

量子アニーリングによる学習データのオーバーサンプリング

身野 良寛¹ 柴田 将² 中村 暢達³

概要: 金融取引等において、機械学習を活用した不正検知システムが導入されつつあるが、不正データが少ないために、学習データの量が十分でないことが問題となっている。量子アニーリングを用いた制限ボルツマンマシンによって、不正データに相当する学習データを生成する手法は、従来のオーバーサンプリング方式よりも確からしい学習データを増量させることができ、結果、不正検知の精度を向上できる可能性がある。本報告では、Kaggleで公開されている金融取引データに対し、量子アニーリングを使ったオーバーサンプリングを適用する検証を行い、その有効性を確認した。

キーワード: 量子アニーリング, RBM (制限ボルツマンマシン), オーバーサンプリング, 機械学習, 不正検知

Over Sampling Technique Using Quantum annealing for Supervised Machine Learning

YOSHIHIRO MINO^{†1} SHO SHIBATA^{†2}
NOBUTATSU NAKAMURA^{†3}

1. はじめに

金融取引においては、ルールベースによって不正検知が行われることが一般的である。しかし、不正手段が巧妙かつ多様化し、次々と新しい手段が登場しており、ルールベースでの対応だけでは不正を十分に検知できない状況となっている。そこで、過去の取引履歴データを活用した機械学習技術を使って、不正の取引履歴に見られるパターンを学習し、より高度な不正検知を行う手法が開発されてきている[1][2]。

ところが、不正な取引は正常な取引に比べて件数が少なく、機械学習に使う学習データの蓄積が困難だという問題がある。その対策として、不正な取引データの件数を仮想的に増加させて機械学習を行う、以下のような手法やツールが開発されてきている[3]。

- ・ ランダムオーバーサンプリング

不正事象のデータをランダムに何回も選択して、学習に使う。データの内容、数値を変更しているわけではない。そのため学習モデルは過学習になる傾向がある。

- ・ 補間サンプリング

2つの不正データの平均のデータを作成し、新しい不正データとして追加する。不正データは同一の凸形状のクラスに含まれていることが前提である。SMOTE[4]という

ツールがよく知られている。

- ・ 中間層サンプリング

ニューラルネットワーク型の機械学習において、ネットワーク中間層のノードの値を補間することで、ランダムオーバーサンプリング相当の効果を得る[5]。

このようにオーバーサンプリングすることで、機械学習による不正検知の精度向上を試みているが、いまだ十分な精度であるとはいえない。その理由として、オーバーサンプリングで得られるデータは、ベースとなっているデータの分布空間の範囲内で線形的にデータを増やしているためであると考えられる。実際にデータを観察するとき、データの分布は線形ではない確率分布、例えば正規分布やボルツマン分布等に従ったデータが得られる。そのため、単純な計算でデータを増やした場合には、実際とは異なるデータの分布となり、機械学習の精度が伸びないのではないかと考えられる。

そこで我々は、与えられたデータをベースに、ボルツマン分布に従ったモデルを得る制限ボルツマンマシンを作り、それを活用してオーバーサンプリングを行うことで、不正検知の精度向上の可能性を検証したので、本稿で報告する。

¹ 株式会社日本総合研究所
The Japan Research Institute, Limited
² NEC ソリューションイノベーター株式会社
NEC Solution Innovators, Ltd.

³ 日本電気株式会社
NEC Corporation

2. 制限ボルツマンマシン (RBM)

ボルツマンマシンでは、図に示すように複数のノードが向きを持たない結合 (エッジ) によって結びついた無向グラフを考える。各ノード i は状態 x_i をもち、0 または 1 のいずれかを取るとする。それぞれのノード i にはバイアス b_i 、エッジ (ノード i とノード j 間) には重み $w_{i,j}$ が設定されているとする。

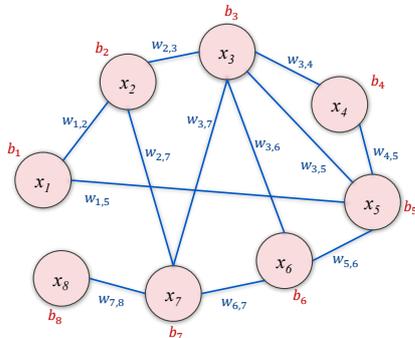


図 1 ボルツマンマシンの構成

Figure 1 The configuration of Boltzmann machine.

