

複数入力を用いた Recurrent Neural Network に基づく時系列予測 Time-series prediction based on Recurrent Neural Network using multiple inputs

安達 凜†
Rin Adachi

六井 淳†
Jun Rokui

1. はじめに

ニューラルネットワークを用いて時系列データを予測する研究が行われている。松井らの研究では再帰型ニューラルネットワーク (Recurrent Neural Network) を用いて時系列予測を行っている。⁴ この研究では特定の株式の利益率を予測対象にしている。予測モデルは訓練データに対して高い精度を示した。一方で未知のテストデータに対しては誤差が大きく、十分な精度とは言えない。

ニューラルネットワークによる時系列予測の精度を向上させるために、単一の入力ではなく複数の入力を与える研究が行われている。南は株価に加えて企業が HP 上に公表する情報、企業の国内外の受注残高、内閣府が統計をとっている工作機械受注高を入力に用いて株価予測を行った。⁵ 結果として株価のみでの予測に対して企業情報を考慮した予測は精度が高いことが報告されている。宮崎らは日経平均株価の予測に、日経平均株価に加え主要な 30 銘柄の株価を入力として予測を行った。⁶ いずれの研究もランダムな予測に対し改善が見られたと報告している。

本研究では Recurrent Neural Network を用いた時系列予測を行う。時系列データを予測する場合、対象の過去の時系列データを入力として用いるが、それだけでは予測誤差が大きくなってしまふ。本研究では、予測対象と関連する可能性のある複数の時系列データを回帰分析を用いて 1 つにまとめ、ニューラルネットワークへ入力することで時系列予測を行う手法を提案する。

2. 提案手法

提案手法全体の模式図を図 1 に示す。システムは前処理部、回帰分析部、Recurrent Neural Network (RNN) 部、比較部に分けられる。前処理部ではデータの正規化と切り出しを行う。回帰分析部では回帰分析を用いて複数の時系列を 1 つに集約する。RNN 部は RNN を用いて予測対象の時系列予測を行う。比較部では予測結果と実測値を比較し精度を検証する。

2.1. 前処理部

本研究で利用される時系列データは複数存在する。各時系列データをそれぞれの中の最大値で除して 0~1 の範囲に正規化する。

全時系列データから予測対象時系列データ Y と入力として用いる時系列データ X を取り出す。入力である X として m 種類の時系列データを用いる。 X, Y の長さは n とする。このとき、 X および Y

は次式のようになる。

$$X = \begin{pmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_m^1 & x_m^2 & \cdots & x_m^n \end{pmatrix} \quad (1)$$

$$Y = \{y^1, y^2, \dots, y^n\} \quad (2)$$

時系列データの t 番目以降を RNN による予測範囲とする。RNN への入力時系列長を A とする。 X および Y を各時系列データの $t-1$ 番目を先頭として長さ A で切り出す。切り出されたデータ \tilde{X}, \tilde{Y} は次式のようになる。

$$\tilde{X} = \begin{pmatrix} x_1^{t-A} & x_1^{t-A-1} & \cdots & x_1^{t-1} \\ x_2^{t-A} & x_2^{t-A-1} & \cdots & x_2^{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_m^{t-A} & x_m^{t-A-1} & \cdots & x_m^{t-1} \end{pmatrix} \quad (3)$$

$$= \begin{pmatrix} \tilde{x}_1^1 & \tilde{x}_1^2 & \cdots & \tilde{x}_1^A \\ \tilde{x}_2^1 & \tilde{x}_2^2 & \cdots & \tilde{x}_2^A \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_m^1 & \tilde{x}_m^2 & \cdots & \tilde{x}_m^A \end{pmatrix} \quad (4)$$

$$\tilde{Y} = \{y^{t-A}, y^{t-A-1}, \dots, y^{t-1}\} \quad (5)$$

$$= \{\tilde{y}^1, \tilde{y}^2, \dots, \tilde{y}^A\} \quad (6)$$

2.2. 回帰分析部

回帰分析は複数の変数から対象の変数の値を予測する。³ 提案手法の回帰分析部では複数の時系列データを 1 つに集約する。これにより、予測対象の過去時系列データを生成している。回帰分析の手法として本研究では、今回、重回帰分析とファジィ回帰分析を用いる。

2.2.1. 重回帰分析

重回帰分析は重要で適用場面の多い多変量解析の手法である。³ 重回帰分析は、予測変数 \underline{Y} を、説明変数 X を用いる多項式で表現する。 \underline{Y} は予測対象 Y の推定値、 X は入力データである。入力データ X の系列数が n 個であったとき重回帰式は $X = x_1, \dots, x_n$ に対し、次式のように定義される。

$$\underline{Y} = w_0 + w_1 x_1 + \cdots + w_n x_n \quad (7)$$

このとき、 w_0 は切片であり、 w_i ($i = 1 \dots n$) は説明変数の係数である。¹

入力データ及び予測対象の各系列ごとのデータ数を m としたとき、式 (8) を最小にするように w_i ($i = 1 \dots n$) を設定することにより、重回帰式が求まる。

$$\sum_{j=1}^m (y_j - \underline{y}_j)^2 \quad (8)$$

†静岡県立大学経営情報イノベーション研究科, Graduate School of Management and Information of Innovation, University of Shizuoka

