Comparison of traditional and neuronet methods in time series prediction

Sergei Raudik
Liliya Mironova

ISP 2020

Dataset - Avocado prices

Google

avocado dataset

Все | Картинки | Видео | Новости | Покупки | Ещё | Настройки | Инструменты

Результатов: примерно 245 000 (0,41 сек.)

www.kaggle.com › neuromusic › avocado-p... ▾ Перевести эту страницу

Avocado Prices | Kaggle

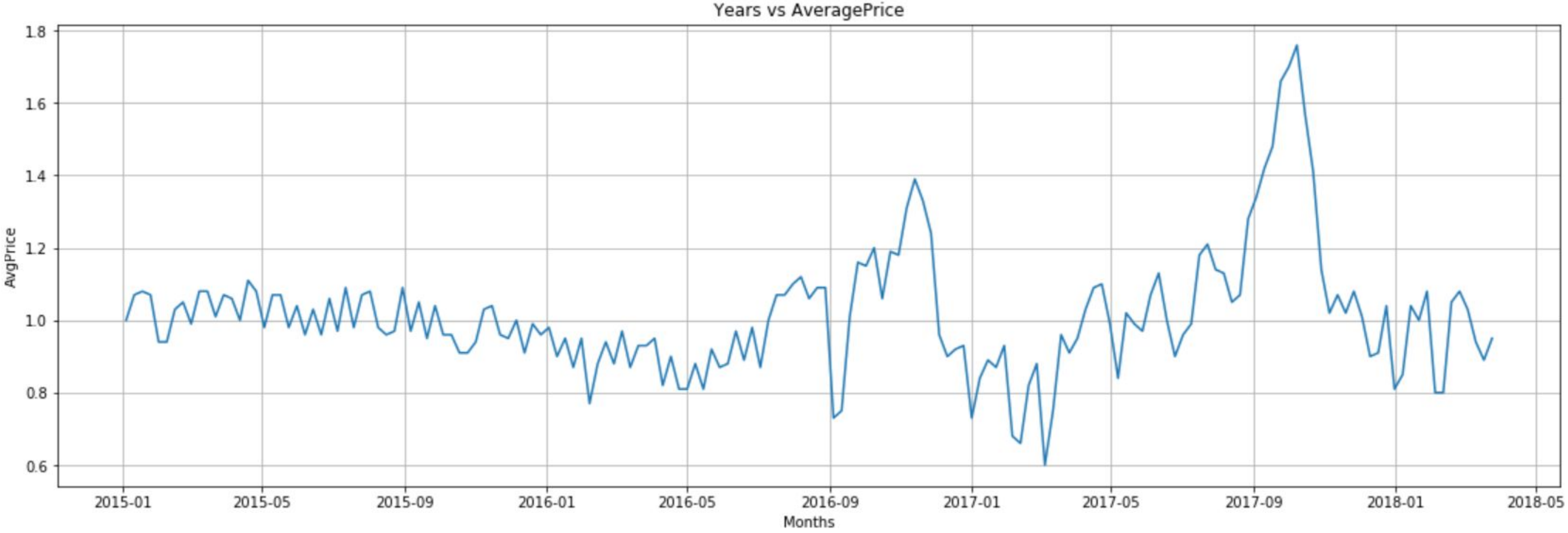
6 июн. 2018 г. - Historical data on **avocado** prices and sales volume in multiple US markets.

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
0	2015-12-27	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015	Albany
1	2015-12-20	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015	Albany
2	2015-12-13	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015	Albany

https://www.kaggle.com/neuromusic/avocado-prices

We considered the prices for the region “Nashville” and the “conventional” type of avocado and got the following plot:

Date	AveragePrice
2015-01-04	1.00
2015-01-11	1.07
2015-01-18	1.08
2015-01-25	1.07
2015-02-01	0.94



Motivation for choosing the dataset

We chose time series among other types of sequential data because we know that we will deal with this type of data in our future work (MSc thesis, Industrial Immersion)