このグラフが持つエネルギーを下記(1)の通り定義する。ただし、 M はノード数、 ϵ はエッジの集合である。

$$\Phi(\mathbf{x}, \mathbf{b}, \mathbf{w}) = -\sum_{i=1}^M b_i x_i - \sum_{i,j \in \epsilon} w_{i,j} x_i x_j \quad (1)$$

ここで $b_i = Q_{i,i}$ 、 $w_{i,j} = Q_{i,j}$ とし、 $x_i = x_i^2$ である点に留意すると、この式はシンプルな行列形式で表現できる。

$$\begin{aligned} \Phi(\mathbf{x}, \mathbf{b}, \mathbf{w}) &= -\sum_{i=1}^M b_i x_i - \sum_{i,j \in \epsilon} w_{i,j} x_i x_j = -\sum_{i=1}^M Q_{i,i} x_i x_i - \sum_{i,j \in \epsilon} Q_{i,j} x_i x_j \\ &= -\sum_{i,j \in \epsilon'} Q_{i,j} x_i x_j \end{aligned} \quad (2)$$

それぞれのユニットがとる状態 x_i を二値の確率変数とみなしたとき、 M 個のノードがとる状態 $\mathbf{x} = [x_1, x_2, \dots, x_M]$ が、次の確率分布 (ボルツマン分布) によって与えられるものをボルツマンマシンという。 Z は正規化のための係数である。

$$p(\mathbf{x}|\mathbf{b}, \mathbf{w}) = \frac{1}{Z(\mathbf{b}, \mathbf{w})} \exp\{-\Phi(\mathbf{x}, \mathbf{b}, \mathbf{w})\} \quad (3)$$

この確率分布では、エネルギー $\Phi(\mathbf{x}, \mathbf{b}, \mathbf{w})$ が大きいほど確率が小さくなる。つまり、エネルギーが小さい \mathbf{x} の (0, 1) の組合せは発生しやすく、エネルギーが大きい \mathbf{x} の (0, 1) の組合せは発生しにくい。グラフ表示すると次図のようになる。

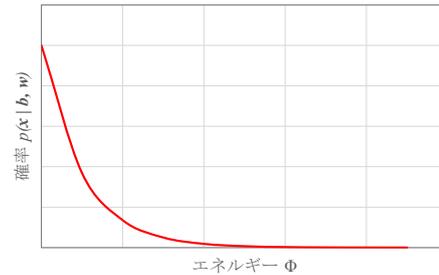


図 2 ボルツマンマシンのエネルギー分布

Figure 2 Energy probability distribution of Boltzmann machine.

式(1)はイジングモデルに変換可能な式であり、あるバイアス b_i および重み $w_{i,j}$ のもとでの最も出現頻度の高い x_i を求めることは、量子アニーリングにおいて、組合せ最適化の求解をすることと同じである。

次に、ボルツマンマシンの構造をニューラルネットワークのような 2 層構造とする。1 つの層を可視層と呼び、そのノードの値は外部から与えられ、もう一方の層を隠れ層と呼び、内部状態を表す。可視層どうし、隠れ層どうしのエッジは省かれている (図 3) [6]。これを制限ボルツマンマシン (RBM) と呼ぶ。RBM の特徴は、ニューラルネットワークの機械学習として、サンプルデータを可視層に入れて、最もエネルギー的に安定させるように学習させることで、バイアス b_i および重み $w_{i,j}$ を決定できることである。なぜならば、RBM はボルツマンマシンの特殊な形なので、エネルギーの計算が可能だからである。さらに、イジング式を解くことでエネルギー $\Phi(\mathbf{x}, \mathbf{b}, \mathbf{w})$ が最も安定する場合のデータセットを求める、つまり尤もらしい $\mathbf{x}, \mathbf{b}, \mathbf{w}$ をサンプリングできるという特徴がある。

