

変分自己符号化器を用いたネットワーク埋め込みと金融ネットワークへの応用

Network Embedding using Variational Autoencoder and its Application to Financial Networks

川上 雄大^{1*} 江口 浩二^{2†}
Yuta Kawakami¹ Koji Eguchi²

¹ 神戸大学 大学院 システム情報学研究科

¹ Graduate School of System Informatics, Kobe University

² 広島大学 情報科学部

² School of Informatics and Data Science, Hiroshima University

Abstract: In this paper, we present low-dimensional embedding methods for interbank transaction networks. To address one important problem: how to obtain latent representations that well capture the structural properties of a given directed network, we propose a new network embedding model, Co-Variational Autoencoder (Co-VAE). Co-VAE simultaneously learns network embedding focusing on the links going into each node and that focusing on the links coming out of each node, attempting to reproduce the original adjacency matrix. Thereby, we can learn the Co-VAE network embedding model, simultaneously capturing both the latent representations of lender patterns and those borrower patterns. Using both latent representations, we can predict interest rates of interbank transactions.

1 はじめに

現実世界には、様々な関係データが数多く存在しており、これらはノード (頂点) とリンク (辺) で構成されるネットワーク構造で表現することができる。このようなネットワークから、構造的特徴をよく表した潜在表現を獲得することがネットワーク分析の課題の一つとしてあげられる。

本稿では、自己符号化器 (Autoencoder: AE) [1] の一種である変分自己符号化器 (Variational Autoencoder: VAE) [2] に着目したネットワーク分析を試みる。自己符号化器は、出力が入力と等しくなるように、符号化器と復号化器の学習を行うモデルであり、入力データの次元数を圧縮した潜在表現を獲得することができる。この潜在表現は、データの余分な情報やノイズを取り除き、重要な情報を残した高密度な表現であるといえる。本稿では、データの特徴をより良く表現する潜在表現を獲得することを目的としており、これが実現できれば、

ノード間のリンクの有無の予測やその属性値の予測などのタスクで精度の向上が見込める。

本稿では、金融ネットワークとりわけ銀行間取引ネットワークに着目し、それに適したデータ変換として softsign 関数を用いたデータ正規化手法や、変分自己符号化器に基づく共変分自己符号化器 (Co-Variational Autoencoder: Co-VAE) ネットワーク埋め込みモデルを提案する。提案モデルの性能評価は、ハイパーパラメータの探索の後、データの再構築と欠損したリンク属性の予測という観点から行う。また、Co-VAE を用いた銀行間取引における銀行間の金利予測に関する実験も行う。

2 関連研究

この節では、提案モデルである Co-VAE に関連する自己符号化器 [1] と変分自己符号化器 [2] やネットワーク埋め込みに関して説明する。

2.1 自己符号化器

自己符号化器 (Autoencoder: AE) [1] は教師なし学習の一種であり、ニューラルネットワークを使用したデータの次元削減を行い、潜在表現 (Latent representations)

*連絡先: 神戸大学 大学院 システム情報学研究科
〒 657-8501 兵庫県神戸市灘区六甲台町 1-1
E-mail: yuta@cs25.scitec.kobe-u.ac.jp

†連絡先: 広島大学 情報科学部
〒 739-8511 東広島市鏡山 1-3-2
E-mail: eguchi@acm.org

を獲得するモデルである。自己符号化器の一番簡単なモデルは、入力層、潜在層、出力層の3層から構成される。

このモデルの目標出力は入力自身であり、入力を X とすると次のように表すことができる。潜在層 Z は、 $Z = f_\theta(X) = \phi(WX + b)$ 、出力層 X' は、 $X' = f_{\theta'}(Z) = \phi'(W'Z + b')$ となる。ここで、 W, W' は重みパラメータ、 b, b' はバイアスパラメータであり、 $\theta = (W, b)$ 、 $\theta' = (W', b')$ とする。また、関数 ϕ, ϕ' は活性化関数であり、シグモイド関数やランプ関数 (Rectified Linear Unit: ReLU) などが用いられる。入力層の表現から潜在層の表現を得ることを符号化 (encode)、潜在層の表現から出力層の表現を得ることを復号化 (decode) と呼ぶ。

モデルパラメータ θ, θ' の学習は、勾配降下法などを用いて損失関数 L を最小化することで行われる。ここで、損失関数 $L(X, X')$ は入力層の表現 X と出力層の表現 X' の誤差で定義され、しばしば二乗誤差や交差エントロピーなどが用いられる。

2.2 変分自己符号化器

変分自己符号化器 (Variational Autoencoder: VAE) [2] は、自己符号化器の一種であり、深層生成モデルである。自己符号化器と異なる点として、潜在変数 z に確率分布を仮定していることがあげられる。変分自己符号化器の目的はデータを生成する真の分布 $p_{\theta^*}(\mathbf{x})$ を求めることであるが、その過程で入力データの潜在表現を獲得するパラメータを学習することができる。

独立同分布な N 個のサンプルからなるデータセット $\mathbf{X} = \{\mathbf{x}^{(i)}\}_{i=1}^N$ を考える。このデータが、観測されない潜在変数 z を含む確率過程によって生成されるとすると、ある値 $z^{(i)}$ がある事前分布 $p_{\theta^*}(z)$ によって生成され、あるデータ $\mathbf{x}^{(i)}$ がある条件付き分布 $p_{\theta^*}(\mathbf{x}|z)$ によって生成されると考えることができる。ここで θ^* は真のパラメータである。しかし、潜在変数 $z^{(i)}$ と θ^* は既知ではない。そこで、複雑な真の事後分布 $p_{\theta^*}(z|\mathbf{x})$ を近似した符号化器 $q_\phi(z|\mathbf{x})$ を導入し、復号化器 $p_\theta(\mathbf{x}|z)$ とともにパラメータの学習を行う。

