グラフ構造を用いた半教師あり学習における近似k近傍グラフ構築手法の提案および評価

岡鼻　雄飛†　　後藤　佑介†

† 岡山大学大学院自然科学研究科　〒 700–8530 岡山県岡山市北区津島中 3-1-1
E-mail: †okahana@mis.cs.okayama-u.ac.jp, gotoh@cs.okayama-u.ac.jp

あらまし　近年，大規模化したデータを処理するため，計算機による機械学習が大きな注目を集めている．一般的な機械学習の手法として，多数のクラスラベル付き訓練事例 (以下，ラベルデータ) を用いる教師あり学習が挙げられる．しかし，教師あり学習では多数のラベルデータが必要になり，人力でラベルを付与するコストは非常に大きい．このため，ラベルデータとクラスラベル無し訓練事例 (以下，未ラベルデータ) を併用して学習する半教師あり学習が研究されている．グラフ構造を用いた半教師あり学習では，各事例を頂点とするグラフをもとに，ラベルを伝播することで未ラベルデータにラベルを付与する．グラフ構造を用いた半教師あり学習で用いるグラフは一般的に k 近傍グラフであるが，k 近傍グラフの構築にかかる計算量は $O(dn^2)$ となる．このため，事例の数が増えると計算量は増大し，処理時間は長大化する．本研究では，グラフ構造を用いた半教師あり学習において，k 近傍グラフと比較して計算量を削減するとともに分類精度を維持する近似 k 近傍グラフの提案を行い，評価する．

キーワード　グラフ構造，半教師あり学習，近似 k 近傍グラフ

1. はじめに

近年，大規模化したデータを処理するため，計算機による機械学習が大きな注目を集めている．一般的な機械学習の手法としてニューラルネットに代表される教師あり学習が挙げられるが，クラスラベル付き訓練事例 (以下，ラベルデータ) が必要となり，人力でラベルを付与するコストは非常に大きい．一方で，クラスラベル無しの事例 (以下，未ラベルデータ) は，インターネットやセンサを介して自動で収集できる．そこで，ラベルデータと未ラベルデータを併用して学習する半教師あり学習が研究されている．

半教師あり学習は，分類器を用いる手法，生成モデルを用いる手法 [1]，Support Vecter Machine (以下，SVM) を用いる手法 [2]，およびグラフ構造を用いる手法の 4 種類に分類される．このうち，グラフ構造を用いる手法では，各事例を頂点とするグラフをもとに，ラベルが既知の頂点からラベルが未知の頂点にラベルを伝播することで，未ラベルデータにラベルを付与する．

グラフ構造を用いる手法におけるラベルの伝播方法に関する研究は多く行われている一方で，グラフの構築方法については研究が進んでいない．論文 [3] では，伝播方法だけでなくグラフ構造が大きく影響を与えることが示されている．既存研究では，グラフ構造として k 近傍グラフを用いることが一般的であるが，k 近傍グラフの構築にかかる計算量は $O(dn^2)$ である．このため，事例の数が増えると計算量は膨大になり，処理時間は長大化する．そこで本研究では，グラフ構造を用いた半教師あり学習において，グラフの構築にかかる処理時間を短縮するとともに，分類精度を維持する手法を提案する．

2. 半教師あり学習

2.1 定　　義

本節では，半教師あり学習を定義する．d ($d \geqq 1$) 次元空間上に存在する n ($n \geqq 1$) 個の事例の集合を $X = \{x_1, ..., x_n\}$，および n 個の各事例に対応するラベルの集合を $Y = \{y_1, ..., y_n\}$ とする．ここで，$y_1, ..., y_l$ ($1 \leqq l \leqq n$) は既知であり，残りの $y_{l+1}, ..., y_n$ は不明である．半教師あり学習では，ラベルが既知である $y_1, ..., y_l$ を用いて，ラベルが不明である $y_{l+1}, ..., y_n$ を予測する．

2.2 グラフ構造を用いる手法

グラフ構造を用いた半教師あり学習は，グラフ構築とラベル伝播の二段階に分解できる．グラフ構築の段階では，すべての事例を頂点とし，各事例間の距離をもとにグラフを構築する．次に，ラベル伝播の段階では，構築したグラフをもとに，ラベルデータから未ラベルデータにラベルを伝播してラベルを付与する．

