

## 第1章 建設投資と社会資本整備

### 「建設経済モデル」へのAI導入

#### 要旨

建設投資額の予測は、着工から施工までに長時間を要するという建設工事特有の理由から、「マクロ経済動向から着工額を予測する」前半パートと「着工額を投資額に展開する」後半パートの2つに分けて予測が行われる。

本研究では、「マクロ経済動向から着工額を予測する」前半パートについて、一般的にAIと呼ばれる機械学習・深層学習手法を適用することで、当研究所「建設経済モデル」の予測精度向上が図れないか検証を行った。

検証の結果、様々な手法の中で深層学習手法の1つである NHiTS (Neural Hierarchical Interpolation for Time Series Forecasting) の予測精度が最も高く、従来の手法より約11%予測精度が向上する結果となった。

検証結果を踏まえ、「建設経済モデル」の「マクロ経済動向から着工額を予測する」前半パートに NHiTS を導入することとする。

なお、深層学習手法は、ハイパーパラメータと呼ばれる検証者が設定すべき係数が多くあり、また1回の検証に長いものだと半日を要するため、係数の設定は初期設定+数パターンでしか行えていない。このため、検証できなかった他の係数を設定すれば、今回選ばれなかった他の手法の方が、予測精度が高くなる可能性が残されている点にご留意頂きたい。

#### はじめに

建設投資額の予測は、工事が着工して着工額として計上されてから、実際に施工して投資額（出来高）に計上されるまでに長い時間を要するため、「マクロ経済動向から着工額を予測する」前半パートと「着工額を投資額に展開する」後半パートの2つに分けて予測が行われる。

昨年度の研究では、「着工額を投資額に展開する」後半パートについて、国土交通省「建設総合統計」の着工額・出来高・手持ち工事高にコイック・ラグ（幾何級数型分布ラグ）を応用した手法を提案し、それまでの当研究所「建設経済モデル」で使用していたアーモン・ラグ（多項式分布ラグ）を用いた手法と比べて、予測精度が向上することを示した。

一方で、「マクロ経済動向から着工額を予測する」前半パートについては、簡便に ARIMA<sup>1</sup> を用いて構築しており、予測精度は従来と比べて同等以上であったものの、改善の余地が残されていた。

そこで、本研究では、「マクロ経済動向から着工額を予測する」前半パートについて、近年急速に発展し、様々な予測シーンで活用され始めている機械学習・深層学習手法を適用することで、「建設経済モデル」の予測精度の向上が図れないか検証を行った。

「1. データ」では検証に用いたデータを説明する。「2. 検証環境」では検証に用いた開発環境を説明する。「3. 予測手法群」では検証に用いた様々な機械学習・深層学習手法を説明する。

「4. 評価方法」では予測精度の評価方法を説明する。「5. 検証結果」では様々な手法の検証結果を説明する。「6. 今後の課題」では今後の研究の方向性を示す。

### 1. データ

検証に用いたデータはつぎのとおりである。

#### (1) 目的変数

予測対象である目的変数は、当研究所「建設経済モデルによる建設投資の見通し」の公表最小単位である7項目（民間住宅、民間非住宅、民間土木、政府住宅、政府非住宅、政府土木一般、政府土木企業）の予測に用いている、「建設総合統計」の着工ベース7項目（民間建築居住用、民間建築非居住用、民間土木、公共建築居住用、公共建築非居住用、公共土木一般、公共土木企業）である。これらの月次値を「1.(2) 実質換算手法」により実質換算し、四半期集計した値を用いた。データ数は1994年1-3月期から2019年1-3月期<sup>2</sup>までの101である。

#### (2) 実質換算手法

目的変数は、図表1のとおり、実質換算した。

図は、左に実績期間を、右に予測期間を、上に名目値系列を、下に実質値系列を記載している。実績期間については、当期末名目手持ち工事高と名目投資額の和をその期の国土交通省「建設工事費デフレーター」で実質換算し、これと先期末（当期首）実質手持ち工事高の差として