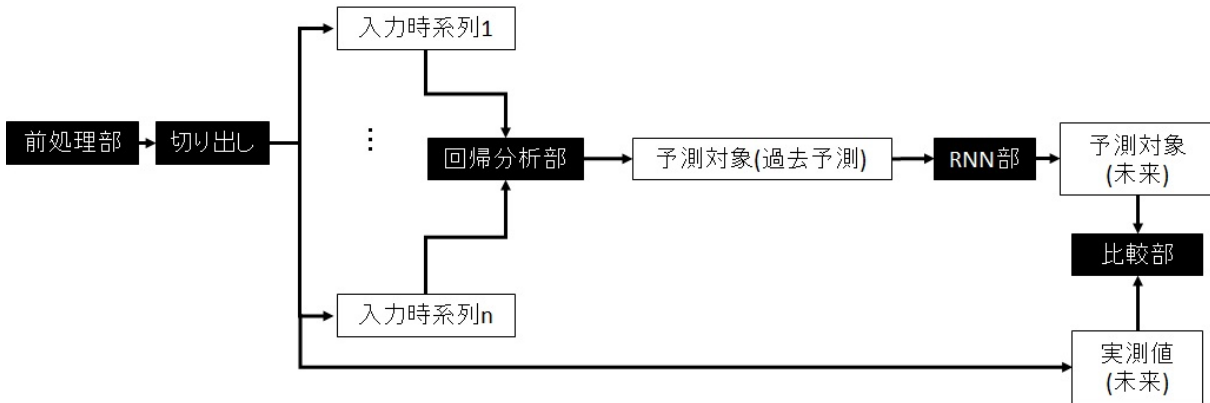


図 1: システムの模式図

2.2.2. ファジィ回帰分析

ファジィとは「あいまいさ」を扱うための理論である.¹⁴ このような「あいまいさ」を持ち, 要素が集合に属しているか否か連続的な値で表される集合をファジィ集合という.⁹ メンバシップ関数 $\mu A(x)$ は, 式 (9) で定義される.

$$\mu A(x) = \begin{cases} 0 & x \leq 15 \text{ のとき} \\ \frac{x - 15}{10} & 15 < x < 25 \text{ のとき} \\ 1 & x \geq 25 \text{ のとき} \end{cases} \quad (9)$$

このファジィ概念を利用した回帰分析の手法が田中らによるファジィ回帰分析^{11,12} である. ファジィ回帰分析において, 説明変数の係数はファジィ集合である. 最終的な予測はファジィ集合の形で出力される.

ファジィ回帰分析においては, ファジィ集合の形を定義するメンバシップ関数は単純なピラミッド型メンバシップ関数とする. これは $\mu A(x) = 1$ となる頂点 b およびその左右に等間隔で広がるあいまいさ c で規定される. ピラミッド型メンバシップ関数 $A = (b, c)$ は次式で定義される.

$$\mu A(x) = \begin{cases} 0 & \text{その他のとき} \\ \frac{1 - (|b - x|)}{c} & b - c < x < b + c \text{ のとき} \end{cases} \quad (10)$$

ファジィ集合はピラミッド型メンバシップ関数で定義されているものとする.

入力データ X が n 系列のとき, 入力データは説明変数 $x_i (i = 1, 2, \dots, n)$ とおく. 係数 A は $A_k (k = 0, 1, \dots, n)$ となり, A_0 は定数項である. 最終的な出力 Y は次式のように定義される.

$$Y = A_0 + x_1 A_1 + \dots + x_n A_n \quad (11)$$

ファジィ集合どうしの和は頂点とあいまいさそれぞれの和となる. ファジィ集合と定数の積は頂点, あいまいさそれぞれが定数との積をとる.

式 (11) により求まる Y に予測対象の実際の値 Y が含まれている可能性の度合を $h (0 \leq h \leq 1)$ とする. h が大きいと Y に Y が含まれる可能性が大きくなるが, あいまいさも大きくなる. h は利用するデータの信頼性を反映しており, データの信頼性が高いほど h は小さくなる.

ファジィ線形回帰モデルは係数 A を求めることで構築される. このとき, Y には Y が h 以上の度合いで含まれていなければならない. また, 回帰モデルのあいまいさを表す c は最小になる必要がある. 入力データ X が n 系列で各系列のデータ数が m 個, 予測対象の実際の値を $Y = y_1, \dots, Y_m$ としたとき, A は式 (12) の問題を解くことで求まる. この問題は制約条件をすべて満たす中で, $\sum_{k=0}^n c_k$ である目的関数を最大化もしくは最小化することを目的とする.

