

アニーリングマシンを用いた二値制約非負値行列分解

朝岡日向子 (指導教員：工藤和恵)

1 はじめに

二値制約非負値行列分解 (Nonnegative/Binary Matrix Factorization; NBMF)[1] とは、行列を非負値行列とバイナリ行列に分解することによって、行列を低次元化して表現する手法である。画像の画素値から作成した行列にこの手法を適用すると、解となる2つの行列は画像の特徴を表すため、それらを利用して通常よりもデータ量が削減された画像解析が可能となる。

また、計算過程のバイナリ行列を算出する箇所、富士通株式会社のデジタルアニーラというアニーリングマシンを利用する。デジタルアニーラは、量子現象を用いた計算方法に着想を得て設計したデジタル回路で、アニーリング方式を用いて組合せ最適化問題を高速に解くことができる。よって、バイナリ行列の計算を組合せ最適化問題と見なしてデジタルアニーラで計算すると、従来のマシンよりも高速な処理が期待できる。

2 モデル

モデルは先行研究 [1] で定義されている計算式を使用するが、そちらでは D-Wave 社のアニーリングマシンを利用しているため、デジタルアニーラで扱える形に適宜変換する。

はじめに、画像 m 枚の画素値から、行列 V を作成する。 V の列ベクトルがそれぞれの画像にあたり、画像 1 枚の画素数が n であるとする、 $n \times m$ の行列になる。この行列に NBMF を施すと、画像の特徴を表す $n \times K$ の非負値行列 W と、それぞれの画像がどの特徴を持っているのかを表す $K \times m$ のバイナリ行列 H に分解される。

$$V \approx WH \quad (1)$$

$$W_{ij} \geq 0, H_{ij} \in \{0, 1\} \quad (2)$$

W の列数 K は特徴の数を表し、任意の値に設定する。 H は V と同じく、列ベクトルが画像を表すが、 K 次元二値ベクトルになっているため、元の V よりもデータ量が削減されている。式 (1) を満たすように、 W と H に関して最小二乗法を交互に利用して逐次更新する。

$$W := \arg \min_{X \in \mathbb{R}^{n \times K}} \|V - XH\|_F + \alpha \|X\|_F \quad (3)$$

$$H := \arg \min_{X \in \{0, 1\}^{K \times m}} \|V - WX\|_F \quad (4)$$

ここで、 $\|\cdot\|_F$ はフロベニウス距離を表す。 W は実数行列で、射影勾配法 [2] を用いて更新する。 H は二値制約であるため、アニーリングマシンで更新式を解くことができる。式 (4) の解となる変数の数は $K \times m$ だが、アニーリングマシンで扱うことのできる変数には限りがある。そのため、 H の更新を列ごとに行うことによって、マシンに渡す変数を K まで減らす。

$$H_m = \arg \min_{\mathbf{q} \in \{0, 1\}^K} \|V_m - W\mathbf{q}\|_2 \quad (5)$$

式 (5) をアニーリングマシンで計算するのだが、デジタルアニーラは QUBO (Quadratic unconstrained binary optimization) 模型を入力として受け付けるため、QUBO 表現 (6) に変換する。

$$f(\mathbf{q}) = \sum_i a_i q_i + \sum_{i < j} b_{ij} q_i q_j \quad (6)$$

$$a_i = \sum_k w_{ki}(w_{ki} - 2v_{km}), b_{ij} = 2 \sum_l w_{li} w_{lj} \quad (7)$$

アニーリングマシンに QUBO の係数 (7) を渡すと、評価関数 (6) を最小にする $\mathbf{q} \in \{0, 1\}^K$ が、解として返ってくる。これが行列 H の各列にあたるため、式 (5) の計算を m 回行うことによって、 H の全体を更新する。求めた H を用いて W を更新するという繰り返しを W が収束するまで行い、最終的な W と H が、 V を NBMF で分解して得られる解となる。

3 結果と考察

ひらがなの画像 [3] から行列 V を作成し、それを NBMF で分解して画像の特徴を抽出した。

はじめに、「あ」の画像 20 枚から行列 V を作成した。画像 1 枚の画素数は $28 \times 28 = 784$ であるため、 V は 784×20 行列となる。この V を、特徴数 $K = 10$ に設定して分解すると、解として 784×10 行列 W と、 10×20 行列 H が得られる。以下の図 1 は、 V の 6 列目を構成する画像である。



図 1: ひらがな「あ」の画像 (行列 V の 6 列目)

式 (1) より、 W と、 H の 6 列目の積で、図 1 に近い画像を再現できる:

$$V_6 \approx WH_6 \quad (8)$$

H の 6 列目は以下のような構成になっている。

$$H_6^T = (0, 0, 0, 1, 1, 1, 1, 0, 0, 1)$$

H は、 W の特徴を持つか否かを二値で表し、 H_6 は 4、5、6、7、10 番目の要素が 1 になっている。すなわち、 W の 4、5、6、7、10 列の特徴を持つということになり、それらをプロットすると図 2 のようになっている。これらを足し合わせると、図 3 の画像になり、図 1 に近いものになっている。従って、NBMF で分解することによって、 V の特徴を正しく抽出できることが分かった。