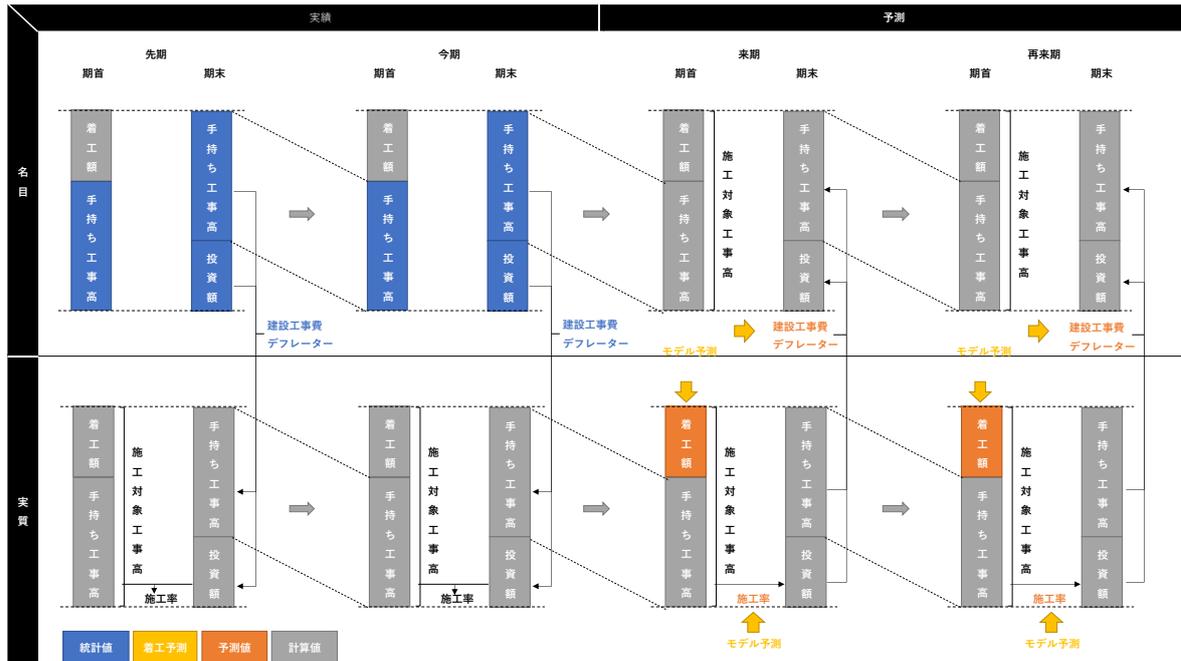
---

1 ARIMA とは、自己回帰和分移動平均（Auto Regressive Integrated Moving Average）のことで、ある変数の階差（この階差が「和分（Integrated）」のことで、例えば、今月と前月の差を意味する。）を、過去の自身の値である自己回帰（Auto Regressive）部分と、過去の自身の誤差項の加重和（Moving Average）部分で表現する手法である。

2 「建設総合統計」は、毎年6月に、確定した建設投資額の実績値から算出される直近の補正率を用いて、前年度から3カ年分が改定されるため、確定値であるこの期間の値を用いた。

実質着工額を推計している。予測期間については、予測した実質着工額と先期末（当期首）実質手持ち工事高の和に予測した施工率を掛けて、当期末実質手持ち工事高と実質投資額を推計し、「建設工事費デフレーター」を掛けて名目換算した上で、これらの和と先期末（当期首）名目手持ち工事高の差として名目着工額を推計している。<sup>3</sup>

図表1 実質換算手法 イメージ



(出典) 当研究所にて作成

### (3) 説明変数

説明変数には、日本経済新聞社「NEEDS 日本経済モデル」の 356 変数<sup>4</sup>を用いた。データ数は目的変数と同じく 1994 年 1-3 月期から 2019 年 1-3 月期までの 101 である。