これを古典コンピュータで行う場合、PCD 法やギブスサンプリングなどの近似的な計算を行わなければならない、精度に懸念がある。

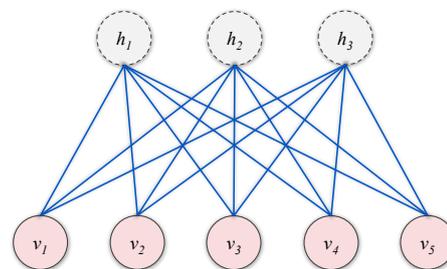


図 3 制限ボルツマンマシンの構成

Figure 3 The configuration of Restrict Boltzmann machine Configuration.

3. アニーリングマシンを用いた RBM

前節で説明した RBM の学習に量子アニーリングもしくはシミュレーテッドアニーリング (まとめてアニーリングマシンと呼ぶ) を活用することで, イジング式で表現されたエネルギーの最適化計算を高速化することが可能である. 各ノードはアニーリングマシンのスピンの相当する. ただし, アニーリングマシンのスピンはバイナリ(0/1)表記となる.

可視変数 N_v 個, 隠れ変数 N_h 個の場合, 単純に合計 $N_v + N_h$ 個のスピンの使用する. ハミルトニアン(制約式)に出てくる項は, 各スピンのバイアス (1 次) の項($N_v + N_h$ 個) とエッジ数 (可視-隠れ変数のペア数) の $N_v N_h$ 個の合計, $N_v + N_h + N_v N_h$ 個になる. 隠れ変数の数が多いほど複雑な内部状態を表現できるようになり, 表現力が増す反面, 計算の規模が大きくなり, 処理時間は増大する.

全体の処理流れは図 4 の通りである. 式(2)のように表現したときの行列 $Q_{i,j}$ を QUBO と呼ぶ. QUBO の計算処理の詳細を次節に述べる.

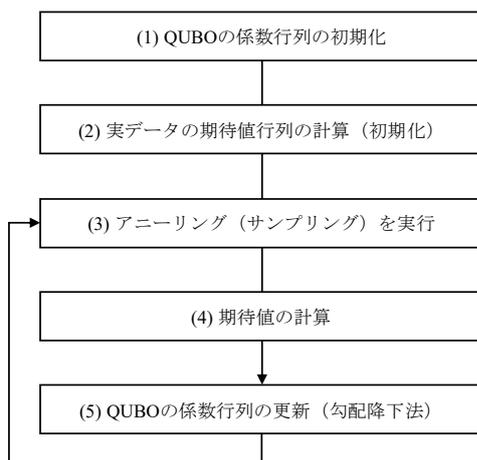


図 4 アニーリングを使った RBM の計算処理フロー
Figure 4 Calculation Process Flow of RBM with annealing

3.1 QUBO の係数行列の初期化

QUBO は可視変数 N_v 個と隠れ変数 N_h 個からなる $(N_v + N_h) \times (N_v + N_h)$ 行列である. 係数行列については, 対角成分のうち可視変数部分は実データから計算する. 頻度確率があらかじめ設定した閾値 ε より小さければ $\log(1 - \varepsilon)/\varepsilon$, 大きければ $\log \varepsilon/(1 - \varepsilon)$ とする. 対角成分のうち隠れ変数部分は正規分布からとった乱数で初期化する. 非対角成分のうち, 可視~隠れ変数間は正規分布に従う乱数で初期化し, 可視変数間, 隠れ変数間の成分は使わない.

Q	v_1	v_2	\dots	v_{N_v}	h_1	h_2	\dots	h_{N_h}
v_1	2.0	0.1	\dots	-0.1	0.2	-0.5	\dots	0.1
v_2	0.1	1.5	\dots	0.1	0.3	-0.2	\dots	-0.4
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\vdots	\dots	\vdots
v_{N_v}	-0.1	0.1	\dots	-2.1	0.1	0.2	\dots	0.3
h_1	0.2	0.3	\dots	0.1	0.2	\dots	\dots	\vdots
h_2	-0.5	-0.1	\dots	0.2	-0.3	\dots	\dots	\vdots
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\dots	\dots	\vdots
h_{N_h}	0.1	-0.4	\dots	0.3	\dots	\dots	\dots	0.05

※値はイメージです

図 5 QUBO の係数行列の初期化
Figure 5 Initialization of coefficient QUBO.