2.3 SVM を用いる手法

Transductive Support Vector Machine (以下，TSVM) は，SVM を半教師あり学習に拡張した手法である．SVM では，2 種類のクラス間の距離を最大化することで分離平面を決定する．一方，TSVM では，事例が密な空間では，分離平面が存在しないという仮定で，事例が疎な空間で分離平面を作成する．通常の SVM と異なり，事例の分布が仮定を満たす場合，TSVM の性能は未ラベルデータの分布を考慮することで向上する．

2.4 EM アルゴリズムの拡張手法

EM アルゴリズムは，事例に対する対数尤度を最大化する確率分布のパラメータを反復法で求める手法である．EM アルゴリズムを半教師あり学習に拡張した手法 (以下，半教師あり

EM アルゴリズム) では，既知となる一部のラベルデータの情報を用いて，EM アルゴリズムによる学習を行う．この手法では，ラベルデータを用いることで精度が上昇する場合がある一方で，事例の分布によっては大幅に精度が低下する場合がある．

3. グラフの定義と探索方法

3.1 定　　義

本節では，グラフを定義する．d 次元空間上に存在する事例を $x_1, x_2, ..., x_n$ とし，各事例から構築したグラフを G とする．G は，各事例を頂点とするため，頂点数は n 個となる．このグラフに対する隣接行列を A とする．各頂点は，自身に対してループする辺をもたない．

二次元空間上に事例 $x_1, ..., x_5$ が存在する場合，ランダムに構築したグラフの例を図 1 に示す．また，図 1 に対応する隣接行列 A を式 (1) に示す．式 (1) の各行は，行番号に対応する事例がもつ辺を示す．例えば，式 (1) の 1 行 2 列目の値が 1 であることは，図 1 において事例 x_1 と事例 x_2 との間に辺をもつことに対応する．

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (1)$$

3.2 最近傍探索

最近傍探索では，問い合わせ元となる事例 (以下，クエリ) からもっとも距離が近い事例を探索する．もっとも単純な最近傍探索である線形探索では，クエリとすべての事例との間の距離を計算するため，探索にかかる計算量は $O(n)$ となる．このため，計算量を削減する手法として，kd-tree [4] が提案されている．

3.3 近似最近傍探索

近似最近傍探索では，クエリからもっとも近い事例を厳密に計算せず，近似的な事例を探索する．近似最近傍探索による探索結果は，最近傍探索と異なり厳密な解ではない．しかし，最近傍探索と比較して高速に探索できる．近似最近傍探索を行うための手法として，Locality Sensitive Hashing (以下，LSH) [5] がある．

3.4 最 良 探 索

本節では，作成したグラフ上で近似最近傍探索を行う方法の一つである最良探索 [6] を説明する．最良探索では，はじめにランダムに探索を開始する頂点を選択する．次に，選択した事例，および選択した事例と隣接する事例の二種類で構成される集合を作成し，クエリと作成した集合との間で，事例間の距離を計算する．このとき，クエリからの距離がもっとも短い事例が選択した事例と異なる場合，この事例を選択した事例として更新し，同じ処理を繰り返す．一方，クエリからの距離がもっとも短い事例が選択した事例と同じである場合は処理を終了し，選択した事例を探索結果とする．

隣接行列が式 (1) となる図 1 のグラフにおいて，クエリ q に対して最良探索を行う例を説明する．はじめに，ランダムに頂点を選択し，探索を開始する．今回は x_2 を選択する．次に，選択した頂点 x_2，および x_2 と隣接している事例 x_1, x_3 の合計 3 個で構成される集合を作成する．クエリ q と集合に含まれる事例 x_1, x_2, x_3 との距離をそれぞれ計算し，クエリからの距離がもっとも短い事例である x_3 を選択する．次に，x_3 および x_3 に隣接する x_2,x_4 で構成される集合を作成する．x_2, x_3, x_4 のうち，クエリとの距離がもっとも小さい事例は x_3 となり，現在選択している事例と同じとなる．この場合，探索を終了し，探索結果は x_3 となる．

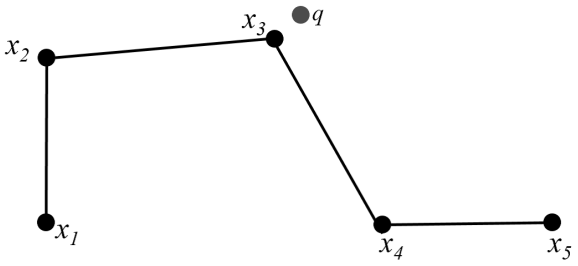

図 1　二次元空間上でランダムに構築したグラフの例

最良探索の探索結果は局所解になる可能性があるため，近似近傍探索となる．また，最良探索を用いて近似 k 近傍探索を行う場合，長さが k のヒープを用いて最良探索を行う．