変分自己符号化器の学習は、生成器 $p_\theta(\mathbf{x})$ の対数尤度の最大化で行われる。しかし、一般的に $p_\theta(\mathbf{x})$ を直接計算することはできないため、変分下限 (variational bound) の最大化を行う。周辺尤度は $\log p_\theta(\mathbf{X}) = \sum_{i=1}^N \log p_\theta(\mathbf{x}^{(i)})$ であり、個々のデータ点 i での周辺尤度は次のようになる。

$$\log p_\theta(\mathbf{x}^{(i)}) = D_{KL}(q_\phi(z|\mathbf{x}^{(i)})||p_\theta(z|\mathbf{x}^{(i)})) + \mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}).$$

右辺の第一項は真の事後分布とその近似との KL ダイバージェンス²である。KL ダイバージェンスとは、カルバック・ライブラー情報量とも呼ばれ、2つの確率分

布の差異を測る尺度であり、非負となる。また、第二項は、データ点 i の周辺尤度の変分下限と呼ばれ、次のように表される。

$$\begin{aligned} \log p_\theta(\mathbf{x}^{(i)}) &\geq \mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}) \\ &= \mathbb{E}_{q_\phi(z|\mathbf{x}^{(i)})}[-\log q_\phi(z|\mathbf{x}^{(i)}) + \log p_\theta(\mathbf{x}^{(i)}, z)]. \end{aligned}$$

この最右辺を変形すると式 (1) のようになり、これの最大化を行うことで個々のデータ点での周辺尤度が最大となり、結果として、周辺尤度 $\log p_\theta(\mathbf{X})$ が最大化される。

$$\begin{aligned} \mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}) &= -D_{KL}(q_\phi(z|\mathbf{x}^{(i)})||p_\theta(z)) \\ &\quad + \mathbb{E}_{q_\phi(z|\mathbf{x}^{(i)})}[\log p_\theta(\mathbf{x}^{(i)}|z)]. \end{aligned} \quad (1)$$

第一項に関して計算を行うために、 $p_\theta(z)$ に正規分布 $\mathcal{N}(z; \mathbf{0}, I)$ 、 $q_\phi(z|\mathbf{x}^{(i)})$ に $\mathcal{N}(z; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^2)$ を仮定する。また、損失関数を微分可能とするために、reparametrization trick [2] を用いて、 $z^{(i,l)} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\sigma}^{(i)} \odot \boldsymbol{\epsilon}^{(l)}$ とする。ただし、 \odot はアダマール積すなわちベクトル要素ごとの積を意味し、 $\boldsymbol{\epsilon}^{(l)}$ は $\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(0, I)$ からの l 回目のサンプリングによって得ることを表す。平均 $\boldsymbol{\mu}^{(i)}$ と標準偏差 $\boldsymbol{\sigma}^{(i)}$ は符号化器によって得られる。したがって、第一項は次のようになる。

$$\begin{aligned} -D_{KL}(q_\phi(z|\mathbf{x}^{(i)})||p_\theta(z)) &= \\ &= \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2). \end{aligned} \quad (2)$$

式 (1) の第二項は、あるデータ \mathbf{x}_i が与えられたとき、 $q_\phi(z|\mathbf{x}^{(i)})$ から z をサンプリングし、その後、 $p_\theta(\mathbf{x}^{(i)}|z)$ からサンプリングして得られた $\mathbf{x}^{(i)}$ が、入力として与えた $\mathbf{x}^{(i)}$ である度合いを表している。すなわち、どれだけうまく再構築できたかを示しており、再構築誤差とも呼ばれる。 $p_\theta(\mathbf{x}^{(i)}|z)$ からサンプリングして得られた $\mathbf{x}^{(i)}$ を $\mathbf{x}'^{(i)}$ と表記したとき、再構築誤差に二乗誤差を用いると誤差の最小化、すなわち、 $-\|\mathbf{x}^{(i)} - \mathbf{x}'^{(i)}\|_F^2$ の最大化と考えることもできる。以上をまとめると式 (3) のようになる。

$$\begin{aligned} \mathcal{L}(\theta, \phi; \mathbf{x}^{(i)}) &= \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2) \\ &\quad - \|\mathbf{x}^{(i)} - \mathbf{x}'^{(i)}\|_F^2. \end{aligned} \quad (3)$$

ここで、 J は潜在表現の次元数、 $\|\cdot\|_F$ はフロベニウスノルムである。なお、行列 A の i 行 j 列の要素を a_{ij} と表記すると A のフロベニウスノルムは $\|A\|_F = \sqrt{\sum_i \sum_j a_{ij}^2}$ と定義される。

2.3 ネットワーク埋め込み

ネットワーク埋め込み (グラフ埋め込み) は、ネットワークデータのノードやリンクなどに関して分析する

² P と Q を連続確率分布とすると、 $D_{KL}(P||Q) = \int P(x) \log \frac{P(x)}{Q(x)} dx$ と定義される。