3.2 実データの期待値行列の計算 (初期化)

期待値行列については, 対角成分のうち可視変数は実データ (実際に観測可能な学習データ) の平均, 隠れ変数は 0 とする. 非対角成分は 0 とする.

$$\begin{aligned}
 \text{expectation}_{data}(i, i) &= \langle v_i \rangle_{data} = \frac{1}{N_{data}} \sum_{n=1}^{N_{data}} v_i^{(n)} \\
 \text{expectation}_{data}(j, j) &= \langle h_j \rangle_{data} = 0 \\
 \text{expectation}_{data}(i, j) &= \langle v_i h_j \rangle_{data} = 0
 \end{aligned} \tag{4}$$

E	v_1	v_2	\dots	v_{N_v}	h_1	h_2	\dots	h_{N_h}
v_1	1.0	0	\dots	0	0	0	\dots	0
v_2	0	0.5	\dots	0	0	0	\dots	0
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\vdots	\dots	\vdots
v_{N_v}	0	0	\dots	0.75	0	0	\dots	0
h_1	0	0	\dots	0	0	\dots	\dots	\vdots
h_2	0	0	\dots	0	0	0	\dots	\vdots
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\dots	\dots	\vdots
h_{N_h}	0	0	\dots	0	\dots	\dots	\dots	0

図 6 期待値行列の初期化
Figure 6 Initialization of expectation matrix.

3.3 アニーリングを実行 (サンプリング)

下記エネルギー関数をハミルトニアンとして, アニーリングを実行する.

$$\begin{aligned}
 \Phi(v, h, a, b, w) &= - \sum_i a_i v_i - \sum_j b_j h_j - \sum_i \sum_j w_{i,j} v_i h_j \\
 &= - \sum_i Q_{i,i} v_i - \sum_j Q_{j,j} h_j - \sum_i \sum_j Q_{i,j} v_i h_j
 \end{aligned} \tag{5}$$

3.4 期待値の計算

サンプリングしたデータから, 期待値行列を計算する. 具体的には, サンプリング結果のスピン積の平均値を計算する.

$$\begin{aligned}
 expectation_{model}(i, i) &= \langle v_i \rangle_{model} = \frac{1}{N_{samples}} \sum_{n=1}^{N_{samples}} \sum_i v_i^{(n)} \\
 expectation_{model}(j, j) &= \langle h_j \rangle_{model} = \frac{1}{N_{samples}} \sum_{n=1}^{N_{samples}} \sum_j h_j^{(n)} \\
 expectation_{model}(i, j) \\
 &= \langle v_i h_j \rangle_{model} = \frac{1}{N_{samples}} \sum_{n=1}^{N_{samples}} \sum_i \sum_j v_i^{(n)} h_j^{(n)}
 \end{aligned} \tag{6}$$

次に、実データの期待値行列を更新する（隠れ変数が関係する部分）。次式で条件付き確率を計算する。

$$p^{(n)}(h_j = 1 | v, Q) = \text{sigmoid} \left(Q_{j,j} + \sum_i Q_{i,j} v_i^{(n)} \right) \tag{7}$$

これをもとに、対角成分の隠れ変数部分を次式で更新する。

$$\begin{aligned}
 expectation_{data}(j, j) &= \langle h_j \rangle_{data} \\
 &= \frac{1}{N_{samples}} \sum_{n=1}^{N_{samples}} p^{(n)}(h_j = 1 | v, Q)
 \end{aligned} \tag{8}$$

また、可視変数－隠れ変数間の相互作用の係数を次式で更新する。

$$\begin{aligned}
 expectation_{data}(i, j) \\
 = \langle v_i h_j \rangle_{data} &= \frac{1}{N_{samples}} \sum_{n=1}^{N_{samples}} v_i^{(n)} p^{(n)}(h_j = 1 | v, Q)
 \end{aligned} \tag{9}$$