$$\left\{ \begin{array}{l} \min \sum_{k=0}^n c_k \\ \text{s.t.} \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{1j}|) + b_0 + \sum_{j=1}^n b_j |x_{1j}| \geq y_1 \\ \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{1j}|) - b_0 - \sum_{j=1}^n b_j |x_{1j}| \geq -y_1 \\ \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{2j}|) + b_0 + \sum_{j=1}^n b_j |x_{2j}| \geq y_2 \\ \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{2j}|) - b_0 - \sum_{j=1}^n b_j |x_{2j}| \geq -y_2 \\ \quad \vdots \\ \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{mj}|) + b_0 + \sum_{j=1}^n b_j |x_{mj}| \geq y_m \\ \quad (1-h)(c_0 + \sum_{j=1}^n c_j |x_{mj}|) - b_0 - \sum_{j=1}^n b_j |x_{mj}| \geq -y_m \\ c_k \geq 0 \quad (k = 0 \dots n) \end{array} \right. \quad (12)$$

2.2.3. 回帰分析部のフロー

回帰分析部の処理の流れを説明する.

1. 予測対象の時系列データ \check{Y} と入力時系列データ \check{X} を受け取る.
2. 受け取った時系列データから回帰モデルを構築する. 回帰モデルは重回帰分析またはファジィ回帰分析とする. 回帰モデルは式 (14) のように表される. このとき, \underline{Y} は予測値, a は係数である.

$$\underline{Y} = \{\underline{y}^1, \underline{y}^2, \dots, \underline{y}^A\} \quad (13)$$

$$\underline{Y} = a_0 + a_1\check{x}_1 + \dots + a_n\check{x}_n \quad (14)$$

3. 構築した回帰モデルに入力時系列データを入力する. 回帰モデルが予測結果を出力する. 出力時の計算は次のようになる.

$$\underline{y}^c = a_0 + a_1\check{x}_1^c + \dots + a_n\check{x}_n^c \quad (c = 1, \dots, A) \quad (15)$$

4. 得られた予測結果を予測対象の過去時系列として返す.

2.3.Recurrent Neural Network 部

Recurrent Neural Network 部では, 回帰分析部によりまとめられた過去時系列データから未来の予測対象の時系列データを予測する. 本研究では Recurrent Neural Network として Long Short Term Memory(LSTM) を使用する.

2.3.1.Long Short Term Memory

Long Short Term Memory(以下,LSTM とする) は,1997年に Hochreiter ら²により開発された Recurrent Neural Network(以下,RNN とする)の一種である. は Neural Network の一種である.RNN は時系列に対し高い適性を持つ.RNN の構造上の特徴として, ネットワークの隠れ層ノードの出力が次の時刻で入力される再帰的な構造をもつ. 再帰構造 W を持つ LSTM の全体構造を図 2 に示す.

LSTM は特徴的な構造として LSTM ブロックを持つ. 図 2 の隠れ層にある LSTM は LSTM ブロックを意味する.LSTM ブロックの内部構造は図 3 のようになる.

ここで, 入力時系列データ X の系列数を m とし, 時系列の長さを l とする. このとき, X は次式のようになる.

$$X = \begin{pmatrix} x_1^1 & x_1^1 & \dots & x_1^l \\ x_2^1 & x_2^2 & & \vdots \\ \vdots & & \ddots & \vdots \\ x_m^1 & \dots & \dots & x_m^l \end{pmatrix} \quad (16)$$

図 3 における f は活性化関数である. f の出力が LSTM ブロックへの入力となる. 図 2 の隠れ層における i 個目の LSTM ブロックを LSTM ブロック

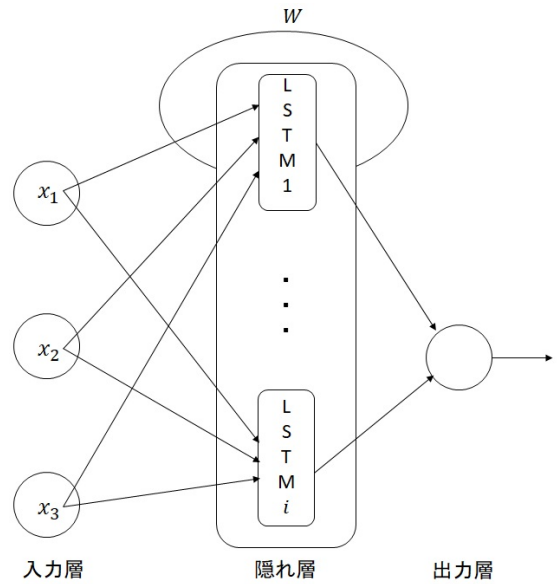


図 2: LSTM の全体構造

i とする. i 時刻 t における LSTM ブロック i の入力 z_i^t は次式で求められる.

$$v_i^t = \sum_{j=1}^m u_{ij}x_j^t + \sum_{k=1}^n w_{ik}h_k^{t-1} \quad (17)$$

$$z_i^t = f(v_i^t + b_i^{in}) \quad (18)$$

このとき,LSTM ブロック数を n , u_{ij} は入力層のノード j から LSTM ブロック i への結合の重み, w_{ik} を LSTM ブロック k から LSTM ブロック i への回帰結合の重み, b_i^{in} をバイアス, v_i^t を外部入力とする.