ことで何らかの洞察を得ることを目的とする。一般的に、ネットワーク埋め込みはネットワーク構造の低次元空間への写像と捉えることができ、その写像によって保存されるネットワーク構造はネットワーク埋め込み手法に依存する。ネットワーク埋め込みには様々な手法が存在しており、Goyal らは文献 [3] で因子分解ベースの手法、ランダムウォークベースの手法、深層学習ベースの手法とその他に分類している。また、それらの手法に関して一般的なデータセットを用いた性能比較を行っている。

ネットワークの埋め込み手法で考慮すべき重要な観点として、ネットワーク構造の保存があげられる。その中でも、1次近接性と2次近接性といったノード間の類似性の尺度を表す近接性という概念がある。1次近接性とは、ノード v_i とノード v_j 間の隣接性またはリンクの重みに着目するものであり、局所的な構造を表す。2次近接性とは、ノード v_i とノード v_j の近傍構造の類似性に着目するものであり、より大域的な構造を表す。2次近接性の構造を保存することで近傍構造の類似性からリンクの欠損を捉えやすくなると考えられる。そのため、リンク予測などのタスクでは2次近接性を保持した埋め込みを行うことがより重要となる。

本稿で提案する Co-VAE によるネットワーク埋め込みは、2次近接性を保持する埋め込み手法に位置づけられる。

3 提案手法

ネットワークは、リンクの向きの有無により、無向グラフと有向グラフに分類することができ、本稿の実験で対象とする銀行間取引データは重み付き有向グラフで表現することができる。この有向グラフからは、各ノードに入ってくるリンク(インリンク)に着目するか、各ノードから出ていくリンク(アウトリンク)に着目するまで2通りの隣接行列を得ることができる。本稿で提案するのは、このリンクの向きを考慮した上で、それぞれの潜在表現を同時に学習するモデルであり、共変分自己符号化器(Co-Variational Autoencoder: Co-VAE)と名付ける。この節では、無向グラフにおけるVAEに基づくネットワーク埋め込みモデルに関する説明の後、提案モデルであるCo-VAEの概要を示し、有向グラフにおけるVAEやCo-VAEに基づくネットワーク埋め込みモデルに関して述べる。

3.1 無向グラフにおけるVAEに基づくネットワーク埋め込み

ノード数が N である重み付き無向グラフのネットワークを考える。このネットワークのノード間の隣接関係、すなわちリンクの有無とリンクの重みは隣接行列で表

現することができ、これは対称行列となる。あるノード v_i の隣接表現を表すベクトル $s_i = [s_{i,0}, s_{i,1}, \dots, s_{i,n-1}]$ が入力データとなる。ただし、ノード v_i とノード v_j が隣接しない場合は、対応する成分が零となり、隣接する場合は、 v_i と v_j の間のリンクの重みをとる。この隣接表現を入力とするモデルの学習により、非線形な写像による再構築を行う。

本稿での基礎技術となる変分自己符号化器のモデルの構造を図1に示す。符号化器(encoder)は、入力層と2層の中間層、そして、2層目の中間層の出力と正規分布 $N(0, 1)$ から ϵ をサンプリングすることで得られる潜在表現を出力とする層から構成される。また、復号化器(decoder)は、潜在表現を入力とする層と2層の中間層、再構築された表現を出力とする出力層から構成される。

このモデルのパラメータ学習は、式(3)に基づいた損失関数(式(4))を最小化することで得られる。パラメータの最適化手法にはAdam[4]などを用いる。

$$loss = \frac{1}{N} \sum_{i=1}^N \left\{ p_\alpha \cdot \left(-\frac{1}{2} \right) \cdot \sum_{j=1}^J \left(1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2 \right) + (1 - p_\alpha) \cdot \|p_\beta(x^{(i)} - x'^{(i)})\|_F^2 \right\}. \quad (4)$$

第一項は、ネットワーク埋め込みで得られる潜在表現が正規分布とどれだけ近いかを表しており、第二項は、入力と出力の二乗誤差を表す。ここで、 N はデータ数、 J は潜在表現 z の次元数、 p_α は第一項と第二項を調整するハイパーパラメータであり、 $x'^{(i)}$ は学習するVAEに $x^{(i)}$ を入力したときの出力である。また、 p_β は零要素と比較して非零要素の誤差をどれだけ重要視するかの程度を表すパラメータである。

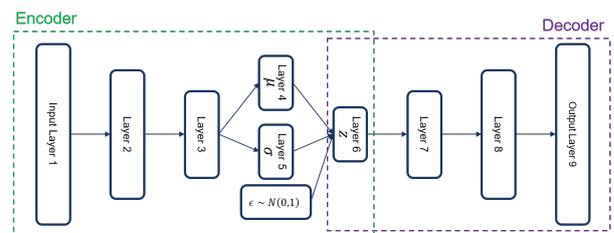


図1. VAEのモデル構造図。

3.2 有向グラフにおけるVAEに基づくネットワーク埋め込み

次に、ノード数が N である重み付き有向グラフのネットワークを考える。このネットワークでは、各ノードに入ってくるリンク(インリンク)に着目するか、各ノードから出ていくリンク(アウトリンク)に着目するまで2通りの $N \times N$ の隣接行列を得ることができる。ただし、有向グラフの隣接行列は一般に非対称行列となることに注意が必要である。すなわち、インリンクに着目した隣接行列を S とすると、アウトリンクに着目した