4. 近傍グラフ

本章では，代表的な近傍グラフについて，k 近傍グラフ [7]，ϵ グラフ [8]，および b-matching グラフ [8] を説明する．この他の近傍グラフとして，Anchor グラフ [9]，および LSH を用いる手法 [10] が挙げられる．

4.1 k 近傍グラフ

k 近傍グラフ [7] は，各事例が自身の k 近傍となる他の事例との間に辺をもつことで作成するグラフである．このとき，事例間で作成する辺は無向辺とする．

4.1.1 k 近傍グラフの作成例

二次元空間上で事例 $x_1, ..., x_5$ が存在する場合に k 近傍グラフ ($k = 2$) を作成した例を図 2 に示す．事例間の距離はユークリッド距離で求める．k 近傍グラフでは，ある事例からの距離が短い順番に選択した k 個の事例との間にそれぞれ辺を作成する．図 2 では，x_1 からの距離が短い順番に二つの事例 x_2 および x_3 を選択し，x_1 と x_2，および x_1 と x_3 との間に辺をそれぞれ作成する．同様に，各事例に対して k 近傍となる他の事例との間に辺をもつことで，図 2 に示す k 近傍グラフを作成できる．また，k 近傍グラフで作成する辺は無向辺であるため，k 近傍グラフの頂点は k 個以上の辺をもち，正則なグラフとはならない．図 2 において，事例 x_3 の場合，x_3 の近傍は x_4 および x_5 となるため，x_3 と x_4，および x_3 と x_5 との間に辺をそれぞれ作成する．また，x_3 は二つの事例 x_1 および x_2 それぞれに対する近傍となり辺を作成するため，x_3 は合計 4 本の辺をもつ．k 近傍グラフを構築する場合の計算量は $O(dn^2)$ となる．

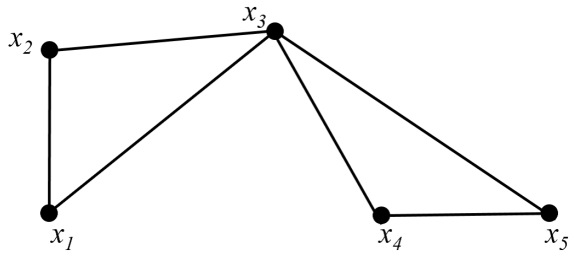

図 2　k 近傍グラフの作成例 ($k = 2$)

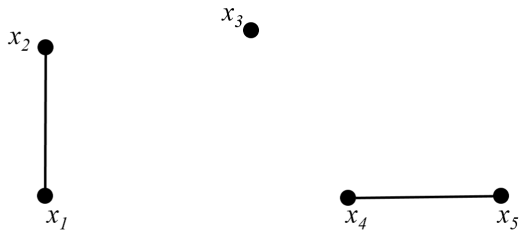

図 3　ϵ グラフの作成例

4.1.2 定　式　化

k 近傍グラフを数学的に定式化する．無向辺で作成した隣接行列を A としたとき，有向辺で作成した隣接行列を $\hat{A}$ とする．また，i 番目の事例と j 番目の事例との距離を D_{ij} とする．このとき，k 近傍グラフは以下の制約付き最小化問題となる．

$$\min_{\hat{A}} \sum_{i=0,j=0}^{n} \hat{A}_{ij} D_{ij} \tag{2}$$

制約条件は，以下の式となる．

$$\sum_{j=0}^{n} \hat{A}_{ij} = k, \hat{A}_{ii} = 0 \tag{3}$$

ここで，A は，$A_{ij} = \max(\hat{A}_{ij}, \hat{A}_{ji})$ となる．

4.2 ϵ グ ラ フ

ϵ グラフ [8] では，事例間の距離が閾値 ϵ ($\epsilon \geqq 0$) 以下の場合に辺をもつ．二次元空間上に事例 $x_1, ..., x_5$ が存在する場合に ϵ グラフ ($\epsilon = 1.0$) を作成した例を図 3 に示す．事例間の距離はユークリッド距離で求める．図 3 において，各事例間の距離が 1.0 以下となる組み合わせは，x_1 と x_2，および x_4 と x_5 となるため，x_1 と x_2，および x_4 と x_5 の間で辺をもつ ϵ グラフとなる．