図 3 における IG は入力ゲートである. 入力ゲートはノードに入力された信号を取捨選択して伝達させる. 伝達させるかどうかは入力ゲート値 $g_i^{ig,t}$ ($0 \leq g_i^{ig,t} \leq 1$) で判定される. 値が大きいほど信号はそのまま伝達される. 入力ゲート値 $g_i^{ig,t}$ は次式で求められる.

$$v_i^{ig,t} = \sum_{j=1}^m u_{ij}^{ig}x_j^t + \sum_{k=1}^n w_{ik}^{ig}h_k^{t-1} + w_i^{ig}c_i^{t-1} \quad (19)$$

$$g_i^{ig,t} = \sigma(v_i^{ig,t} + b_i^{ig}) \quad (20)$$

このとき, u_{ij}^{ig} は入力層のノード j から LSTM ブロック i の入力ゲートへの結合の重み, w_{ik}^{ig} を LSTM ブロック k から LSTM ブロック i の入力ゲートへの回帰結合の重み, $v_i^{ig,t}$ を時刻 t の LSTM ブロック i の入力ゲートへの外部入力, w_i^{ig} をメモリセルから入力ゲートへの結合の重みとする. c_i^t を時刻 t の LSTM ブロック i のメモリセルの出力, z_i^t

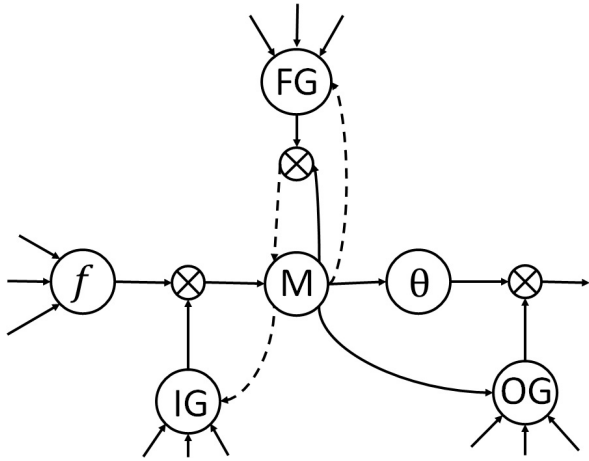


図 3: LSTM ブロックの概略

を時刻 t の LSTM ブロック i の出力, b_i^{fg} をバイアスとする。

図 3 における M はメモリセルである。メモリセルは過去の情報を記憶する。時刻 t における LSTM ブロック i のメモリセルへの入力 a_i^t は次式で求められる。

$$a_i^t = z_i^t g_i^{ig,t} \quad (21)$$

図 3 における FG は忘却ゲートである。忘却ゲートはメモリセルの情報の消去を制御する。忘却ゲートの値 $g_i^{fg,t}$ ($0 \leq g_i^{fg,t} \leq 1$) は次式で求められる。

$$v_i^{fg,t} = \sum_{j=1}^m u_{ij}^{fg} x_j^t + \sum_{k=1}^n w_{ik}^{fg} h_k^{t-1} + w_i^{fg} c_i^{t-1} \quad (22)$$

$$g_i^{fg,t} = \sigma(v_i^{fg,t} + b_i^{fg}) \quad (23)$$

このとき, u_{ij}^{fg} は入力層のノード j から LSTM ブロック i の忘却ゲートへの結合の重み, w_{ik}^{fg} を LSTM ブロック k から LSTM ブロック i の忘却ゲートへの回帰結合の重み, $v_i^{fg,t}$ を時刻 t の LSTM ブロック i の入力ゲートへの外部入力, w_i^{fg} をメモリセルから忘却ゲートへの結合の重みとする。 c_i^t を時刻 t の LSTM ブロック i のメモリセルの出力, h_i^t を時刻 t の LSTM ブロック i の出力, b_i^{fg} をバイアスとする。

メモリセルの回帰結合により, メモリセルの出力 c_i^t は次式で求められる。

$$c_i^t = a_i^t + g_i^{fg,t} c_i^{t-1} \quad (24)$$

図 3 における OG は出力ゲートである。出力ゲートは LSTM ブロックの出力を制御する。出力ゲ

トのゲート値 $g_i^{og,t}$ は次式で求められる。

$$v_i^{og,t} = \sum_{j=1}^m u_{ij}^{og} x_j^t + \sum_{k=1}^n w_{ik}^{og} h_k^{t-1} + w_i^{og} c_i^t \quad (25)$$

$$g_i^{og,t} = \sigma(v_i^{og,t} + b_i^{og}) \quad (26)$$

このとき, u_{ij}^{og} は入力層のノード j から LSTM ブロック i の出力ゲートへの結合の重み, w_{ik}^{og} を LSTM ブロック k から LSTM ブロック i のゲートへの回帰結合の重み, $v_i^{og,t}$ を時刻 t の LSTM ブロック i の入力ゲートへの外部入力, w_i^{og} をメモリセルから忘却ゲートへの結合の重みとする。 c_i^t を時刻 t の LSTM ブロック i のメモリセルの出力, h_i^t を時刻 t の LSTM ブロック i の出力, b_i^{og} をバイアスとする。