隣接行列は S の転置行列である S^T で表され、一般に $S \neq S^T$ となる。このうち、どちらをモデルの入力データとするか、すなわち、どちらの行列における行ベクトルに着目し、モデルを用いた次元削減を行うかによって、得られる潜在表現と解釈が異なってくる¹。このことから、本稿では変分自己符号化器 (VAE) ネットワーク埋め込みモデルとそれを拡張した共変分自己符号化器 (Co-VAE) ネットワーク埋め込みモデルを提案する。VAE ネットワーク埋め込みモデルはインリンクに着目した潜在表現を獲得するモデル、もしくはアウトリンクに着目した潜在表現を獲得するモデルであり、モデルの構造としては、無向グラフと同様に図 1 で表される。Co-VAE ネットワーク埋め込みモデルはインリンクに着目した潜在表現を獲得する符号化器とアウトリンクに着目した潜在表現を獲得する符号化器を同時に学習し、両者の潜在表現から復号化を行うモデルである。このモデルの構造は、図 2 で表される。VAE ネットワーク埋め込みモデルと Co-VAE ネットワーク埋め込みモデルは、ともに式 (4) を最小化することでパラメータを学習する。

3.3 時系列を考慮した学習

ネットワークが時系列的に変化するとき、その変化を捉えつつネットワーク埋め込みを行うことが重要となる。そこで、前述の VAE と Co-VAE に基づいたネットワーク埋め込みモデルにおける時系列を考慮した学習として次の 3 種類を提案する。

1 つ目が、モデルパラメータ継承法である。これは、ある時刻 t で学習したモデルの重みパラメータのすべてを時刻 $t+1$ のモデルの重みパラメータの初期値として継承し、学習を行う方法である。すなわち、ネットワークの時系列的な変化の大きさに応じてパラメータ更新を行う手法であると解釈できる。ある時刻 t と時刻 $t+1$ でネットワークに変化がないときは時刻 t の重みパラメータを維持し、ネットワークの変化が大きいときは時刻 t の重みパラメータの影響を小さくしつつ、時刻 $t+1$ のデータから得られる特性を強く反映するパラメータ更新を行う。この手法では、損失関数に変更はなく、式 (4) の最小化によりパラメータを学習する。

2 つ目が、深層部パラメータ制約法である。これは、前節まで述べた VAE ネットワーク埋め込みモデルおよび Co-VAE ネットワーク埋め込みモデルの符号化器の 2 つ目の中間層から μ, σ の層へと順方向に伝播することを表す重みパラメータを前の時刻のパラメータと近くなるように制約を課して学習を行う方法である。 μ と σ につながる重みパラメータは、モデルの構造上一番次元数が削減されている部分であるため、深層部パラメータ制約法と呼ぶことにする。この手法では、式 (4)

に、該当する重みパラメータの誤差項を追加した式 (5) を最小化することでパラメータを学習する。

$$loss = \frac{1}{N} \sum_{i=1}^N \left\{ p_\alpha \left(-\frac{1}{2} \right) \sum_{j=1}^J \left(1 + \log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2 \right) + (1 - p_\alpha) \|p_\beta(x^{(i)} - x'^{(i)})\|_F^2 + p_\gamma \|W_{d_t} - W_{d_{t-1}}\|_F^2 \right\}. \quad (5)$$

ここで、 p_γ はその影響力を調整するハイパーパラメータであり、 W_{d_t} と $W_{d_{t-1}}$ は学習中の t 時刻のモデルと、前時刻の $t-1$ の学習済みモデルの該当箇所の重みパラメータである。

3 つ目は、モデルパラメータ制約法と深層部パラメータ制約法を組み合わせる方法である。

4 実験データ

実験に用いたデータは欧州債務危機と呼ばれる時期を含む、2009 年 7 月 1 日から 2012 年 12 月 31 日におけるイタリアの銀行を主とした、欧州の銀行間取引のデータである (<https://www.e-mid.it/en/>)。このデータは、取引の契約日、取引を持ちかけられた側の銀行 (Quoter)、取引を持ちかけた側の銀行 (Agressor)、取引金額、取引の金利、取引の状態などの要素から構成されている。この期間の内、取引に関わった銀行は 153 行、取引件数は 162,075 件であった。取引金額は百万ユーロ単位で示されており、取引の状態は "Sell" と "Buy" の 2 種類がある。

取引状態 "Sell" とは、Agressor が Quoter に "Sell" という取引を持ちかけたものである。すなわち、取引状態 "Sell" では、Agressor から Quoter に取引金額分の金額が動く。取引状態 "Buy" とは、Agressor が Quoter に "Buy" という取引を持ちかけたものである。すなわち、取引状態 "Buy" では、Quoter から Agressor に取引金額分の金額が動く。この 2 種類の取引状態をもとに、一定期間で集計することで重み付き有向グラフを作成することができる。本稿では、月単位で集計した重み付き有向グラフを表す、42ヶ月分の隣接行列 (153×153) を時系列性のあるデータセットとして実験を行う。以下、時刻 t の隣接行列を S_t と表記し、 i 行 j 列の要素を $S_{t_{ij}}$ とする。ただし、 $t \in \{1, \dots, 42\}$ である。この隣接行列の要素 $S_{t_{ij}}$ は、銀行 i から銀行 j に $S_{t_{ij}}$ 分の金額が動いたことを表している。すなわち、この隣接行列の行ベクトルは銀行 i が他の銀行にお金を貸したという関係 (アウトリンク) に着目した表現である。また、この隣接行列を転置することで得られる S_t^T は行ベクトルが他の銀行からお金を借りたという関係 (インリンク) に着目した表現である。この節以降、それぞれの表現を貸し手側、借り手側と呼ぶ。