### (4) 前処理

深層学習手法の学習効率を高めるため、すべての変数を 0 から 1 の間になるように相似変換<sup>5</sup>して学習を行った。また、目的変数の予測結果は、逆相似変換で元の大きさに戻している。

3 詳細は、本レポート別稿「建設業の物価に関する把握の課題と予測可能性」の「1.(2) ② 契約時価格指数の推計方法」をご参考頂きたい。

4 季節指数・調整項・ダミー変数・スイッチ変数・タイム変数・予測対象自体の変数・データ数が少ない変数を除いたすべての変数。

5 Scikit-Learn の MinMaxScaler を使用。

## 2. 検証環境

検証環境はつぎのとおりである。

計算機	Google Colab Pro
プログラミング言語	Python 3.7.13
主なライブラリ	Darts 0.20.0 ほか

Google Colab は Google 社が提供するクラウドベースの機械学習用開発環境サービスである。Python は機械学習・深層学習分野で最も使われるプログラミング言語である。Darts は時系列の操作と予測を容易に行える Python ライブラリであり、古典的手法から深層学習手法まで、様々な時系列予測手法を含んでいる<sup>6</sup>。

## 3. 予測手法群

Darts に含まれる様々な手法を分類したのが図表 2 である。統計的手法には、ARIMA の他に Exponential Smoothing (指数平滑化法) や Fast Fourier Transform (高速フーリエ変換)、Facebook (Meta) 社が開発した Prophet などがある。機械学習手法には、Decision Tree (決定木) を弱学習器<sup>7</sup>として、Bagging<sup>8</sup>を行う Random Forest、Boosting<sup>9</sup>を行う LightGBM や CatBoost がある。深層学習手法は、様々な手法が日々開発されており、一概に分類できないが、FNN・CNN・RNN・Transformer に分類した。なお、右列の「説明変数追加 (過去値のみ)」とは説明変数 (過去値のみ) を追加できるかどうかを、「説明変数追加 (将来値含む)」とは説明変数 (将来値含む) を追加できるかどうかを○×で表記したものである。今回は、このうち各分類を代表する※印の 10 手法について検証を行った<sup>10</sup>。

6 Darts 以外にも Sktime や Kats など様々な時系列予測ライブラリが存在するが、今回は分野的な種類の多さから Darts を選択した。

7 弱学習器とは、単独では精度の低い予測手法のこと。この場合は決定木が弱学習器であり、これを複数用いることで強学習器 (精度の高い予測手法) にしている。

8 バギングとは、学習データから復元抽出により多数のサンプルデータを作り、それぞれを独立・並列に多数の弱学習器 (決定木) に学習させた結果の多数決をとったものを出力とする手法である。

9 ブースティングとは、バギングのように弱学習器 (決定木) を独立・並列に学習させるのではなく、最初の弱学習器で上手く推定できなかった部分を推定するために重みを付けてつぎの弱学習器で学習を行い、この弱学習器で上手く推定できなかった部分を推定するために重みを付けてそのつぎの弱学習器で学習を行い…というように 1 つずつ順番に直列的に弱学習器を構成していく。最終的には、精度の高い弱学習器に重みを付けて出力を決定する手法である。

10 一度 Darts 内の全手法を係数初期値で検証したが、紙面や係数調整の時間的な制約上、予測精度が高いと思われる手法や分類上検証すべき手法に絞って詳細な検証を行った。