図 3 から分かるように，ϵ グラフは，事例間の繋がりがない事例が増える点，および適切な ϵ の値を推定することが難しい点により，実用的でない．

4.3 b-matching グラフ

b-matching グラフ [8] は，k 近傍グラフに対称性の制約条件を追加したグラフである．半教師あり学習において，b-matching グラフは高い分類精度を実現でき [8], [11], [12]，以下の制約付き最小化問題となる．

$$\min_{A} \sum_{i=0,j=0}^{n} A_{ij} D_{ij} \tag{4}$$

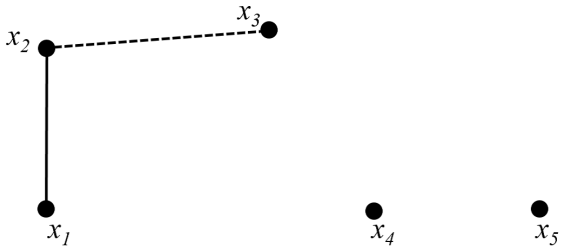

図 4　逐次添加型による近似近傍グラフの作成例

制約条件は，以下の式となる．

$$\sum_{j=0}^{n} A_{ij} = b, A_{ii} = 0, A_{ij} = A_{ji} \tag{5}$$

b-matching グラフは，k 近傍グラフと異なり，制約条件として $A_{ij} = A_{ji}$ が追加されるため，正則なグラフとなる．しかし，対称性の制約を追加するため，計算量が大きくなる．b-matching グラフの構築における計算量は $O(bn^3)$ となる．

4.4 相互 k 近傍グラフ

相互 k 近傍グラフ [3] は，k 近傍グラフと異なり，互いが k 近傍となる事例に対して辺を作成するグラフである．k 近傍グラフと同様に，式 (2), (3) に示した最適化問題を解いた後に，$A_{ij} = \min(\hat{A}_{ij}, \hat{A}_{ji})$ となる A_{ij} を求めることで作成できる．相互 k 近傍グラフは，k 近傍グラフと同じ計算量となり，k 近傍グラフと比べて精度が向上する．

5. 近似 k 近傍グラフ

4.1.1 項で説明したように，k 近傍グラフの計算量は $O(dn^2)$ となるため，大規模なデータをもとに k 近傍グラフを構築する場合，膨大な時間が必要となる．このため，k 近傍グラフの構築にかかる計算量を削減する近似 k 近傍グラフが提案されている．以下で，逐次添加型 [6]，分割統治型 [13]，および NN-Descent 法 [14] による 3 種類の近似 k 近傍グラフを順番に説明する．

5.1 逐次添加型

逐次添加型の近似 k 近傍グラフでは，逐次的に事例をグラフに追加する．このとき，作成中のグラフを用いて近似最近傍探索を行うことで，グラフの構築にかかる計算量を削減する．

逐次添加型の近似 k 近傍グラフを作成する例を図 4 に示す．逐次添加型の近似 k 近傍グラフでは，初期状態における事例として x_1 を選び，他の事例を一つずつ順番に追加する．次に，事例 x_2 を追加する．空間上に存在する事例は x_1 のみであるため，x_1 と x_2 との間に辺を作成する．さらに，事例 x_3 を追加する場合，x_1 および x_2 で構成されるグラフを用いて近似最近傍探索を行うことで，x_3 との間に辺を作成する頂点を決定する．図 4 では，クエリを x_3 とした場合，最良探索の結果は x_2 となるため，破線で示す x_2 と x_3 との間に辺を作成する．この処理を繰り返して，近似 k 近傍グラフを作成する．逐次添加型の近似 k 近傍グラフの作成にかかる計算量は，ほぼ線形となる．

