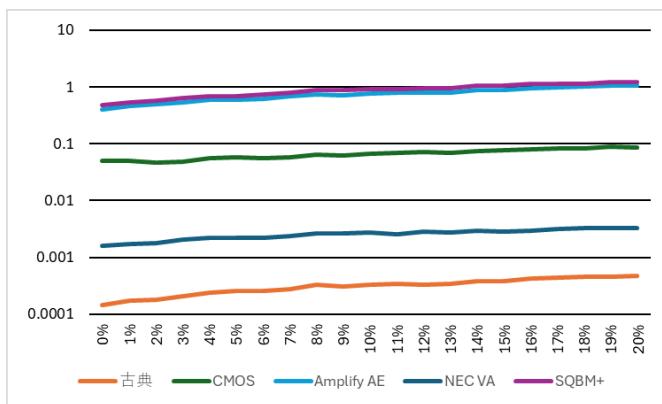


図 13 線形分離可能な問題における、誤ラベル混入率と TTS (秒)

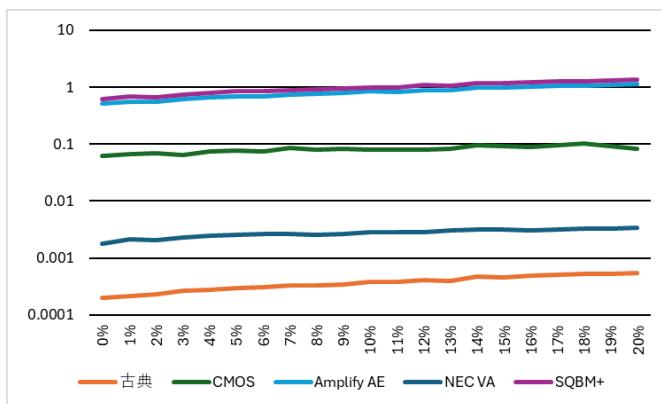


図 14 線形分離不可能な問題における、誤ラベル混入率と TTS (秒)

また、それぞれを誤差で平均したものを、表 9、表 10 に示す。

表 9 各アニーラによる分類精度(%)

分類精度(%)	線形分離可能	線形分離不可能
古典	86.0	83.9
CMOS	84.3	78.2
Amplify AE	85.7	83.3
NEC VA	85.6	82.4
SQBM+	84.3	81.1

表 10 各アニーラによる TTS(秒)

TTS(秒)	線形分離可能	線形分離不可能
古典	0.0003	0.0004
CMOS	0.0663	0.0808
Amplify AE	0.7563	0.8237
NEC VA	0.0026	0.0027
SQBM+	0.8853	0.9912

図 11, 12 及び表 9 から、分類精度について、古典コンピュータによる SVM が最も高く、それに続いて Fixstars Amplify AE, NEC VA, Toshiba SQBM+, CMOS アニーリングマシンの順となった。

Fixstars Amplify AE の分類精度がとても高く、線形分離可能、不可能のどちらにおいても、古典の正解率にとても近い値となった。

CMOS アニーリングマシンによる分類精度は、線形分離可能な問題については古典と 1.7 ポイント差となっているが、線形分離不可能な問題については、5.7 ポイント差となり、他の疑似量子アニーラと比べても、正解率の落ちが顕著となった。

図 13, 14 及び表 10 から、TTS について、古典コンピュータによる SVM が最も小さく、それに続いて NEC VA, CMOS アニーリングマシン, Amplify AE, Toshiba SQBM+ の順となった。

#### 4.4 考察

Fixstars Amplify AE と NEC VA について、分類精度が他の疑似量子アニーラに比べて高くなつた。これは、これら二つの疑似量子アニーラが、局所解からの脱出機構を持っていることが原因である可能性が高い。

今回の実験では、Fixstars Amplify AE と Toshiba SQBM+ について、クライアントパラメータのタイムアウトを 1000ms に設定している。それによって、実行時間が 1 秒前後になっており、TTS も 1 秒前後になっていると考えられる。

#### 5. 関連研究

本報告では疑似量子アニーラを取り扱つたが、比較対象として、量子特性を使用する量子アニーラに興味を持っている。量子アニーラに SVM を適用した例として、D. Willsch による研究[8]がある。研究[8]では、量子アニーラである D-Wave 2000Q に SVM を適用しており、量子効果によりデータ数が少ない時、古典コンピュータによる SVM よりも高品質な解が出ると主張している。本研究では疑似量子アニーラを用いたため、量子効果は適用されない。

疑似量子アニーラによる SVM の適用を行つた研究[12]では、CMOS アニーリングマシンと、Fixstars Amplify AE を用いて、線形分離可能な問題一種類と、線形分離不可能な問題二種類に SVM を適用している。誤ラベルについては、混入無し、5%, 10%で性能評価を行つており、これらの点に違いがある。