図表2 予測手法群の分類

分類			手法	説明変数追加	
				(実績値のみ)	(将来値含む)
統計的(古典的)			Naïve Seasonal ※	×	×
			ARIMA ※	×	○
			LinearRegression ※	○	○
			ExponentialSmoothing	×	×
			BATS and TBATS	×	×
			Theta and FourTheta	×	×
			Prophet	×	○
			Fast Fourier Transform	×	×
			KalmanForecaster	×	○
			Croston method	×	×
機械学習	Decision Tree	bagging	RandomForest ※	○	○
		boosting	LightGBM ※	○	○
			CatBoost	○	○
深層学習	FNN		NBEATS	○	×
			NHiTS ※	○	×
	CNN		TCN ※	○	×
	RNN		RNN	○	○
			LSTM ※	○	○
			GRU	○	○
	Transformer		Transformer ※	○	×
		TFT ※	×	○	

FNN : Feedforward Neural Network / 順伝播型ニューラルネットワーク  
 CNN : Convolutional Neural Network / 畳み込み型ニューラルネットワーク  
 RNN : Recurrent Neural Network / 再帰型ニューラルネットワーク  
 ※本研究で検証する手法

(出典) 当研究所にて作成

## 4. 評価方法

評価方法はつぎのとおりである。

評価項目	「1.(1) 目的変数」に示した7項目
評価期間	8四半期先予測の直近10年間（計10回） <sup>11</sup>
評価指標	7項目合計のRMSE <sup>12</sup> の平均値
評価部門	①目的変数+説明変数（過去値のみ） ②目的変数+説明変数（将来値含む）

「建設総合統計」の着工ベース7項目の実績値について、2年先（8四半期先）予測を直近10年間分繰り返し、予測値と実績値との差をRMSEの平均値で評価した。

なお、評価は、①目的変数のみを用いた予測や目的変数に説明変数の過去値のみを加えた予測と②目的変数に説明変数の将来値（予測期間の実績値）も加えた予測の2部門に分けて行った。

今回のような検証における予測では、予測期間より前の実績値しか用いない①と比べて、予測期間の実績値を用いることができる②の方が一般的に予測精度は高い。一方で、実際の予測では、②は予測期間の予測値（正解値ではない）を用いるため、この予測値が誤っていた場合には、その影響を受けて、予測精度が下がる可能性がある。①はこのような予測精度の低下の可能性がないため、②より頑健な予測といえる。これらを同等に扱うことは適切でないため、区別して評価を行った。

11 例えば、12年度までのデータを用いて13年度と14年度を予測し、つぎに13年度までのデータを用いて14年度と15年度を予測するというを直近10年間について繰り返した。

12 Root Mean Squared Error（平均二乗偏差）。今回の場合、予測値と実績値の差の二乗を予測期間の8四半期分合計し、8で割って平方根をとったもの。

## 5. 検証結果

検証結果はつぎのとおりである。

### (1) 目的変数+説明変数（過去値のみ）

目的変数のみや目的変数に説明変数の過去値のみを加えた予測の検証結果を示したのが図表3である。各列は様々な手法群を、各行は予測期間を、最終行は予測期間平均を表す。各セルは、評価指標である7項目合計のRMSEの平均値を記載し、各予測期間での最小値（最も予測精度が高いもの）を太字にした。図表3のとおり、予測期間平均の評価指標はNHITSが最も小さく、従来の手法より約11%予測精度が向上する結果となった。

図表3 様々な手法の検証結果（説明変数過去値のみ）

予測期間	単変量(Univariate)			多変量（過去値のみ）					
	Naïve Seasonal	AutoARIMA ≡ 昨年度研究 使用モデル	Linear Regression	Random Forest	LightGBM	LSTM	NHiTS	TCN	Transformer
08-09	475,468	421,595	575,266	687,437	731,834	365,581	<b>325,803</b>	397,209	486,048
09-10	536,580	367,074	380,817	785,714	737,781	475,081	404,945	469,791	579,532
10-11	317,910	346,326	<b>228,709</b>	404,192	445,082	317,911	272,325	472,834	285,801
11-12	<b>244,382</b>	356,336	322,561	282,394	274,445	<b>231,594</b>	336,384	496,001	413,077
12-13	356,987	374,450	373,472	<b>289,871</b>	304,624	291,293	368,904	459,475	321,452
13-14	<b>319,117</b>	355,582	381,890	341,435	328,763	372,628	408,875	575,370	342,252
14-15	318,499	237,503	<b>229,951</b>	325,832	352,029	323,226	238,634	327,922	312,302
15-16	297,223	425,227	251,098	233,081	302,173	282,796	244,211	290,874	<b>169,771</b>
16-17	<b>194,404</b>	221,116	208,552	208,300	205,800	212,903	197,708	201,960	<b>157,515</b>
17-18	219,857	276,617	<b>163,588</b>	183,027	204,739	234,181	168,487	185,556	203,248
Ave	343,506	344,774	332,236	420,835	430,547	319,746	<b>307,372</b>	407,136	351,761