5.2 分割統治型

分割統治型の近似 k 近傍グラフ [13] では，初期状態の空間を

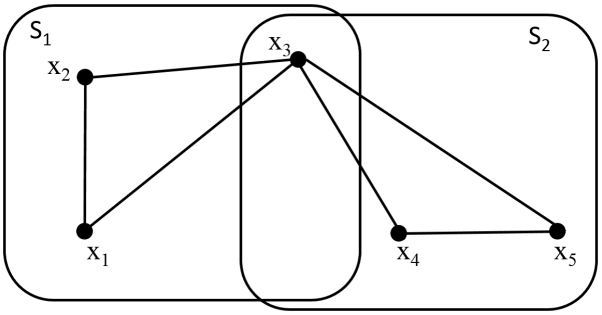

図 5　分割統治型による近似近傍グラフの作成例

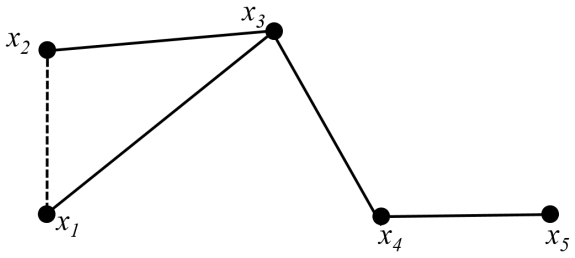

図 6　NN-Descent 法による近似近傍グラフの作成例

もとに重なりを許容する複数の部分空間に分割し，部分空間ごとに k 近傍グラフを作成することで，グラフの構築にかかる計算量を削減する．

分割統治型の近似 k 近傍グラフを作成する例を図 5 に示す．図 5 では，初期状態の空間を二つの部分空間である S_1 と S_2 に分割する．次に，部分空間 S_1 および S_2 それぞれで k 近傍グラフを作成することで，空間全体に対する近似 k 近傍グラフを作成する．分割統治型の近似 k 近傍グラフの作成にかかる計算量は $O(dn^t)(1 \leqq t \leqq 2)$ である．

5.3　NN-Descent 法

NN-Descent 法 [14] では，はじめに，事例間でランダムに辺を追加したグラフ (以下，ランダムグラフ) を作成する．ランダムグラフの更新では，更新の対象となる頂点は，自身から隣接する頂点に隣接する頂点までの範囲に限定して探索することで，計算量を削減する．

NN-Descent 法で近似 k 近傍グラフを作成する例を図 6 に示す．はじめに，図 6 において実線で示す辺を引きランダムグラフを作成する．次に，ランダムグラフの更新について，事例 x_1 に隣接する頂点となる事例 x_3 に隣接する頂点となる事例は，x_2 および x_4 である．このとき，x_1 と x_2，および x_1 と x_4 との間の距離をそれぞれ計算する．x_1 と x_2 との間の距離は x_1 と x_3 との間の距離より短いため，x_1 と x_3 との間の辺を削除し，破線で示す x_1 と x_2 との間で実線の辺を引き，グラフを更新する．辺の更新が無くなるまでこの処理を繰り返し行い，近似 k 近傍グラフを作成する．このとき，NN-Descent 法による近似 k 近傍グラフの作成にかかる計算量は $O(dn^{1.14})$ である．

6.　ラベル伝播手法

グラフ構造を用いた半教師あり学習において，ラベルを伝播する手法は数多く提案されている．ここでは，Local and Global Consistency [15](以下，LGC) を説明する．LGC では，3.1 節で述べた隣接行列を用いて，未ラベルデータにラベルを付与する．

はじめに，ラベルの伝播に用いる行列 D および S を計算する．D は式 (6) で，S は式 (7) でそれぞれ表される行列である．

$$D_{ii} = \sum_j A_{ij} \tag{6}$$

$$S = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{7}$$

このとき，ラベルの予測値を行列 F とする．行列 F の初期値 $F(0)$ は，$F(0) = Y$ とする．ラベルを予測する計算式を以下に示す．

$$F(t+1) = \alpha S F(t) + (1-\alpha) Y \tag{8}$$

式 (8) が収束するまで反復して，ラベルの伝播は終了する．論文 [15] では，$F(t)$ が最終的に式 (9) に収束することを証明している．

$$\lim_{t \to \infty} F(t) = (1-\alpha)(I - \alpha S)^{-1} Y \tag{9}$$

また，LGC 以外の手法として，Label Propagation [16] [17] がある．

7.　提 案 手 法

7.1　概　　要

グラフ構造を用いた半教師あり学習において，逐次添加型の近似 k 近傍グラフおよび NN-Descent 法を組み合わせて計算量を削減する手法を提案する．提案手法では，NN-Descent 法において初期状態で用いるランダムグラフの代わりに逐次添加型の近似 k 近傍グラフを用いる．これにより，k 近傍グラフを構築する場合に比べて計算量を削減するとともに，k 近傍グラフと同程度の分類精度を維持できる．