3.5 QUBO の更新（勾配降下法）

実データの期待値行列、サンプリングしたデータの期待値行列をもとに勾配を計算する。具体的には、実データとサンプリングを比較し、実データの方で1が出やすければ係数を小さくし、0が出やすければ係数を大きくする。

更新式（ η は学習率）は、下記の通りとなる。

$$\begin{aligned}
 Q_{i,i} &\leftarrow Q_{i,i} + \eta(\langle v_i \rangle_{model} - \langle v_i \rangle_{data}) \\
 Q_{j,j} &\leftarrow Q_{j,j} + \eta(\langle h_j \rangle_{model} - \langle h_j \rangle_{data}) \\
 Q_{i,j} &\leftarrow Q_{i,j} + \eta(\langle v_i h_j \rangle_{model} - \langle v_i h_j \rangle_{data})
 \end{aligned} \tag{10}$$

4. 不正検知への適用

本報告ではアニーリングマシンを使った RBM（以下、QRBM と呼ぶ）を活用してオーバーサンプリングを行い、それを学習データとすることで不正検知モデルの構築を行う。

不正検知を行う具体的な金融取引としてクレジットカード利用を想定し、Kaggle の Credit Card Fraud Detection[7] にて公開されているデータを用いた。このデータは取引時刻、取引金額、不正取引かどうかを示すフラグ、及び秘匿化された（何を意味するのかが不明な）数値データからなる。正常取引が約 18 万件あるのに対し、不正取引が 320 件しか存在しない非常に偏った分布となっている。

4.1 アニーリングマシンによる RBM(QRBM)の学習

QRBM を構築し、そこからサンプリングを行うことで、実在しないが不正の可能性が高い取引データを生成できることを確かめる。比較対象として、古典コンピュータによる RBM[8]（以下、CRBM と呼ぶ）も構築する。

両者の QUBO 係数行列の更新回数、及び学習データと RBM からサンプリングされたデータとの KL ダイバージェンスを観測し、アニーリングマシンの学習効率を評価する。RBM の実装と不正取引のサンプリングには以下のツールを利用した。

- シミュレーテッドアニーリング実装
D-Wave neal の SimulatedAnnealingSampler
- 古典コンピュータによる RBM 実装
scikit-learn に含まれる BernoulliRBM

アニーリングマシンを使って学習を行う際に、以下の点に注意する必要がある。

(1) 実数値の符号化

実数を 2 進数で符号化すると、その下位ビットの値が確率分布との相関がないために、RBM の学習が困難になる。そこで、実数の値域をいくつかの領域に区分けし、実数が属する領域に従って符号化を行う（図 7）。さらに、1 つのサンプルがいずれか 1 つの領域に入る制約（One-Hot 制約）を課す。

(2) 実数値のサンプリング

サンプリングの後に、アニーリングマシンの出力を実数値に復号する。このとき領域の代表値で復号すると、値の分布が歪んでしまうため、領域の範囲内で乱数を返すようにする。

(3) 正例／負例の同時学習

①正例（不正取引）データの特徴量のみを学習させる方式と、②正例／負例（正常取引）データの特徴量と目的変数を学習させる正例／負例の同時学習方式の 2 つが考えら

れる。後者の方がより良い精度が期待できるが、処理時間が長くなる。

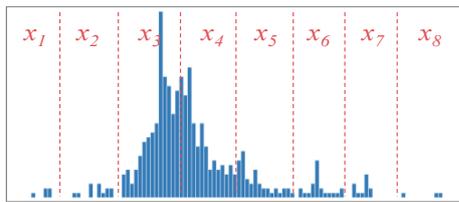


図 7 8 ビット符号化のイメージ
Figure 7 Encoding by 8 bits.