図 3 における θ は活性化関数である。メモリセルの出力は θ に入力される。 θ の出力と出力ゲート値により LSTM ブロックの出力が決定される。 LSTM ブロック i の時刻 t における出力を h_i^t とする。 h_i^t は次式で求められる。

$$h_i^t = \theta(c_i^t) g_i^{og,t} \quad (27)$$

2.3.2.RNN 部のフロー

RNN 部の処理の流れを **Algorithm** に示す。 **Algorithm** は定義ステップ, 初期処理ステップ, 学習ステップ, 予測ステップからなる。

定義ステップは 1~5 行目である。ここでは変数の定義を行う。

初期処理ステップは 7~9 行目である。 8 行目では過去時系列データを入力データと教師データの組に分割する。入力データは連続する n 個, 教師データはその直後のデータである。

学習ステップは 10~20 行目である。ここでは RNN に入力データを与える。結果として出力されたデータと教師データの誤差を記録する。記録された誤差が一定回数連続で減少していなければ, 学習は完了したと判断する。その場合学習ステップを終了し予測ステップに移る。減少していれば学習ステップを繰り返す。

予測ステップは 21~28 行目である。 22 行目で RNN に過去時系列データの最後から n 個のデータを入力する。末尾までのデータを与えられた RNN は過去時系列データ全体の直後の未来の値を予測する。 23, 24 行目で出力を予測結果と過去時系列データの末尾に追加する。 22~24 行目を繰り返すたびに 1 個ずつ未来を予測することができる。これらの処理は B 回繰り返される。

Algorithm RNN 部の処理の流れ

```

1: ·  $n$  = 連続して誤差が減少した回数
2: ·  $e$  = 誤差
3: ·  $\dot{e}$  = 前回の誤差
4: ·  $B$  = 予測する長さ
5: ·  $t$  = 現在の繰り返し回数
6: start
7: 回帰分析部より過去時系列データを受け取る
8: 受け取った時系列データを入力データと教師データのペアに分割する
9: RNN のネットワークを構築する
10: while  $n < 10$  do
11:   ネットワークに入力データを入力
12:    $e$  = ネットワークの出力と教師データの誤差
13:   ネットワークを学習させる
14:   if  $\dot{e} < e$  then
15:      $n = n + 1$ 
16:   else
17:      $n = 0$ 
18:   end if
19:    $\dot{e} = e$ 
20: end while
21: while  $t < B$  do
22:   過去時系列データの末尾までを入力する
23:   出力を予測結果の末尾に追加する
24:   出力を過去時系列データの末尾に追加する
25:    $t = t + 1$ 
26: end while
27: 予測結果を返す
28: end

```

2.4. 比較部

比較部では予測結果と実際の値を比較し精度を検証する. 予測結果の精度を定量的に評価するための指標として平均 2 乗誤差 (Mean Squared Error, 以下 MSE とする) を利用する. MSE は各 n 個の予測値 $Y = \{y_1, y_2, \dots, y_n\}$ と実際の値 $Z = \{z_1, z_2, \dots, z_n\}$ に対し, 次式で求められる.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - z_i)^2 \quad (28)$$

3. 検証実験

本研究で提案した手法を用いて時系列予測を行い, 予測の精度を検証した. 使用する時系列データは GDP, 株価, 総人口など実際の統計データを合計 11 系列使用した. 時系列の期間は, いずれも 1973 年~1998 年のものを使用している. 表 1 に時系列データの詳細を示す. データはいずれも 1 年ごとのデータである. 日経平均株価は毎日の終値の年平均, 米ドル円レートは 17 時時点の毎月平均の年平均により年次に変換している. 出国者数, 失業率, GDP は RESAS⁷ から入手した. 消費者物価指数, 総人口,

出生率, 住宅着工件数は e-Stat¹⁰ から入手した. 企業物価指数, 公定歩合, 米ドル/円レートは日本銀行⁸ から入手した. 日経平均株価は Yahoo!Finance¹³ から入手した.

時系列データ	本論文内での略称
島根県の出国者数	syukkoku
全国の失業率	situgyou
消費者物価指数	cpi ¹⁰
企業物価指数	cgpi ⁸
総人口	japanese
日経平均株価	nikkei ¹³
公定歩合	rate
出生率	syusseip
米ドル円レート	usdjpy
住宅着工件数	zyuutaku
GDP	gdp

表 1: 使用したデータ一覧

3.1. 実験概要

提案手法により予測する対象を表 1 の 11 系列の中から 1 系列選択する. 予測対象を除く残りの 10 系列を入力時系列の候補とする. 検証実験では, 回帰分析の手法として重回帰分析とファジィ回帰分析をそれぞれ使用した場合を比較し, 予測の精度を検証する. 次に RNN に対する入力時系列の長さ, RNN による予測時系列の長さを変更し, 精度がどのように変化するか検証する. 最後に回帰分析に入力する時系列データの系列数を変更し, 精度の変化を検証する.