7.2　処 理 手 順

提案手法の処理手順は，以下の通りである．

（1） 逐次添加型の近似 k 近傍グラフを作成する．

（a） ランダムに事例を一つ選択する．

（b） 選択した事例をクエリとし，最良探索を行い辺を作成する事例を探索する．

（c） 選択した事例と探索結果の事例との間に辺を作成する．

（d） すべての事例を選択するまで (a)∼(d) を繰り返す．

（2） NN-Descent 法を使用して (1) で作成したグラフを更新する．

（a） ランダムに事例を一つ選択する．

（b） 選択した事例に隣接する頂点に隣接する頂点を探索し，選択した事例との距離を計算する．

（c） 選択した事例と自身からの辺で結ばれた事例との距離を計算する．

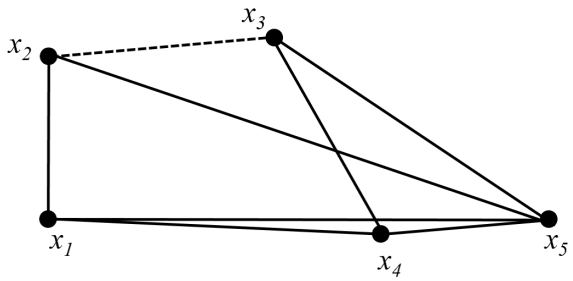

図 7　提案手法による近似 k 近傍グラフの例

（d） (b) で求めた距離について，(c) で求めた距離より短い事例が存在する場合，この事例と選択した事例との間に辺を作成し，(c) のうちもっとも距離の長い辺を削除する．

（e） 辺の更新が無くなるまで (a)∼(d) を繰り返す．

7.3 作　成　例

提案手法の作成例を図 7 に示す．図 7 では，x_1, x_2, x_5, x_4 および x_3 の順に事例を追加し，$k = 2$ とする．x_1 のみが空間上に存在する初期状態において．x_2 を追加して，x_1 と x_2 との間に辺を作成する．ここで $k = 2$ であるが，x_2 以外の事例が一つであるため，辺は一つとなる．次に，x_5 を追加する場合，x_5 以外の事例は二つであるため，x_1 と x_5，および x_2 と x_5 との間に辺を作成する．x_4 を追加する場合，x_4 をクエリとし，空間上に存在する x_1, x_2, x_5 からなるグラフを用いて最良探索を行うことで，辺を作成する事例を決定する．この場合，最良探索の結果は x_1, x_5 となるため，x_4 と x_1，および x_4 と x_5 との間に辺をそれぞれ作成する．x_3 についても同様に，x_3 と x_4 および x_3 と x_5 との間に辺をそれぞれ作成して，逐次添加型の近似 k 近傍グラフを作成した後に，NN-Decent 法を用いてグラフを更新する．

x_2 に注目する．x_2 に隣接する頂点は x_1 および x_5 であり，x_1 および x_5 に隣接する頂点はどちらも x_3 および x_4 となるため，x_2 と x_3 および x_4 と x_2 との距離をそれぞれ計算する．このとき，x_2 からの距離のもっとも短い事例は x_3 であり，x_2 と x_3 との距離は，x_2 と辺をもつ事例のうち x_2 との距離がもっとも長い x_5 との距離より短い．このため，x_2 と x_5 との間に作成した辺を削除し，x_2 と x_3 との間に破線で示す辺を作成する．更新する辺が無くなるまでこの処理を繰り返し，近似 k 近傍グラフを作成する．

8. 評　　価

8.1 グラフ構築にかかる処理時間

事例数に応じて近似 k 近傍グラフの作成にかかる処理時間の変化について，結果を図 8 に示す．今回の評価では，空間上に存在する事例数は 100, 1000, 10000, 100000 の 4 種類，特徴空間は 128 次元とする．横軸は空間上に存在する事例数，縦軸は近傍グラフの作成にかかる処理時間である．評価では，提案手法，逐次添加型の近似 k 近傍グラフ，および NN-Descent 法で作成した近似 k 近傍グラフの 3 種類の処理時間に加えて，k 近傍グラフの作成にかかる処理時間を示して比較する．