データ S_t は、23409 (= 153×153) の要素で構成されている。このデータセットは非常にスパースであり、要

¹無向グラフでは隣接行列は対称行列となるため、 $S = S^T$ となり、どちらの行列からでも得られる潜在表現と解釈は同じである。

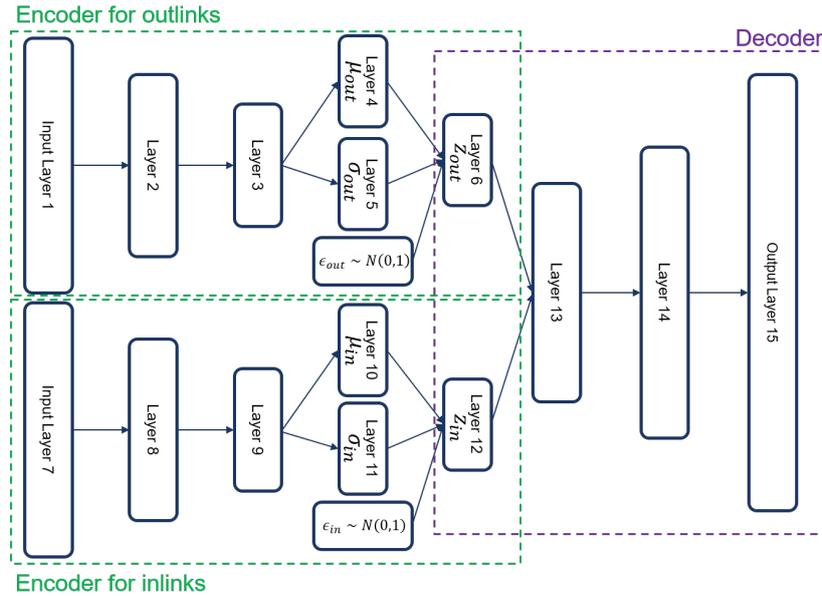


図 2. Co-VAE のモデル構造図.

素の値の度数分布をヒストグラムで表すと 0 に偏った分布となる。また、非零要素の平均値はおよそ 95.7, 中央値は 25.0 となる。銀行間の取引時に決定される金利についても同様に隣接行列が作成できる。ただし、この隣接行列の要素の値は、総和ではなく期間内にノード間で行われた取引での金利の平均値とする。なお、この隣接行列は R_i で表す。

4.1 データセットの前処理

ニューラルネットワークを用いた機械学習を行う際には、適切に正規化などのデータ変換を学習前に行うことがしばしば求められる。代表的なデータ変換手法として、min-max 正規化などがあげられる。min-max 正規化とは、与えられたデータ X の最小値と最大値を用いて、最小値を 0, 最大値を 1 に線形変換するようなデータ変換である。これは、 $\tilde{x}_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$ により得られる。ここで、 x_{min} と x_{max} はデータ X の最小値と最大値である。

min-max 正規化には外れ値の影響を受けやすいことや、最小値や最大値が既知であることを前提としているなどの問題点が存在していた。そこで、本稿では、実データの分布を意識した非線形なデータ変換として、softsign 関数を用いた正規化を提案する。softsign は文献 [5] で、活性化関数として扱われており、tanh と同程度の性能を示すと述べられている。なお、文献 [5] では softsign 関数を活性化関数として用いたのに対し、本稿ではデータ変換として用いる。

softsign 正規化 softsign 関数は $\text{softsign}(x_i) = \frac{x_i}{1+|x_i|}$ で表される。この関数の特徴として、 $\text{softsign}(x) \in$

$(-1, 1)$ や $x_i = 1$ で $\tilde{x}_i = 0.5$ となることなどがあげられる。この特徴を活かしつつ、定数 1 をパラメータ α で置き換えることで変換後の分布を制御することが可能となる。これを式で表すと $\text{softsign}_\alpha(x_i) = \frac{x_i}{\alpha + |x_i|}$ となり、これを用いた正規化手法を softsign_α 正規化とする。様々なパラメータ α における softsign_α の関数の形状を図 3 に示す。この正規化手法の特徴としては、最大値と最小値が既知でなくてよいことや、非線形な変換であるがデータの大小関係は保持されること、分布の偏りをパラメータ α によって制御できることなどがあげられる。

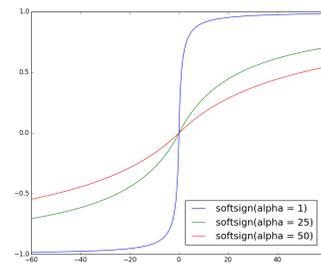


図 3. softsign 関数の波形 ($\alpha = 1, 25$).

実際に、min-max 正規化と softsign_α 正規化を適用したデータセット全体での要素の値の度数分布をヒストグラムで表して、図 4 に示す。ただし、縦軸は対数スケールとしており、ビンの数は 50 である。min-max 正規化による変換では、ヒストグラムの概形はデータ変換前の概形と変わらず、0 に偏った分布となる。また、 softsign_1 正規化は、非零要素と零要素を分割した概形となる。 softsign_{25} 正規化は、データセットの非零要素

の中央値である 25.0 で 0.5 を取るようになっており、非零要素の数がグラフにおいて左右に概ね均等に分割される。

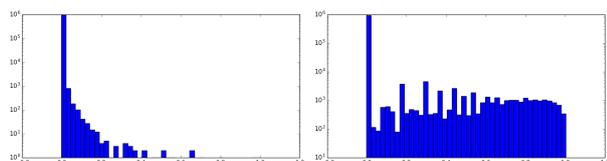


図 4. データ変換後のヒストグラム. (左)min-max. (右)softsign_25.