3.2. 回帰手法比較検証

回帰分析の手法を重回帰分析とした場合, ファジィ回帰分析とした場合の両方で RNN による予測を行い, 予測の精度を比較した. いずれも 1973 年~1992 年の対象データを回帰分析により他時系列よりまとめて RNN に入力した. RNN は 1993 年~1998 年の対象の数値を予測する. 入力時系列数は予測対象ごとに 1~10 まで変化させた. 1 パターンにつき 10 回予測を行い, 予測対象ごとに MSE の平均をとっている. ファジィ回帰分析の場合, 対象が統計データであり信頼性は高い. よってファジィ線形回帰においてデータの信頼性の低さを表す h は $h = 0.1$ とした. 実験結果を図 4,5 に示す.

図 4 では予測対象を企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数, GDP とした場合, 図 5 では住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口とした場合の結果を示している. 結果としては, 重回帰分析がファジィ回帰分析に比べ高い精度を示している. この結果より, 本研究においては回帰手法として重回帰分析を採用する.

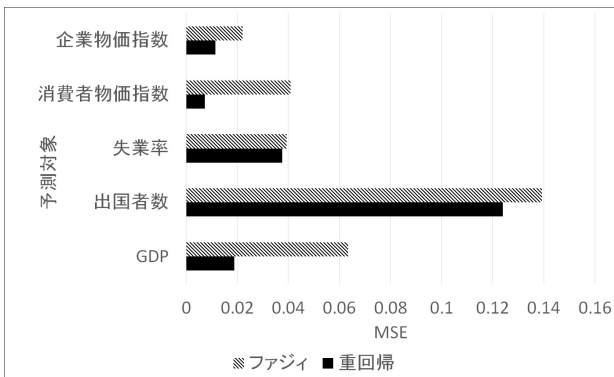


図 4: 回帰手法比較 (予測対象:企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数,GDP の場合)

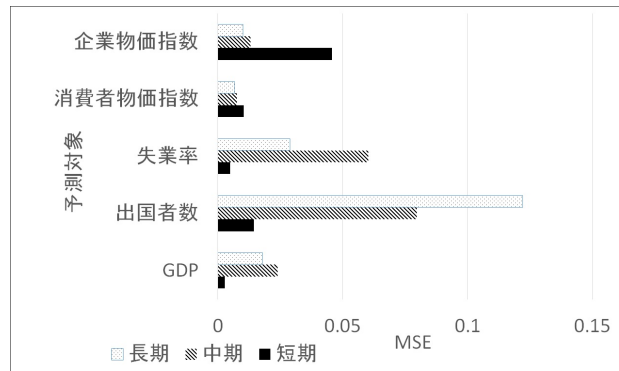


図 6: 入力時系列長比較 (予測対象:企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数,GDP の場合)

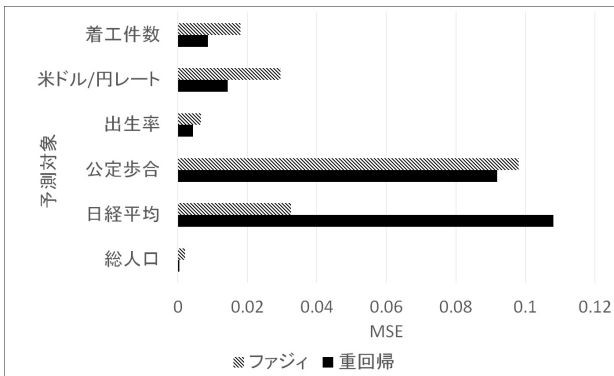


図 5: 回帰手法比較 (予測対象:住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口の場合)

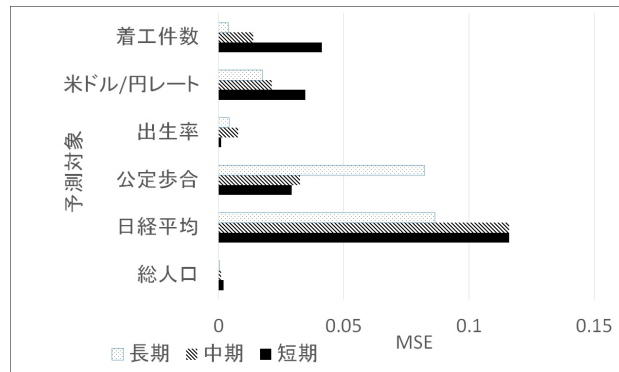


図 7: 入力時系列長比較 (予測対象:住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口の場合)

3.3. 時系列長検証

この節では RNN への入出力データの長さを変更し, 精度がどのように変化するか検証している. 入力時系列数は予測対象ごとに 1~10 まで変化させ, かつ 1 パターンにつき 10 回予測を行い, 予測対象ごとに MSE の平均をとっている.