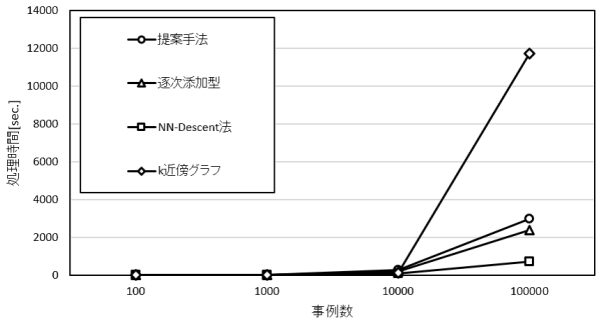

図 8　空間上の事例数と処理時間

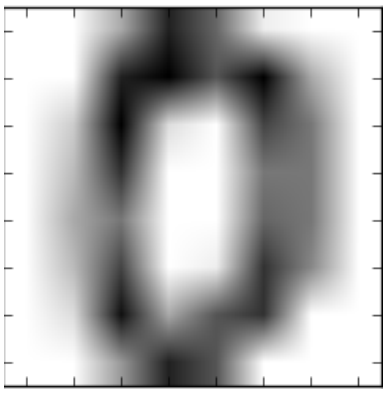

図 9　digit データの画像例

図 8 より，k 近傍グラフでは，事例数が増加すると処理時間は長大化することが分かる．一方，提案手法を用いた近似 k 近傍グラフ，逐次添加型の近似 k 近傍グラフ，および NN-Descent 法で作成した近似 k 近傍グラフでは，空間上の事例数が増えた場合，k 近傍グラフと比較して処理時間は短くなる．空間上の事例が 100000 の場合，提案手法における近似 k 近傍グラフの作成にかかる処理時間は約 2985.42 秒，k 近傍グラフの作成における処理時間は約 11712.29 秒となり，処理時間を約 74.5%短縮する．

8.2 digit データセットを用いた分類精度

digit データセットを用いた分類における精度を表 1 に示す．digit データセットは，0 から 9 までの数字を手書きしたデータの集合であり，データの大きさは 8×8 ピクセル，画像は 1797 枚である．digit データセットの画像例を図 9 に示す．クラスラベルは 0 から 9 の 10 種類で構成され，このうち一つの数字を画像としてデータセットに格納する．図 9 の場合，クラスラベルは 0 である．表 1 では，digit データ全体のうち 5%をラベルデータ，残り 95%を未ラベルデータとした．評価では，提案手法，k 近傍グラフ，逐次添加型の近似 k 近傍グラフ，および SVM の 4 種類について，分類精度を比較した．

表 1 より，提案手法における分類精度は k 近傍グラフの次に高い．また，近似 k 近傍グラフと比較して，提案手法の分類精度は大幅に向上した．以上より，提案手法は高い精度で k 近傍グラフを近似できる．

8.3 MNIST を用いた分類精度

Mixed National Institute of Standards and Technology database (以下，MNIST) を用いた分類における精度を表 2

表 1　digit データを用いた分類精度

手法	分類精度
提案手法	93.1%
k 近傍グラフ	93.8%
近似 k 近傍グラフ (逐次添加型)	88.6%
SVM	88.4%

図 10　MNIST データの画像例

表 2　MNIST を用いた分類精度および処理時間

手法	分類精度	処理時間 (秒)
提案手法	87.1%	6075.75
k 近傍グラフ	87.6%	16810.91
近似 k 近傍グラフ (逐次添加型)	81.9%	5336.21
SVM	79.4%	0.851538

に示す．MNIST は，0 から 9 までの数字を手書きしたデータの集合であり，データの大きさは 28×28 ピクセル，画像は 70000 枚である．MNIST データセットの画像例を図 10 に示す．クラスラベルは 0 から 9 の 10 種類であり，このうち一つの数字を画像としてデータセットに格納する．表 2 では，データ全体のうち，0.5%をラベルデータ，残りの 95.5%を未ラベルデータとする．提案手法，k 近傍グラフ，逐次添加型の近似 k 近傍グラフ，および SVM の 4 種類について分類精度を比較した．

表 2 より，提案手法における処理時間は，k 近傍グラフと比較して約 63.8%短縮する．また，提案手法における MNIST を用いた分類精度は k 近傍グラフとの間で次に精度が高く，提案手法と k 近傍グラフの精度の差は 0.5%であることが分かる．また，SVM では，処理時間がもっとも短くなる一方で，分類精度がもっとも低い．以上より，提案手法を用いて作成する近似 k 近傍グラフは，他の手法と比べてグラフの作成にかかる処理時間を短縮するとともに，k 近傍グラフと比べて分類精度を維持できる．