4.2 データセットの分割

6 節で述べる実験は検証実験とテスト実験からなり、検証実験でハイパーパラメータの探索を行い、そこで決定されたハイパーパラメータを用いてテスト実験を行う。そのため、データセットを時系列性を考慮した上で検証実験用データセットとテスト実験用データセットに分割する。検証実験に用いる学習データを、 $S_t(1 \leq t \leq 36)$ とし、検証データとしては S_{t+1} を用いる。すなわち、学習済みモデルに検証データを入力し、出力をもとに評価値を算出する。また、テスト実験に用いる学習データを、 $S_t(37 \leq t \leq 41)$ とし、テストデータとして S_{t+1} を用いる。すなわち、学習済みモデルにテストデータを入力し、出力をもとに評価値を算出する。次の節で述べるリンク予測評価は、実際に取引が行われている非零のリンクのうち、15%のリンクをランダムに隠したデータを用いて評価値を算出する。この 15%を隠したデータセットは、 S_t 毎に 10 通り用意している。

5 評価方法

この節では、提案モデルの評価方法である再構築評価とリンク予測評価や各評価指標について説明をする。

再構築評価 本稿では再構築評価として、入力データ²から符号化・復号化を経て入力データを再現した出力がどれだけランキングを保持しているか、すなわちデータ値の大きさに基づく順序関係を保持して再現されているかを測る。ランキングの一貫性を評価するために、後述のケンドール順位相関係数を用いる。変分自己符号化器の特徴として、同一の入力データであっても、符号化器の出力を計算する際に、正規分布 $\mathcal{N}(0, 1)$ から ϵ のサンプリングを行っているため、若干のズレが生じる。そのため、入力データと出力データの相関を複数回測り、その平均値をもって再構築の評価に用いる。

リンク予測評価 リンク予測評価とは、一般的に学習データに存在しなかったリンクの有無を予測するものであるが、本稿では、学習データに存在しなかったリン

クの重みを予測することを指す。そのため、4.2 節でも述べたように、予め観測されているデータの一部を隠すことで未観測のリンクを作成し、その欠損データ(欠損値を含むデータ)を学習済みのモデルに入力する。これにより、欠損データの補完がどれだけランキングを保持して行えているかを測る。再構築評価と同様に、ランキングの保持の評価には後述のケンドールの順位相関係数を用いる。未観測のリンクとして予め隠すリンクの重要度により、補完の精度に影響を及ぼすことが考えられる。したがって、予め用意した共通の欠損データを用いて、モデルの評価毎に欠損部の差異がないようにする。また、欠損部により精度が変わることが考えられるため、複数の欠損データで評価を行い、その平均をもってリンク予測評価の評価値とする。

ケンドール順位相関係数 ケンドール順位相関係数 τ (Kendall's tau) は 2 つのランキングデータの相関性を測るノンパラメトリックな評価尺度である。これは、 τ で表記され、 -1 から 1 までの範囲の値を取る。 $|\tau|$ が大きいほど強い相関があることを示す。

平均二乗誤差 平均二乗誤差 (Mean Squared Error: MSE) は、真値と予測値の誤差を測る尺度である。データ数が N 、真値が $y_i \in Y$ 、予測値が $\hat{y}_i \in \hat{Y}$ であるとき、 $MSE(Y, \hat{Y}) = \frac{1}{N} \sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2$ と定義される。

平均絶対パーセント誤差 平均絶対パーセント誤差 (Mean Absolute Percentage Error: MAPE) は真値に対する予測値の誤差をパーセンテージで表すものである。データ数が N 、真値が $y_i \in Y$ 、予測値が $\hat{y}_i \in \hat{Y}$ であるとき、 $MAPE(Y, \hat{Y}) = \frac{100}{N} \sum_{i=0}^{N-1} \left| \frac{y_i - \hat{y}_i}{y_i} \right|$ と定義される。

6 実験

本節では、提案モデルの性能を確認するための実験を行う。また、4 節で述べたように、銀行間の取引ネットワークのデータは、貸し手側に着目した隣接行列と借り手側に着目した隣接行列を獲得することができ、貸し手側の隣接行列の行ベクトルは融資状況の表現、借り手側の隣接行列の行ベクトルは借金状況の表現といえる。したがって、融資状況の表現から VAE ネットワーク埋め込みモデルを用いて融資パターンの潜在表現を獲得することを貸し手着目モデル、借金状況の表現から VAE ネットワーク埋め込みモデルを用いて借金パターンの潜在表現を獲得することを借り手着目モデル、Co-VAE ネットワーク埋め込みモデルで融資パターンと借金パターンの両者の潜在表現を同時に獲得することを貸し手借り手同時着目モデルと呼ぶこととする。

本実験では、正規化手法や出力層の活性化関数、時系列考慮手法などの組み合わせをいくつか用いた。よって、表 1 のように正規化手法や出力層の活性化関数の

²本稿における実験では、4 節で定義した隣接行列 S 、またはそれに対して 4 節に示した正規化を施した行列を指す。

組み合わせを Config_ID で略記する。また、時系列考慮手法として、モデルパラメータ継承法、深層部パラメータ制約法、両者の組み合わせの3種類で実験を行うが、それぞれ、Config_ID の後部に $_v1$, $_v2$, $_v3$ を加えることで表記する。