（出典）当研究所にて作成

(2) 目的変数+説明変数（将来値含む）

目的変数に説明変数の将来値（予測期間の実績値）も加えた予測の検証結果を示したのが図表4である。なお、「単変量（Univariate）」の列は参考再掲した。図表4のとおり、予測期間平均の評価指標はLSTMが最も小さく、従来の手法より約9%予測精度が向上する結果となった。

図表4 様々な手法の検証結果（説明変数将来値含む）

予測期間	単変量(Univariate)			多変量（将来値含む）			
	Naïve Seasonal	AutoARIMA ≡ 昨年度研究 使用モデル	Linear Regression	Random Forest	LightGBM	LSTM	TFT
08-09	475,468	421,595	575,266	542,856	475,424	<b>292,421</b>	834,288
09-10	536,580	367,074	380,817	516,905	487,186	<b>333,878</b>	843,252
10-11	317,910	346,326	<b>228,709</b>	326,512	449,733	284,360	729,642
11-12	<b>244,382</b>	356,336	322,561	274,936	402,412	263,782	658,956
12-13	356,987	374,450	373,472	287,710	<b>277,471</b>	307,470	592,997
13-14	<b>319,117</b>	355,582	381,890	334,973	353,259	415,265	421,581
14-15	318,499	237,503	<b>229,951</b>	310,388	305,530	333,988	466,052
15-16	297,223	425,227	251,098	<b>206,609</b>	253,979	300,921	355,634
16-17	<b>194,404</b>	221,116	208,552	195,132	214,027	346,327	375,434
17-18	219,857	276,617	<b>163,588</b>	212,452	204,331	230,309	296,401
Ave	343,506	344,774	332,236	340,780	357,039	<b>314,562</b>	589,540

（出典）当研究所にて作成

以上の2部門の検証結果をみても、予測期間平均の評価指標はNHITSが最も小さかった。先述のとおり、②の手法の方が予測精度は高いが、①の手法の方が予測は頑健と相反するのが一般的であるにもかかわらず、NHITS（①の手法）は予測精度もLSTM（②の手法）より高く、「建設経済モデル」の予測精度向上の観点においては最良の手法と言える。

この検証結果を踏まえ、「建設経済モデル」の「マクロ経済動向から着工額を予測する」前半パートにNHITSを導入することとする。

なお、深層学習手法は、ハイパーパラメータと呼ばれる検証者が設定すべき係数が多くあり、また1回の検証時間が長いものと半日を要するため、係数の設定は初期設定+数パターン（3～10パターン）でしか検証を行えなかった。また、深層学習手法としてはデータが少ないことを考慮し、この数パターンは手法を簡素化する方向で選択した。このため、今回選ばれなかった他の手法でも他の係数を設定すれば、今回選ばれたものより予測精度が高くなる可能性が残されている点にはご留意頂きたい。

## 6. 今後の課題

最後に、今後の「建設経済モデル」の改良の方向性を示す。

### (1) 特徴量選択

本研究では、「NEEDS 日本経済モデル」の変数群を原則すべて用いたが、説明変数の数に対して、データ数が少なく、説明変数のノイズに過学習して、予測精度が低下するいわゆる「次元の呪い」が起きている可能性がある。このため、様々な特徴量選択手法を用いて、有用な説明変数に絞り込みを行いたい。