9. ま と め

本研究では，グラフ構造を用いる半教師あり学習において，逐次添加型の近似 k 近傍グラフおよび NN-Descent 法を組み合わせて近似 k 近傍グラフの構築にかかる計算量を削減する手法を提案した．提案手法では，NN-Descent 法において初期状態で用いるランダムグラフの代わりに逐次添加型の近似 k 近傍グラフを用いる．これにより，既存の近似 k 近傍グラフの作成手法と比較してグラフの構築にかかる処理時間を短縮するとともに，k 近傍グラフと比較してデータの分類精度を維持する．MNIST を用いた評価において，提案手法を用いて作成する近似 k 近傍グラフは，k 近傍グラフと比較してグラフの構築にかかる処理時間を約 63.8%短縮するとともに，分類精度の低下は 0.5%であった．

今後の予定として，提案手法と TSVM [2] との比較評価を行う．

文　献

[1] D. Kingma, S. Mohamed, D. J. Rezende, M. Welling : Semi-supervised Learning with Deep Generative Models, *Proceeding of Advances in Neural Information Processing Systems*, pp.3581-3589 (2014).

[2] T. Joachims : Transductive Inference for Text Classification using Support Vector Machines, *Proceedings of 16th International Conference on Machine Learning*, pp.200-209 (1999).

[3] C. Sousa, S. Rezende, G. Batista :Influence of Graph Construction on Semi-supervised Learning, *Proceeding of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, Vol.8190, pp.160-175 (2013).

[4] J. L. Bentley: Multidimensional binary search trees used for associative searching, *Communication of the ACM*, Vol.18, Issue 9, pp.509-517 (1975).

[5] A. Gionis, P. Indyk, R. Motwani: Similarity Search in High Dimensions via Hashing, *Proceeding of 25th International Conference on Very Large Data Bases*, pp.518-529 (1999).

[6] 岩崎 雅二郎: 木構造型インデックスを用いた近似 k 最近傍グラフによる近傍検索, 情報処理学会論文誌, Vol.52, No.2, pp.817-828 (2011).

[7] M. Szummer, T. Jaakkola: Partially labeled classification with Markov random walks, *Proceeding of Advances in Neural Information Processing*, Vol.14, pp.945-952 (2002).

[8] T. Jebara, J. Wang, and S. Chang: Graph construction and b-matching for semi-supervised learning, *Proceeding of International Conference on Machine Learning*, pp.441-448 (2009).

[9] W. Liu, J. He, S. Chang : Large Graph Construction for Scalable Semi-Supervised Learning, *Proceeding of 27th International Conference on Machine Learning*, pp.679-686 (2010).

[10] Y. Zhang, K. Huang, G. Geng, C. Liu: Fast kNN Graph Construction with Locality Sensitive Hashing, *Proceeding of Machine Learning and Knowledge Discovery in Database*, Vol.8189, pp.660-674 (2013).

[11] 小嵜 耕平, 小町守, 新保 仁, 松本 裕治: ハブを作らないグラフ構築法を用いた半教師あり語義曖昧性解消, 情報処理学会研究報告, Vol.2010-NL-199, No.14, pp.1-8 (2010).

[12] 小嵜 耕平, 小町守, 新保 仁, 松本 裕治: 半教師あり語義曖昧性解消のためのグラフスパース化, 情報処理学会研究報告, Vol.2010-NL-196, No.19, pp.1-6 (2010).

[13] J. Chen, H. Fang, and Y. Saad: Fast Approximate kNN Graph Construction for High Dimensional Data via Recursive Lanczos Bisection, *Proceedings of Machine Learning Research*, Vol.10, pp.1989-2012 (2009).

[14] W. Dong, C. Moses, K. Li : Efficient k-nearest neighbor graph construction for generic similarity measures, *Proceedings of 20th international conference on World wide web*, pp.577-586 (2011).

[15] D. Zhou, O. Bousquet, T. Navin, J. Weston, and B. Scholkopf: Learning with local and global consistency, *Proceeding of Advances in Neural Information Processing Systems*, pp.321-328 (2004).

[16] X. Zhu, and Z. Ghahramani: Learning from labeled and unlabeled data with label propagation, *Proceeding of Carnegie Mellon University*, pp.1-17 (2002).

[17] X. Zhu, and Z. Ghahramani: Semi-supervised learning us-

ing Gaussian fields and harmonic functions, *Proceeding of International Conference on Machine Learning*, pp.912-919 (2003).
[18] 泉谷 暁彦, 上原 邦昭: 逆伝播を持つ有向グラフ上でのラベル伝播を用いた半教師付き学習, 情報処理学会論文誌:数理モデル化と応用, Vol.49, No.SIG4(TOM20), pp.57-65 (2008).
[19] THE MNIST DATABASE of handwritten digits: http://yann.lecun.com/exdb/mnist/