表 1: 各 Config ID が示す正規化手法と出力層の活性化関数の組み合わせ。

| Config IDs | 正規化手法 | 出力層の活性化関数 |
|------------|------------------------|-----------|
| Config_01 | min-max | linear |
| Config_02 | softsign ₁ | linear |
| Config_03 | softsign ₁ | sigmoid |
| Config_04 | softsign ₂₅ | sigmoid |
| Config_05 | softsign ₂₅ | relu |

各 Config_ID で、貸し手着目モデル、借り手着目モデル、貸し手借り手同時着目モデルで重みパラメータの学習を行う。このとき、各 Config_ID に対する着目モデルごとに、損失関数である式 (4) や式 (5) のハイパーパラメータの格子探索を行う。格子探索の範囲は、 $p_\alpha = [0.9, 0.99, 0.999]$, $p_\beta = [1, 10, 20, 40, 80]$, $p_\gamma = [1, 10, 100, 1000, 10000]$ とする。また、貸し手着目モデルと借り手同時着目モデルの各層の次元数は図 1 を参考にすると、Layer 番号順に (153, 100, 50, 10, 10, 10, 50, 100, 153) であり、貸し手借り手着目モデルの各層の次元数は図 2 を参考にすると、Layer 番号順に、(153, 100, 50, 10, 10, 10, 153, 100, 50, 10, 10, 10, 100, 200, 306) である。パラメータ更新は Adam を用いており、学習率は 0.001, エポック数は 1000 とし、出力層以外の活性化関数には relu を用いている。これらを通して、各 Config_ID の着目モデルごとに最適パラメータを検証データを用いて決定し、テストデータを用いて性能を比較する。

まずはじめに、表 1 の各設定で着目モデルごとに学習を行い、再構築評価とリンク予測評価の 2 つの観点から比較する。ここでは、貸し手借り手同時着目モデルの結果のみを表 2 と表 3 に示すが、他の 2 つでも同様の傾向となった。結果として、min-max 正規化よりも softsign 正規化がデータ変換として有効であることが示された。また、両評価タスクで Config_05 が良好な評価値を示すことが確認できた。さらに、評価タスクによって最適なハイパーパラメータが異なることも確認できた。

再構築評価では、データセット全体のランキングの相関を測っているため、零や非零を区別しつつ再構築を行っていることが求められる。その結果、誤差項を重要視する p_α がより大きな値をとるようになったと考えられる。一方、リンク予測評価では、隠されたリンクを再構築した結果どれだけ大小関係を捉えているかを測っている。したがって、隠されていない部分から構造的特徴を捉えた再構築を行うことが求められる。その結果、

非零要素を重要視する p_β がより大きな値をとるようになったと考えられる。これ以降は、両評価タスクで良い性能を示した Config_05 の設定でのリンク予測評価に関する結果を示す。

次に、Config_05 の設定における各着目モデルの性能比較を行い、Co-VAE ネットワーク埋め込みモデルが優れた性能を示すかを確認する。VAE ネットワーク埋め込みモデルでは捉えられていなかった情報を貸し手側と借り手側の潜在表現を同時に学習する Co-VAE ネットワーク埋め込みモデルが捉えることで、優れた性能を示すと期待される。表 4 に示した結果において、Co-VAE を用いた貸し手借り手同時着目モデルは他よりも優れた評価値を示した。

表 2: 再構築評価における貸し手借り手同時着目モデルの各設定での最適ハイパーパラメータと Kendall's tau.

| Config IDs | p_α | p_β | 学習データ | 検証データ |
|------------|------------|-----------|---------------|---------------|
| Config_01 | 0.900 | 1 | 0.182 ± 0.015 | 0.166 ± 0.015 |
| Config_02 | 0.999 | 40 | 0.279 ± 0.025 | 0.251 ± 0.024 |
| Config_03 | 0.999 | 10 | 0.283 ± 0.026 | 0.252 ± 0.024 |
| Config_04 | 0.999 | 10 | 0.282 ± 0.026 | 0.252 ± 0.024 |
| Config_05 | 0.999 | 10 | 0.588 ± 0.019 | 0.461 ± 0.027 |

表 3: リンク予測評価における貸し手借り手着目モデルの各設定での最適ハイパーパラメータと Kendall's tau.

| Config IDs | p_α | p_β | 検証データ |
|------------|------------|-----------|---------------|
| Config_01 | 0.900 | 80 | 0.278 ± 0.025 |
| Config_02 | 0.900 | 80 | 0.278 ± 0.029 |
| Config_03 | 0.990 | 80 | 0.299 ± 0.027 |
| Config_04 | 0.990 | 80 | 0.347 ± 0.034 |
| Config_05 | 0.990 | 80 | 0.335 ± 0.034 |

表 4: リンク予測評価における各着目モデルでの最適ハイパーパラメータと Kendall's tau.

| Models | p_α | p_β | 検証データ | テストデータ |
|--------|------------|-----------|---------------|----------------------|
| 貸し手着目 | 0.9 | 80 | 0.312 ± 0.045 | 0.335 ± 0.034 |
| 借り手着目 | 0.9 | 80 | 0.327 ± 0.039 | 0.342 ± 0.059 |
| 同時着目 | 0.99 | 80 | 0.335 ± 0.034 | 0.375 ± 0.026 |