本検証では、実数値を 50 個の QUBO 変数で符号化し、正例/負例の同時学習を行った。

ただし、事前に Credit Card Fraud Detection に含まれる特徴量 28 個のうち、機械学習モデルへの寄与度が大きい 5 個のみを用いた¹。従って、可視層の QUBO 変数の数は特徴量 250 個(5x50)と目的変数 1 個を合わせた 251 個となる。隠れ層の QUBO 変数は可視層と同数の 251 個とした。

特徴量を削減した理由はアニーリングマシンで利用できる QUBO 変数の数や変数間の結合数に制約があるためである。それでも D-wave 実機で QRBM を動作させることはできないため、本報告ではシミュレーテッドアニーリングを使用する。

4.2 RBM を活用したオーバーサンプリング

不正取引の識別を行う機械学習モデルには、LightGBM[9]を使用した。学習データの内訳は、正例(不正)データ 391 件、負例(正常)データ 184,920 件であり、正例データを N 件オーバーサンプリングした 391+N 件の正例データを使って LightGBM の学習を行った。このうち 65%を学習データ、35%を評価データとした。

正例・負例の件数比が 1/500 のときに、表 1 の範囲で LightGBM の主要パラメータの探索を行い、AUC が最大となる値(太字で表示)を採用した。

表 1 LightGBM のパラメータ

Table 1 LightGBM parameters

パラメータ名	探索値
num_leaves	15, 30 , 45, 60
min_child_samples	50, 100, 200, 500
max_depth	5, 6 , 7, 8

各オーバーサンプリング手法に対して N の値を変化させて、正例・負例の件数比が 1/100~1/1 となる範囲で、次の指標で評価を行った。

- Precision / Recall
- 抽出件数 : LightGBM の予測確率(predict probability)の高いサンプルから順に取り出したとき、正例データの 50%/75%/90%/95%/100%が検知されるまでに取り出された累積件数

5. 結果

5.1 RBM の学習

QRBM と CRBM について、学習の epoch 数と KL ダイバージェンスの関係を図 8 に示す。QRBM の方が少ない epoch 数で KL ダイバージェンスが減少しており、学習が速く進んでいる。しかも、QRBM の方が KL ダイバージェンスの絶対値が小さくなった。

QRBM の計算時間は長くなった(図 9)が、本検証ではシミュレーテッドアニーリングを使用せざるを得なかったためと考えられる。今後、D-wave 等の量子アニーリング実機を利用できるようになれば改善される想定である。

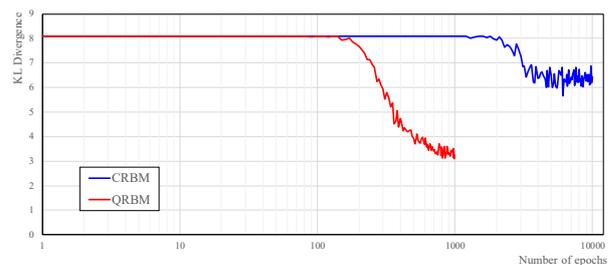


図 8 KL ダイバージェンス
Figure 8 KL divergences.

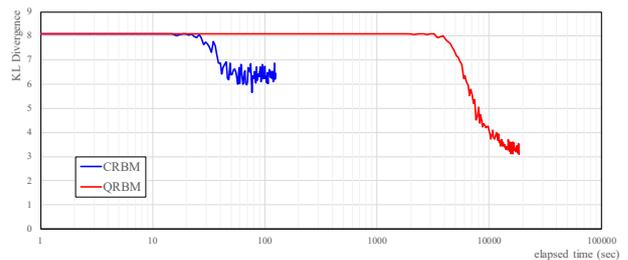


図 9 学習時間と KL ダイバージェンスの関係
Figure 9 Relationship between learning time and KL divergence.

5.2 オーバーサンプリングの評価結果

オーバーサンプリングの品質を評価するために、各手法を使って増加させた正例データを学習して、不正取引識別モデルを構築し、その精度を比較する。

比較するオーバーサンプリング手法と、以下のグラフに

¹ オーバーサンプリングせずに LightGBM によって不正検知の機械学習モデルを事前に構築し、特徴量の寄与度を算出した。

おける凡例は表 2 の通りである。

表 2 オーバーサンプリング手法とグラフ凡例

Table 2 Oversampling method and graph legend.

