

深層学習を用いた日経平均株価の予測手法の改良～ミニバッチ学習の導入～ Improvement for the Forecasting Method of the Nikkei Stock Average using Deep Learning ~Adopting for Minibatch Learning~

1941083 猫本 遼太

Ryota NEKOMOTO

指導教員 秋葉 知昭

The purpose of this study is to improve the accuracy of the Nikkei Stock Average forecast so that an increasing number of individual investors can use it as an indicator for selecting stocks and timing their trades. In this study, I considered Nikkei Stock Average forecast using deep learning with LSTM and CNN. In addition, this study compared these accuracies by introducing improvements such as hyperparameter tuning and minibatch learning, adding a new optimization algorithm, and relaxing the correctness criterion. As a result, the hyperparameter tuning improved the accuracy, but the minibatch learning didn't improve the accuracy.

1. 緒言

近年,自己資産形成の必要性が謳われ,株取引を始める一般投資家の数が増加している.そこで銘柄の選択や売買のタイミングを計る指標として日経平均株価の予測する方法を提案する.

本研究では株価や指標の予測精度向上の要因の1つである LSTM と畳み込みの導入を行った関口の研究[1]に新たに最適化アルゴリズムの追加とミニバッチ学習を導入し,結果を比較する.

2. 先行研究

2.1 先行研究の概要

関口の研究[1]では,LSTM を使用し,1 日の四本値 10 日分のデータから 11 日目の終値を予測した.畳み込みは 11 個目の 1 つの値を元データに追加し,再度学習する 2 回トレーニング方式を使用した.最適化アルゴリズム (以下 Optimizer) は SGD, RMSprop, Adam, Radam, Adadelata を用いた結果を比較し考察した.

2.2 収集データ

関口の研究[1]では, 20 年間,4899 日分の日経平均株価,TOPIX,為替対円米ドル,日経平均株価先物の四本値のデータを収集し,日経平均株価単体とそれぞれの指標を日経平均株価と同時に学習させた場合の 4 つの組み合わせで比較した.

2.3 パラメータと結果

損失関数は MSELoss を使用し,学習率は SGD が 0.1, Adadelata は 1 から 1.5 他の 3 つは 0.01 から 0.05 とした.epoch は 500 でバッチサイズは 10 とした.

結果は loss のグラフにおいて SGD は学習曲線が見られず,他 4 つの Optimizer は学習曲線が確認

できた.最も高い正解率は畳み込みを利用した Adadelata の 13%で,正解条件は誤差 10 円未満だった.

日経平均株価+畳み込み

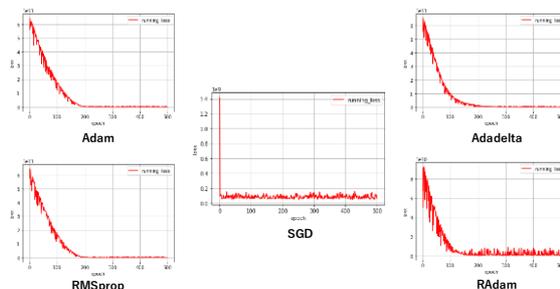


図1 各 Optimizer における loss の推移

3. 検知手法の提案

3.1 先行研究の問題点と改善

関口の研究[1]の問題点は SGD が学習できなかったこと,他の金融指標を用いても正解率が改善しなかったこと,正解条件が厳格だったことだ.

本研究では Optimizer に新たなアルゴリズムを追加し,ミニバッチ学習を導入する.

3.2 検知手法の概要

本研究では LSTM と畳み込みを使用し,1 日の四本値 x 日分のデータを入力し $x + 1$ 日目の終値を予測する.ここでフレームワークは Pytorch を使用した.またミニバッチ学習を導入し,精度の向上を目指す.Optimizer は関口の研究[1]に AdamW を加えて結果を比較する.

3.3 収集データ

本研究では HYPER SBI より 2003 年から 2023 年までの 20 年間,4897 日分の日経平均株価の四本値を収集し,①チューニング前②チューニング後③

ミニバッチ学習を導入の3つの組み合わせで比較する。また②,③において正解条件を誤差 50 円未満としたものも示す。

3.4 パラメータ及びモジュールの設定

損失関数は MSELoss を使用した。学習率は Adadelata には設定せず、その他は 0.1 から 0.9 の間で、データ入力日数 x の値は 5 から 50 の間で、層の数は 1 から 5 の間で、バッチサイズは 2 から 50 の間でそれぞれ Optuna を用いてチューニングを行い、一番損失が低くなったものを採用した。epoch は 500 と 1000 に設定した。