### (2) 確率論的予測

現在の「建設経済モデル」は、1つの値のみを予測するという決定論的予測モデルであるが、予測自体の不確実性を考慮して、予測値がどれくらいの確率でどのくらいの範囲に収まるかを示す確率論的予測モデルも存在する。機械学習・深層学習手法にも確率論的予測を行える手法が提案されており、確率論的予測の実装を行いたい。

### (3) 感度分析やアンサンブル予測

NHiTS は予測精度が高いものの、説明変数の過去値しか使用していないため、説明変数の将来値が変わると予測値がどう変わるかといった感度分析を行うことができない。このため、LSTM などの説明変数の将来値も含んだ手法の採用や、こうした手法とのアンサンブル予測の実装を検討したい。

### (4) 説明可能な AI

深層学習手法は、一般的に各説明変数がどの程度予測に影響を与えたかがわかりにくい「ブラックボックス」な手法と呼ばれる。一方で、「なぜそう予測したか」の説明性が重要な場合がある。このため、深層学習手法を線形回帰により近似する LIME や SHAP などの局所的説明手法を適用したり、データ数が少ない場合に予測精度が低くなりがちではあるが、TFT などのどの変数に着目しているかを明示できる手法を採用したりといった検討を行いたい。

## (5) 最新手法の検証

深層学習を用いた時系列予測は、世界中で日々新たな手法が開発されている。予測精度が高い手法が新たに開発された場合には、「建設経済モデル」に適用できないか継続的に検証を行いたい。

## 参考文献

- BREIMANLEO. (2001) . Random Forests. Kluwer Academic Publishers. 参照先:  
<https://link.springer.com/content/pdf/10.1023/A:1010933404324.pdf>
- Challu Cristian, Olivares Kin G., Oreshkin Boris N., Garza Federico, Mergenthaler Canseco Max, Dubrawski Artur. (2022) . N-HiTS: Neural Hierarchical Interpolation for Time Series Forecasting. arXiv. 参照先: <https://arxiv.org/pdf/2201.12886.pdf>
- Chen Yitian, Kang Yanfei, Chen Yixiong, Wang Zizhuo. (2019) . Probabilistic Forecasting with Temporal Convolutional Neural Network. arXiv. 参照先:  
<https://arxiv.org/pdf/1906.04397.pdf>
- Herzen Julien, Lassig Francesco, Piazzetta Giuliano Samuele, Neuer Thomas, Tafti Leo , Raille Guillaume, . . . Grosch Gael . (2022) . Darts: User-Friendly Modern Machine Learning for Time. Journal of Machine Learning Research. 参照先:  
<https://www.jmlr.org/papers/volume23/21-1177/21-1177.pdf>
- Ke Guolin, Meng Qi, Finley Thomas, Wang Taifeng , Chen Wei, Ma Weidong, . . . Liu Tie-Yan. (2017) . LightGBM: A Highly Efficient Gradient Boosting Decision Tree. 参照先:  
<https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf>
- Lim Bryan, Arık Sercan O., Nicolas Loef, Pfister Tomas. (2020) . Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting. arXiv. 参照先: <https://arxiv.org/pdf/1912.09363.pdf>
- Vaswani Ashish, Shazeer Noam, Parmar Niki, Uszkoreit Jakob, Jones Llion , Gomez Aidan N., . . . Polosukhin Illia. (2017) . Attention Is All You Need. arXiv. 参照先:  
<https://arxiv.org/pdf/1706.03762.pdf>
- Zeng Ailing, Chen Muxi, Zhang Lei, Xu Qiang . (2022) . Are Transformers Effective for Time Series Forecasting? arXiv. 参照先: <https://arxiv.org/pdf/2205.13504.pdf>