グラフ凡例	オーバーサンプリング手法
	QRBM
	CRBM
	ランダムオーバーサンプリング
	SMOTE

(1) Precision/Recall

全体の傾向として、正例・負例の件数比が上昇するにつれ、QRBM, CRBM, SMOTE 各手法は Precision が低下し、Recall が上昇する。一方で、ランダムオーバーサンプリングはその傾向が弱い。手法別にみると、QRBM と SMOTE は Recall が高く、Precision が低い。この中では SMOTE が最も Recall が高い。

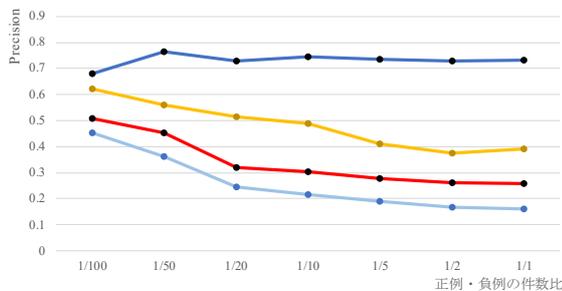


図 10 Precision の結果
Figure 10 Precision Result.

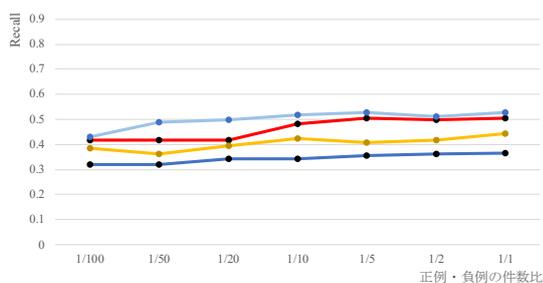


図 11 Recall の算出結果
Figure 11 Recall Result.

学習データにおける正例データ件数を増やすために、学習データと評価データの比率をそれぞれ 80%, 20% に変えたデータにおいても検証を行ったところ、図 12 のように正例・負例の件数比が小さいときの Recall にて QRBM が優位になった。

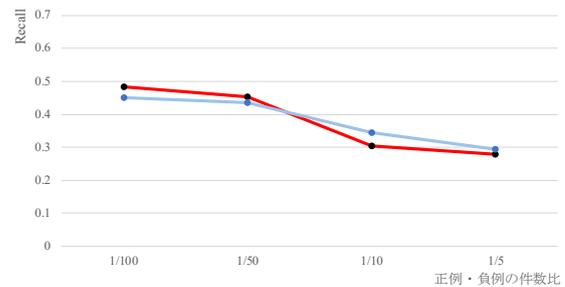


図 12 Recall の算出結果
Figure 12 Recall Result.

(2) 抽出件数

不正取引を 50%~100%抽出するまでに必要となる件数を図 13 にまとめた。CRBM の件数がやや少なく、次いで QRBM とランダムオーバーサンプリングが同程度の件数になっている。

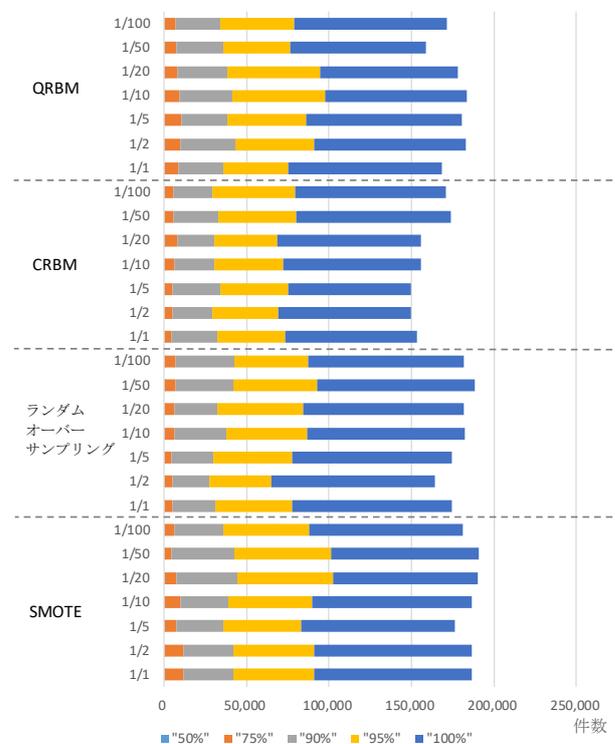


図 13 不正検知に必要な抽出件数
Figure 13 Required count for fraud detection.