3.3.1. 入力時系列長検証

入力時系列長を変更する. 入力時系列長は短期:1973~1983, 中期:1973~1988, 長期:1973~1992 の 3 パターンとした. その後の RNN による予測の長さは 5 年とした.RNN による予測は 1 パターンにつき 10 回行い,MSE の平均をとった. 結果を図 6,7 に示す. 縦軸は MSE 値であり, 低いほど精度が良いことを示している.

図 6 では予測対象が企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数,GDP の場合, 図 7 では住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口とした場合の結果を示している. 長期の時系列データを与えたほうが高精度な予測が可能であると考えられたが, 実際には必ずしもそうならない. これは, データごとに

長期変動と短期変動の特性があるためと考えられる. 短期変動の影響が強い時系列を予測する場合, 短期の時系列データを与えたほうがむしろその特性をより学習できるものと考えられる.

3.3.2. 出力時系列長検証

出力時系列を変更する場合は, 入力時系列の長さを 1973~1983 で固定とした. 予測する長さを短期:1984~1989, 中期:1984~1994, 長期:1984~1998 とした.RNN による予測は 1 パターンにつき 10 回行い,MSE の平均をとった. 結果を図 8,9 に示す. 図 8 では予測対象を企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数,GDP とした場合, 図 9 では住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口とした場合を示している. 縦軸は MSE 値であり, 低いほど精度が良いことを示している.

全体的に長期時系列の精度が低下する傾向が見られる. これは遠未来の予測は困難なため, 精度が低下するためと考えられる.

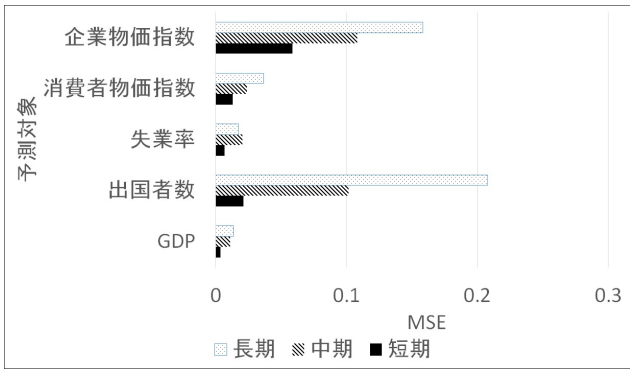


図 8: 出力時系列長比較 (予測対象:企業物価指数, 消費者物価指数, 失業率, 島根県の出国者数,GDP の場合)

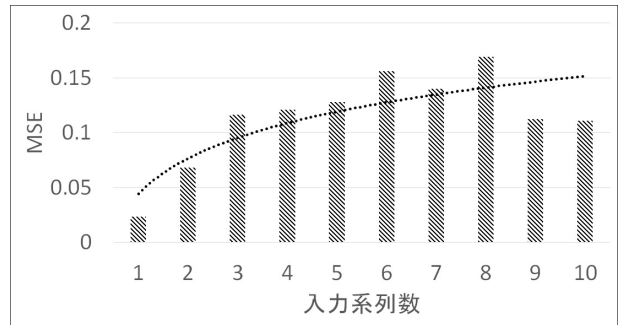


図 11: 系列数変更 (予測対象:日経平均株価)

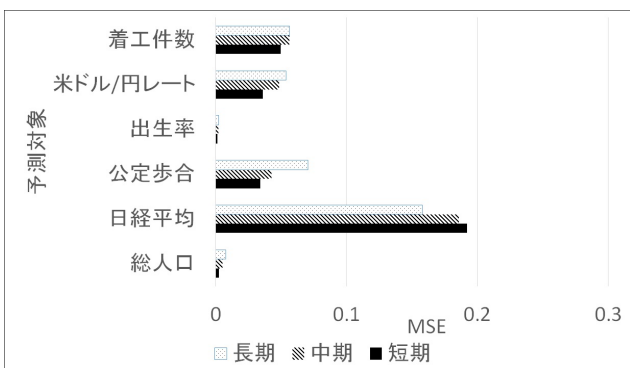


図 9: 出力時系列長比較 (予測対象:住宅着工件数, 米ドル/円レート, 出生率, 公定歩合, 日経平均株価, 総人口の場合)

3.4. 入力時系列数検証

入力時系列数を 1~10 個まで変化させたとき, 最終的な RNN による予測の精度がどう変動するかを検証する. いずれも入力時系列データは 1973~1992 のものとする.RNN により 1993~1998 の対象の数値を予測した.RNN による予測は 1 パターンにつき 10 回行い,MSE の平均をとった. 実験結果を図 10,11 で示す. 縦軸は MSE 値であり, 低いほど精度が高いことを表す. 曲線は対数関数による近似曲線である.

