Forex Forecasting

Combining statistical methods with neural networks

Eustathios Kotsis [1] Darmanis Michael [1] Vasilios Venieris [1]

September 26, 2023

[1]National and Kapodistrian University of Athens

Smoothed Convolutional Neural Network (S-CNN)

S-CNN: Origin and basic idea

- Implementation based on the paper "Time-series analysis with smoothed Convolutional Neural Network"[1] .
- Model is univariate, multi-step.
- Simple exponential smoothing (remove outliers, average) + CNN.

S-CNN: Simple Exponential Smoothing i

$$s_t = \alpha x_t + (1 - \alpha)s_{t-1} = s_{t-1} + \alpha(x_t - s_{t-1})$$

- For α values close to 1 we get very little influence of the previous smoothed value.
- For α close to 1 we also see that the previous unsmoothed series observation influences the result more than the smoothed values.
- For the calculation of α, since there is no golden rule, we adopted the α value from the aforementioned publication.

S-CNN: Simple Exponential Smoothing ii

- In particular the authors argue that α should be dependent on the dataset and must be of course between 0 and 1. Also the average value of the series should be less than the difference between max and min.

So we end up with the following formula

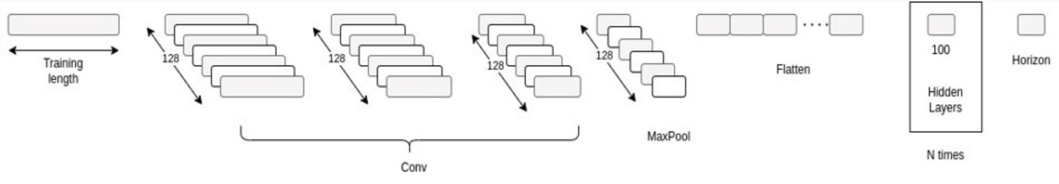
$$\alpha_{\text{Optimum}} = \frac{(X_{\text{max}} - X_{\text{min}}) - \frac{1}{n}\sum_{t=1}^{n} X_t}{X_{\text{max}} - X_{\text{min}}},$$

and the simple smoothing becomes

$$S_t = S_{t-1} + \frac{(X_{\text{max}} - X_{\text{min}}) - \frac{1}{n}\sum_{t=1}^{n} X_t}{X_{\text{max}} - X_{\text{min}}} \left(X_t - S_{t-1}\right).$$

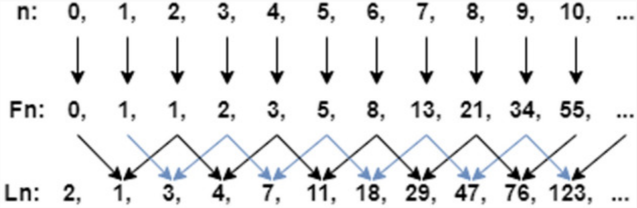
S-CNN: Architecture

Basic architecture from Wibawa et al.[1] and experimented with different number of hidden layers.



How to pick the number of hidden layers?

We used the Lucas numbers suggested.They derive from the Fibonacci Sequence if instead of adding every 2 successive observations we skip the intermediate and add the K with the K+2 together.



S-CNN: Training ii

Model yielded the lower MSE, for the most stable currencies, for 76 layers of training.

Table 1: Training time for different numbers of hidden layers

Number of Hidden Layers	Training Time (hh:mm:ss)
3	0:27:43.15
11	0:34:27.77
47	1:05:21.78
76	1:28:47.70

S-CNN: Training iii

Training steps...

- Split the dataset into series of different frequencies.
- For each time series we normalise + smooth.
- Feed the output into the convolutional model and train the model to predict a horizon of K steps for this frequency, where K is a model's designers choice.

S-CNN: Training iv

Training history.

In order to balance training with only recent data, or feeding with irrelevant outdated history, we train with a list of possible training lengths.

- Daily: [14, 20, 240]
- Weekly: [52, 13, 26]
- Monthly: [24, 18]
- Quarterly: [12, 8]
- Yearly: [6]

S-CNN: Training v

Train/validation/test split.

- We focused on data after 2010 since in 2008 – 2010 there was the European debt crisis. A plot from the data can show that the economy and the currency fell rapidly in most countries in the European continent and in many other countries.
- For the frequencies except Yearly, we trained with splits of 80%-20%, 70%-30%, and approximately 60%-40% for quarterly (training/validation).

S-CNN: Training vi

- We split the dataset accordingly and fed the created datasets to the generators.
- The models were trained for 250 epochs, using the MSE loss function and the Adam optimiser.

S-CNN: Results i

For stable currencies, longer training tends to yield more accurate predictions.