6. 考察

5章で示したように、RBM は従来のランダムオーバーサンプリングや SMOTE と同様に、オーバーサンプリングに活用することができる。さらに、RBM の中でも QRBM は CRBM よりも学習効率が良好であることから、アンニリングマシンの活用用途としても適している。

金融における不正取引検出等の用途を想定する場合、

“多少の誤検知を許容しても不正は見逃さない”ことが重視され、偽陽性よりも偽陰性に対してペナルティを課す評価が行われる[10]。そこで本報告においても、Precision より Recall を重視した評価を行うこととすると、学習データ中の正例件数によって変動するものの、QRBM は SMOTE に迫る Recall を記録し、上回る場合もある。この結果より、量子アニーリングのハードウェアの発展と共に、QRBM の品質面での優位性が備わっていくことが期待できる。

一方で、“不正を 90%抽出するまでには何件の取引を調査する必要があるか”という観点で抽出件数を考慮した評価を行えば、QRBM の方が SMOTE よりも件数が少なく優位になっている。これは QRBM が不正データの全体的な統計的特徴を表現できているためではないかと考えられる。

7. おわりに

本報告では金融取引の一種のデータを使って、QRBM を使ったオーバーサンプリングの用途を示し、従来よりも優位な結果を示す場合があることを提示した。

RBM 及びアニーリングを使ったオーバーサンプリングは少量の教師データであっても、その統計的特徴を柔軟に表現できるという特長があるため、クレジットカード取引に限らず、様々な用途で適切なサンプリングを実現できる可能性がある。さらには将来的に量子アニーリング実機を使って量子の特性を活かすことができれば、より確からしいオーバーサンプリングができることが期待される。

今後の課題は 2 点挙げられる。1 つは他のデータセットを用いて QRBM がより有用な用途を示すこと、もう 1 つは従来半導体を用いたシミュレーテッドアニーリング専用のハードウェアを用いる等の手段によって、特徴量が 5 個よりも多いデータへの適用可能性を検討することである。

参考文献

- [1] “SAS® Detection and Investigation for Banking”, <https://www.sas.com/content/dam/SAS/documents/product-collateral/fact-sheets/ja/sas-detection-investigation-banking-103791-1807jp.pdf>
- [2] 参議院常任委員会調査室・特別調査室：“金融サービスをめぐる AI 活用の経緯”，https://www.sangiin.go.jp/japanese/annai/chousa/rippou_chousa/ba cknumber/2018pdf/20181001032.pdf. 立法と調査 2018. 10 No. 405, P.40, 2018
- [3] 澤崎他：“量的不均衡データに対する学習精度改善のための文書かさ増し手法”，ARG W12 No.11, 2017 W12-2017-24, https://www.sigwi2.org/wp-content/uploads/2018/12/W12_2017_24.pdf, 2017
- [4] N. V. Chawla:“Smote: Synthetic minority over-sampling technique”, Journal Of Artificial Intelligence Research, Volume 16, pages 321-357, 2002
- [5] “NEC、ディープラーニングに必要な学習データを半減、中

- 間層で得られる特徴量を意図的に変化”，<https://it.impress.co.jp/articles/-/18399>
- [6] Steven H. Adachi, Maxwell P. Henderson: “Application of Quantum Annealing to Training of Deep Neural Networks”, arXiv:1510.06356, 2015
 - [7] “Kaggle, Credit Card Fraud Detection”, <https://www.kaggle.com/mlg-ulb/creditcardfraud/>
 - [8] Hinton, G. E., Osindero, S. and Teh, Y.: “A fast learning algorithm for deep belief nets.”, Neural Computation 18, pp.1527-1554, 2006
 - [9] “LightGBM”, <https://lightgbm.readthedocs.io/en/latest/>
 - [10] Alejandro Correa Bahnsen: “Feature engineering strategies for credit card fraud detection”, Expert Systems with Applications Volume 51, pp.134-142, June 2016