次に、時系列を考慮した学習法としてモデルパラメータ継承法 (v1)、深層部パラメータ制約法 (v2)、両者の組み合わせ (v3) の比較実験を Config_05 の設定で行う。この結果を貸し手借り手着目モデルのみ表 5 に示す。深層部パラメータ制約法 (v2) が最良であった。時系列を考慮することで評価値自体はやや低下しているものの、時区間を横断した分析が可能となるのが利点である。

最後に、得られた潜在表現を用いた応用として、ニューラルネットワークによる回帰モデルを用いた金利予測の実験を行う。これは、入力層、中間層、出力層の 3 層で構成される。入力はネットワーク埋め込みによって得ら

表 5: リンク予測評価での各時系列考慮法に対する貸し手借り手同時着目モデルの最適ハイパーパラメータと Kendall's tau.

| Config IDs | p_α | p_β | p_γ | 検証データ | テストデータ |
|--------------|------------|-----------|------------|---------------|----------------------|
| Config_05_v1 | 0.999 | 40 | - | 0.290 ± 0.028 | 0.265 ± 0.032 |
| Config_05_v2 | 0.999 | 20 | 1000 | 0.325 ± 0.042 | 0.345 ± 0.025 |
| Config_05_v3 | 0.990 | 10 | 100 | 0.296 ± 0.045 | 0.322 ± 0.024 |

れた潜在表現を利用することで $X =$ [時刻 t の i 銀行の融資パターンの潜在表現, 時刻 t の j 銀行の借金パターンの潜在表現] と接続し, 目標出力を $Y = R_{t,i,j}$ とする回帰モデルの学習を行う. ここで, 活性化関数は relu, パラメータ更新には Adam を用いて学習率は 0.003 とした. また, 学習や評価に用いた $R_{t,i,j}$ は, t 月の金利の隣接行列の要素のうち非零である, すなわち取引を行っているものを対象とした. この $R_{t,i,j}$ は, i 銀行が貸し手側, j 銀行が借り手側のときに設定される金利を意味している.

VAE ネットワーク埋め込みモデルは, 貸し手着目モデルと借り手着目モデルの 2 つを用いてそれぞれの潜在表現を獲得することで X を作成する. Co-VAE ネットワーク埋め込みモデルは, 貸し手側の潜在表現と借り手側の潜在表現を同時に獲得できるため, これのみで X を作成する. このとき, X は 20 次元であり, 中間層は 5 次元とした. また, それぞれに深層部パラメータ制約法を適用したときに得られる潜在表現を用いた予測も行う. なお, これらの着目モデルのハイパーパラメータはリンク予測評価で最適となったパラメータセットを用いる. 比較対象として, softsign₂₅ 正規化を適用後の元の隣接行列を用いた予測も行う. このとき, 入力データは $X = [i$ 銀行の貸し手側の隣接表現, j 銀行の借り手側の隣接表現] の 306 次元からなり, 中間層の次元数は 18 に設定した. これらの予測性能は, MAPE で評価する.

それぞれの金利予測の結果, 元の隣接行列を用いたときは 58.82%, Config_05 の VAE を用いたときは 55.41%, Co-VAE を用いたときは 50.16%, Config_05_v2 の VAE を用いたときは 52.13%, Co-VAE を用いたときは 49.75% となった. 結果として, Co-VAE を用いることや深層部パラメータ制約法を適用することによる性能の向上が確認できた.

7 むすび

本稿では, 銀行間取引ネットワークの分析を行う手法として, 共変分自己符号化器 (Co-VAE) ネットワーク埋め込みモデルを提案した. これは, ノード間のインリンクとアウトリンクに着目するモデルであり, 実験ではインリンクに着目した借金パターンの潜在表現とアウ

トリンクに着目した融資パターンの潜在表現を同時に学習した. また, softsign _{α} 正規化や時系列を考慮した学習法も提案した.

実験の結果, softsign _{α} 正規化はモデルの性能向上に大きく貢献したといえる. また, 応用上重要とされるリンク予測評価においては提案モデルである Co-VAE ネットワーク埋め込みモデルの有効性と時系列を考慮した学習として深層部パラメータ制約法が有効であることが確認できた. 金利予測に関する実験でも良好な結果を示した.

今後の課題としては, より詳細なハイパーパラメータ探索をとまなう精緻な分析・評価があげられる. 本稿では, Co-VAE の復号化器における中間層の次元数を VAE ネットワーク埋め込みモデルの 2 倍と設定したが, 適切な次元数を選択することで, さらなる精度の向上が見込める.

謝辞

本研究を行うにあたり, 有益な助言を頂いた神戸大学大学院経済学研究科の羽森茂之教授と金京拓司教授に感謝する. 本研究に使用したデータを提供して頂いた神戸大学大学院システム情報学研究科谷口隆晴准教授に感謝する. 本研究の一部は科学研究費補助金基盤研究 (B) (15H02703) の援助による.

参考文献

- [1] Geoffrey Hinton and Ruslan Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, Vol.313, No.5786, pp.504-507, (2006)
- [2] Diederik P. Kingma and Max Welling, "Auto-Encoding Variational Bayes," *ICLR 2014*, (2014)
- [3] Palash Goyal and Emilio Ferrara, "Graph Embedding Techniques, Applications, and Performance: A Survey.," *CoRR*, (2017)
- [4] Diederik P. Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization," *ICLR 2015*, (2015)
- [5] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," *AISTATS'10*, (2010)