また、研究[8]と研究[12]では評価基準に分類精度を用いている。そのため、TTS の観点で性能評価を行つてゐる点に違いがある。

## 6. おわりに

本報告では、複数の疑似量子アニーラを用いて、SVM の評価を行った。2 値分類問題における線形分離可能な問題と線形分離不可能な問題の 2 種について、誤差を含む問題に対して、分類精度と TTS の観点で性能評価を行った。

Fixstars Amplify AE や NEC VA については、古典環境にかなり近い分類精度が得られた。だが古典環境には及ばず、Toshiba SQBM+と CMOS アニーリングマシンについてはより分類精度が落ちる結果となった。

また、TTS の観点で見ると、古典環境が圧倒的に良く、次点である NEC VA と比較しても 10 倍ほどの差が出る結果となった。

今回、ハイパーパラメータチューニングにはグリッドサーチを使用したが、ハイパーパラメータが 4 変数となる疑似量子アニーラ環境においては、96 点のグリッドサーチではチューニング不足である可能性が高い。そのため、より効率よくハイパーパラメータ探索が行えるよう、自動チューニング(AT)技術[13]などを導入し、かつベイズ最適化などによる AT の検討を進めたい。

疑似量子アニーラだけでなく、D-Wave Advantage のような量子アニーラを用いた SVM による誤差を含むクラス分類の性能評価も検討したい。

また、本報告では詳しい検証を行っていないが、Fixstars Amplify AE と Toshiba SQBM+について、タイムアウトを 1000ms から 100ms などに設定すると、分類精度がほぼ変化せずに実行時間が十分の一となり、TTS が大幅に改善されることが分かっている。そのため、疑似量子アニーラにおける SVM のハイパーパラメータだけでなく、各疑似量子アニーラのクライアントパラメータを含めた AT 手法の提案が重要な今後の課題となる。

## 謝辞

本研究は、学際大規模情報基盤共同利用・共同研究拠点 (JHPCN)，および、革新的ハイパフォーマンス・コンピューティング・インフラ (HPCI) の支援による(課題番号: jh240002)。また、本研究は JSPS 科研費 JP23K11126, JP24K02945 の助成を受けたものです。

## 参考文献

- [1] V. N. Vapnik, “The Nature of Statistical Learning Theory. Springer”, New York, 1995.
- [2] CMOS アニーリングについて, <https://annealing-cloud.com/ja/about/cmos-annealing-machine.html> (参照 2025-02-04)
- [3] Fixstars Amplify Annealing Engine, <https://amplify.fixstars.com/ja/docs/amplify/v1/clients/fixstars.html> (参照 2025-02-04)
- [4] NEC Vector Annealing サービス, <https://jpn.nec.com/nec-vector-annealing-service/index.html> (参照 2025-02-04)

- [5] 量子インスペイアード最適化ソリューション SQBM+, <https://www.global.toshiba/jp/products-solutions/ai-iot/sbm.html> (参照 2025-02-04)
- [6] 金森敬文, “データサイエンスと機械学習：理論から Python による実装まで”, 東京化学同人, p. 212, 2022.
- [7] 金森敬文, “データサイエンスと機械学習：理論から Python による実装まで”, 東京化学同人, p. 176, 2022.
- [8] D. Willsch, M. Willsch, J. De Raedt, K. Michielsen, “Support Vector Machines on the D-Wave Quantum Annealer”, Computer Physics Communications, Vol. 248, 2020.
- [9] Amplify SDK - Fixstars Amplify - 量子コンピューティング クラウド, <https://amplify.fixstars.com/ja/sdk> (参照 2025-02-17)
- [10] ベクトルエンジン紹介: ラインアップ - NEC, <https://jpn.nec.com/hpc/sxauroratsubasa/specification/vector.html> (参照 2025-02-04)
- [11] Hitachi Annealing Cloud Web, <https://amplify.fixstars.com/ja/docs/amplify/v1/clients/hitachi.html> (参照 2025-02-17)
- [12] R. Fukuhara, M. Morishita, T. Katagiri, M. Kawai, T. Nagai, T. Hoshino, “Performance Evaluation of CMOS Annealing with Support Vector Machine”, Proc. of 2024 IEEE 17th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC), (2024). DOI: 10.1109/MCSoC64144.2024.00094
- [13] T. Katagiri, D. Takahashi, “Japanese Autotuning Research: Autotuning Languages and FFT”, Proc. of the IEEE, Vol. 106, Issue 11, pp. 2056 – 2067, (2018). DOI: 10.1109/JPROC.2018.2870284