4. 結果及び考察

4.1 出力データ

epoch, loss, 正解率, training_accuracy, test_accuracy を出力した。図 2 と図 3 は正解条件を誤差 50 円未満とし、畳み込みを利用したミニバッチ学習導入後の loss と正解率の推移を表したグラフである。

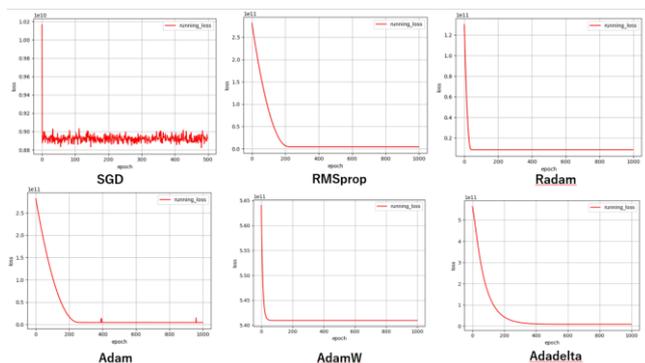


図 2 各 Optimizer における loss の推移

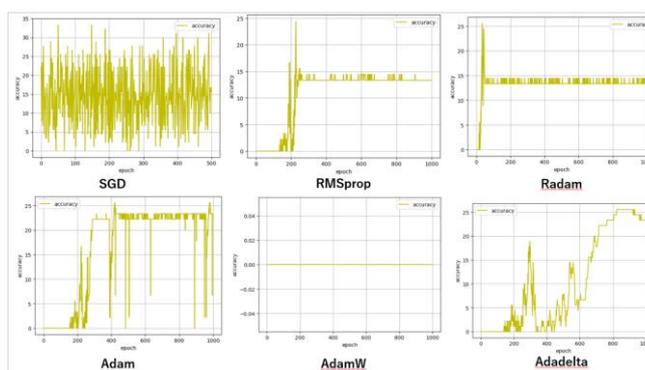


図 3 各 Optimizer における正解率の推移

4.2 Optimzier の比較

学習に使う損失を数値化した loss を関口の研究[1]と比較した結果、SGD では loss の学習曲線が見られなかった。また新たに追加した AdamW は一定の段階で学習が止まった。一方で他 4 つの手法は関

口の研究[1]と同じように学習曲線が見られ、損失の値も大幅に減少している。さらに学習が進行しているときに起きていた分散が大幅に減少して滑らかな学習曲線を描くようになった。しかし Radam については 100epoch 程度早く急激に損失が低下した。次に正解率を関口の研究[1]と比較すると SGD は分散しており、AdamW は一度も正解しなかった。しかし、他 4 つの手法は正解率を比較すると、2 倍から 3 倍程度高くなった。さらに Adam, RMSprop, Radam の正解率の推移は一定の精度を維持し続けるように変化した。一方で Adadelata は学習の後半において正解率が高くなるようになった。

4.3 考察

本研究の提案手法では、SGD, AdamW を使用した場合極小値から抜け出せずに学習が進まなかった。しかし他 4 つの手法は極小値から抜け出し、学習している様子が見られる。本研究で最も高い正解は畳み込みを導入したチューニング後の Adadelata で正解率が 33%となったが、ミニバッチ学習を導入した成果として単にチューニングを行ったものよりも分散が収まり、一定の正解率を維持できることが分かった。それでも正解率は 15%から 20%程度にとどまり、十分に予測ができていない結果となった。十分に予測ができていないというためには最低でも 50%の正解率が必要であると考えられる。しかしミニバッチ学習の導入により学習の安定性が向上したことは確認できた。

5. 結言

本研究では関口の研究[1]からパラメータチューニングやミニバッチ学習を導入した。パラメータチューニングでは精度の向上や学習速度の向上、ミニバッチ学習では学習の安定性が向上といった結果は見られたが、運用可能なレベルの精度には達しなかった。そのため、チューニングするハイパーパラメータや範囲を先例の研究だけを参考にするのではなく、新たに検証用データを作成して、MAE を学習データと検証用データの両方で算出することにより、チューニング範囲の決定をより問題に沿った形で決めることができ、汎化性能を高めることができる。また、Yahoo! ファイナンスからより長い年月のデータを取得するという方法もある。

文献

- [1] 関口晴紀: 深層学習を用いた日経平均株価の予測手法の改良~畳み込みの導入~, 2021 年度千葉工業大学卒業研究(2021)