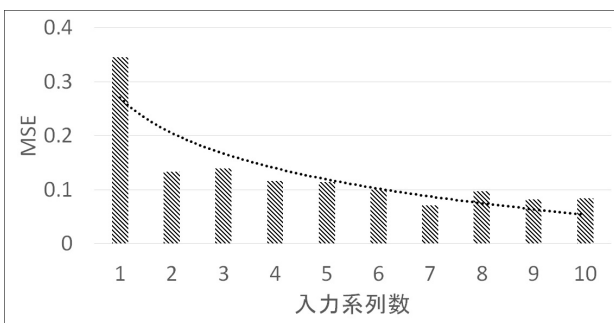


図 10: 系列数変更 (予測対象:出国者数)

出国者数を予測した場合 (図 10), 系列数が増加すると精度も向上する傾向が見られる. しかし, 日経平均株価を予測した場合 (図 11) はむしろ系列数が増加すると精度が悪化する傾向が見られる. 入力データの系列数は多ければ精度が向上するとは言いきれず, むしろ対象によっては精度の悪化を招くことが実験的に確認された.

4. おわりに

本研究では, 回帰分析と RNN を用いて複数時系列から単一時系列予測を行う手法を提案した.RNN として長期時系列の予測に優れる LSTM を利用した.

提案した手法を基に回帰分析の手法, 入力として与える時系列データの長さ, 系列数, 予測として出力する時系列の長さをそれぞれ変更する検証を行った. 予測の精度は MSE を用いて評価した. 回帰分析の手法の比較検証では重回帰分析を用いた方がファジィ回帰分析を用いるより全体的に良好な精度を示すことが確認された. 日経平均株価など一部のデータではファジィ回帰分析を用いた方が高い精度を出している. 日経平均株価は短期的に複雑な変動をする. ファジィ回帰分析のあいまいさにより変動の一部が吸収され, 結果として全体的な変動を予測しやすくなった可能性がある. 入力時系列長を変更する検証では長期の時系列データを与えれば単純に精度が向上するわけではないことが確認された. 出力時系列長を変更する検証では, 長期の予測は精度が悪化する傾向があることが確認された. 入力時系列数を変更する検証では, 予測対象によって系列数変更に伴う精度の変化の様態が異なることが確認された. 日経平均株価を予測した場合は, 全体的に系列数が増えるに従い精度が悪化する傾向がある. しかし, 一時的に精度が向上する場合もある. 単純な数だけでなく入力時系列データごとの特性が影響を及ぼしている可能性がある. 予測対象によって, 適切な入力時系列データの長さや系列数は大きく異なると考えられる.

今後の課題として, 現在入力時系列の系列数を変更する場合, 系列を順番に追加することで系列数のみを変更している. これに対し今後は, 入力時系列の組み合わせを考慮する. また, 本研究では入力時系列として自己系列を含んでいない. 自己系

列を含むことにより精度が向上する可能性がある。
今後は自己系列を含む場合についても検証する。

参考文献

- [1] Grimm, Laurence G., and Yarnold, Paul R 編.
”研究論文を読み解くための多変量解析入門
基礎編 -重回帰分析からメタ分析まで-.”北
大路書房, 2016.
- [2] Hochreiter, Sepp, and Jürgen Schmidhuber.
”Long short-term memory.” *Neural computa-
tion* 9.8, pp.1735-1780, 1997.
- [3] 栗原伸一. ”入門 統計学 -検定から多変量解
析・実験計画法まで-.” オーム社. 2011.
- [4] 松井藤五郎, and 汐月智哉. ”LSTM を用いた
株価変動予測.” 人工知能学会全国大会論文集
2017 年度人工知能学会全国大会 (第 31 回)
論文集 2D3-OS-19a-2, pp.1-2. 一般社団法人
人工知能学会, 2017.
- [5] 南正太朗. ”LSTM-RNN を用いたイベント考
慮後の株価時系列予測.” 人工知能学会 金融
情報学研究会 (第 19 回) 資料, pp.42-44. 一般
社団法人 人工知能学会, 2017.
- [6] 宮崎邦洋, and 松尾豊. ”深層学習を用いた株
価予測の分析.” 人工知能学会全国大会論文集
2017 年度人工知能学会全国大会 (第 31 回)
論文集 2D3-OS-19a-3, pp.1-3. 一般社団法人
人工知能学会, 2017.
- [7] 内閣府. ”RESAS 地域経済分析システム.”
<https://resas.go.jp/>.
- [8] 日本銀行. ”日本銀行時系列統計データ検索サ
イト.” [http://www.stat-search.boj.or.
jp/index.html](http://www.stat-search.boj.or.jp/index.html).
- [9] 趙強福, and 樋口龍雄. ”人工知能 AI の基礎
から知的探索へ.” 共立出版, 2017.
- [10] 総務省. ”e-Stat 政府統計の総合窓口.” [https:
//www.e-stat.go.jp/](https://www.e-stat.go.jp/).
- [11] 田中英夫, 上嶋暁, and 浅居喜代治. ”ファジィ
関数による線形回帰モデル.” 日本オペレー
ションズ・リサーチ学会論文誌 25.2, pp.162-
174, 1982.
- [12] 田中英夫. ”可能性回帰分析.” 日本ファジィ学
会誌 5.6, 1260-1272, 1993.
- [13] Yahoo. ”YahooFinance.” [https:
//finance.yahoo.com/](https://finance.yahoo.com/).
- [14] 渡辺則夫. ”ファジィ時系列解析.” 共立出版.
2018.