Table 2: sMAPE for Daily with 76 Hidden Layers (Biggest Lucas Number Tested)

	SMAPE	
	Training length 20 days	Training length 240 days
CAD	104.24	26.63
AUD	226.39	79.63
DKK	1.357	0.937

S-CNN: Results ii

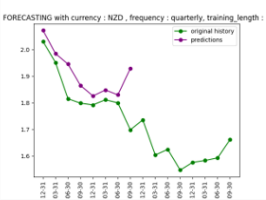
Unpredictable currencies benefit from short-term history and possibly higher α for forecasting. More hidden layers may overfit. Fewer hidden layers excel for daily predictions of unstable currencies.

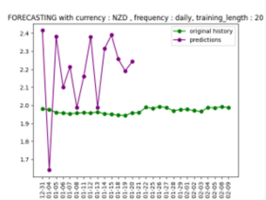
Table 3: sMAPE for 76 Hidden Layers

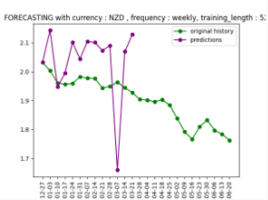
	sMAPE			
	T.L. 20 days	T.L. 240 days	T.L. 8 quarters	T.L. 12 quarters
USD	59.67	138.76	72.69	82.30

S-CNN: Results iii

Quarterly and monthly predictions outperform daily ones for stable currencies. Daily and weekly forecasts show many fluctuations and are less accurate.

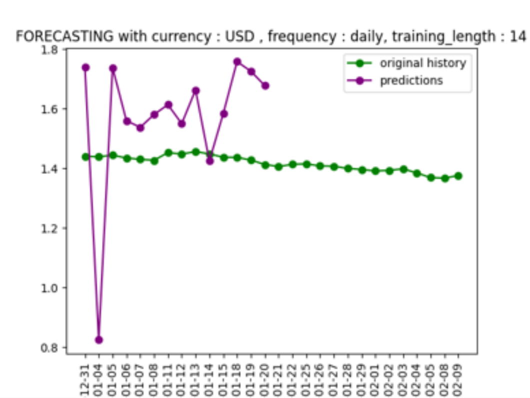


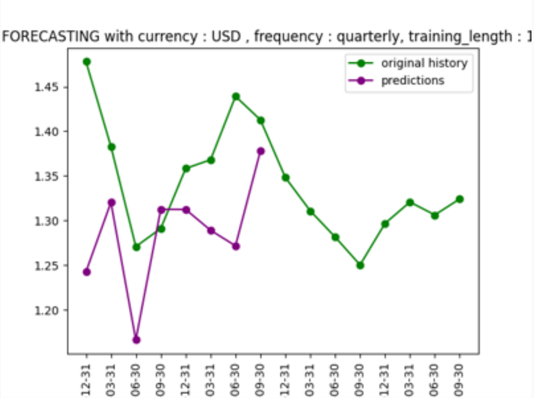




S-CNN: Results iv

Model struggles with unpredictable currencies like *USD* and *GBP*.





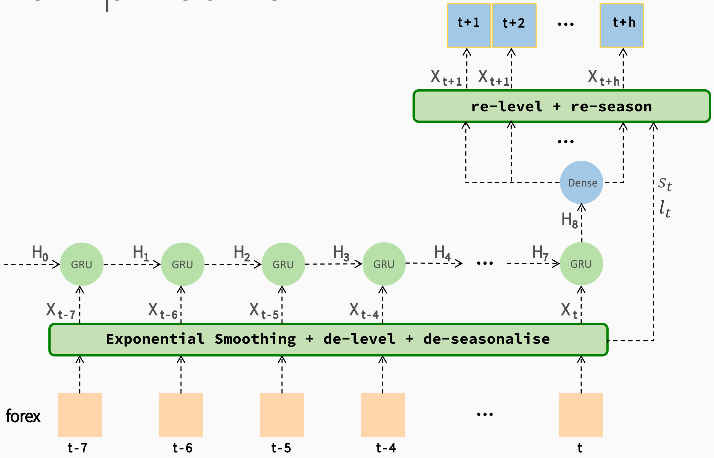
Exponentially Smoothed Recurrent Neural Network

ES-RNN: Origin and basic idea

ES-RNN: Holts-Winters model

ES-RNN: Architecture

Simplified ES-RNN



$$\hat{y}_{t+1|t} = l_t \cdot s_{t+1} \cdot RNN(x_t)$$

$$RNN(x_t)$$

Smoothing: $l_t = \alpha \frac{y_t}{s_t} + (1-\alpha) l_{t-1}$

Seasonal: $s_{t+m} = \gamma \frac{y_t}{l_t} + (1-\gamma) s_t$

De-seasonalised and de-leveled time series: $x_t = \frac{y_t}{l_t s_t}$

ES-RNN: Training i

ES-RNN: Results i

References i

A. Wibawa, A. Utama, H. Elmunsyah, et al.
Time-series analysis with smoothed convolutional neural network.
J Big Data, 9:44, 2022.