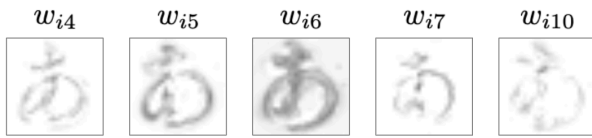


図 2: 特徴行列 W の 4、5、6、7、10 列 ($i = 1, \dots, n$)



図 3: 特徴行列 W の 4、5、6、7、10 列の合計

次に、「か」の画像 15 枚と、「き」の画像 15 枚の、合計 30 枚の画像から行列 V を作成し、NBMF で分解した。特徴数は $K = 10$ に設定したため、解として 784×10 行列 W と、 10×30 行列 H が得られる。

先ほどと同じく、 V の 6 列目を構成する画像を図 4 に示す。



図 4: ひらがな「か」の画像 (行列 V の 6 列目)

W と、 H の 6 列目の積で、図 4 に近い画像を再現できるはずなので、 H の 6 列目を参照する。

$$H_6^T = (1, 1, 0, 0, 1, 0, 0, 1, 0, 0)$$

今回は、 H_6 の 1、2、5、8 番目の要素が 1 になっているので、 W の 1、2、5、8 列の特徴を持つということになる。それらをプロットすると図 5 のようになり、足し合わせると図 6 になる。

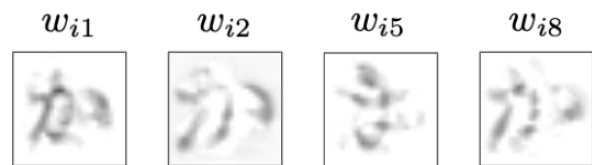


図 5: 特徴行列 W の 1、2、5、8 列



図 6: 特徴行列 W の 1、2、5、8 列の合計

W には 2 種類のひらがなの特徴が入っており、実際 w_{i5} は「き」に近い特徴が出ているが、それでも H の通りに合計すると元の画像である図 4 に近いものが再現できた。従って、2 種類の画像から V を作成した場合でも、特徴を正しく抽出できることが分かった。

このとき、同じ文字の種類の画像どうしならば似た

特徴を持ち、違う文字ならば特徴が分かれていることを、どの特徴を持っているかを示す行列 H から確認する。 H は V と同じく、各列が画像に対応しているため、画像の種類によって行列を分割する。分けた後の列ベクトルの集合の平均値、つまり特徴をどのくらい持っているかを示す平均値 $h = \sum_{i=1}^m h_{ki}/m$ を算出して、種類ごとに図 7 のグラフにプロットした。

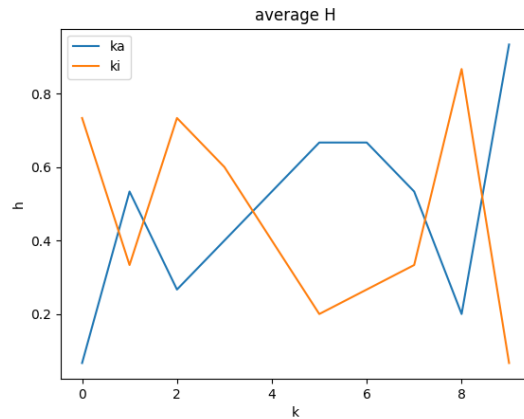


図 7: H の平均値のグラフ

図 7 を見ると、文字の種類によって特徴の平均値が正反対になっていることがわかる。よって、 H で特徴を設定する際に、画像の種類によって異なる特徴を選んでいることを確認できた。

4 まとめと今後の課題

画像行列を NBMF で分解することによって、画像が持つ特徴を正しく抽出することができた。また、その特徴を足し合わせるによって、元の画像に近いものを再現した。

画像解析にバイナリ行列の H を利用することによって、元の行列 V を使うよりも少ないデータ量で解析を行うことができた。また、 H の列ベクトルの平均値をグラフにプロットして、特徴が画像の種類によってはっきりと分かれていることを確認した。

しかし、今回は多くて 2 種類の画像を対象としたため、正しい結果を得ることができたが、種類を増やしていくと特徴が混ざって解析の精度が下がる可能性がある。 V の作成に使用するデータの増加や、アルゴリズムの改良で対策したい。また、今回実験に使用したひらがなは、形が一意に定まることが多く、解析の対象としては簡単なものであると言える。今後は動物や人の顔など、より複雑な画像を解析したい。

謝辞 本研究は部分的に未踏ターゲット事業の支援を受けている。

参考文献

- [1] D. O'Malley, V. V. Vesselinov, B. S. Alexandrov, and L. B. Alexandrov, arXiv:1704.01605 (2017).
- [2] Chih-Jen Lin, Neural Computation, **19**, 2756-2779 (2007).
- [3] NDL ラボ, <https://lab.ndl.go.jp/cms/hiragana73> (2019 年 1 月 25 日アクセス).