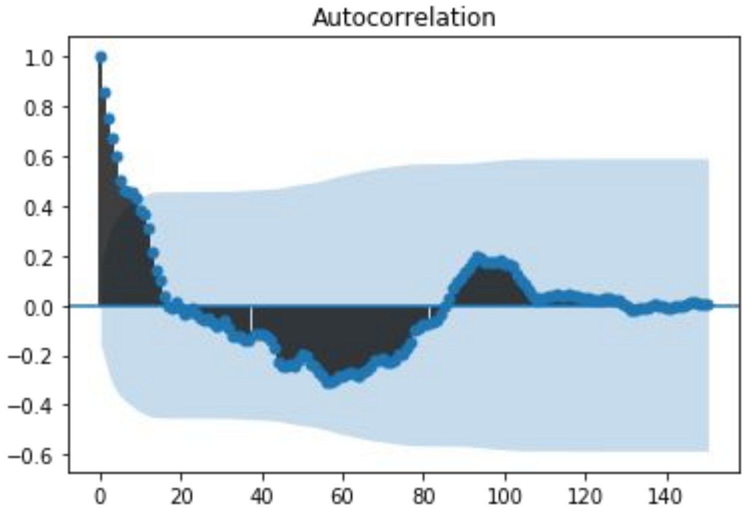
The choice of such an unusual dataset (Avocado prices) is based on the fact that it has a big set of features and requires different types of preprocessing (ex. aggregation under the conditions)

Exponential Smoothing: diagnostics

Stochastic trend:

KPSS p-values - 0.1

ADF p-values - 0.0008

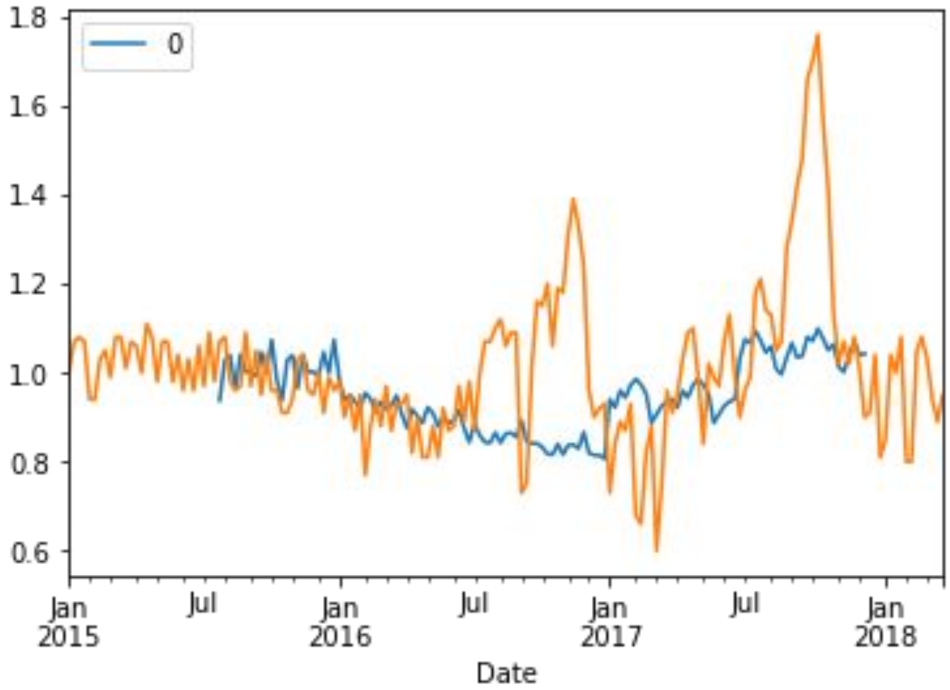


Exponential Smoothing: playing with the parameters

It was shown that additive trend and seasonality have better result than multiplicative.

Also it was shown that log-transformation can improve results but changes are very small.

This method showed good enough result on smooth data, but failed on sharp changes.

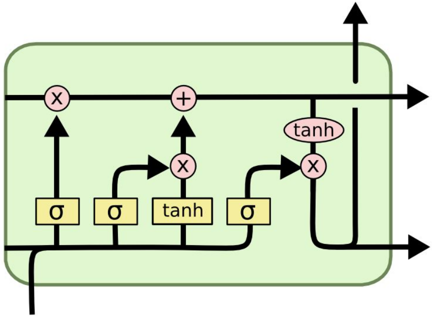


Train data and predicted data from cross-validation

Models we compared

Exponential smoothing vs **LSTM** (PyTorch)

$$F_{t+1} = \alpha D_t + (1-\alpha) F_t$$



Why: because we wanted to compare traditional and neural network methods in time series prediction.

LSTM: preprocessing

Feature scaling - normalization:

we subtracted the mean and our data became scaled to a fixed range (-0.41 to 0.74). The purpose of this it to end up with smaller standard deviations, which can prevent the effect of outliers. Normalisation is most commonly used in Deep Learning and image processing.

```
test_data_size = 15

all_data = data_df.values.reshape(-1)

train_mean = all_data[:-test_data_size].mean()
all_data_normalized = all_data - train_mean
all_data_normalized.min(), all_data_normalized.max()
```

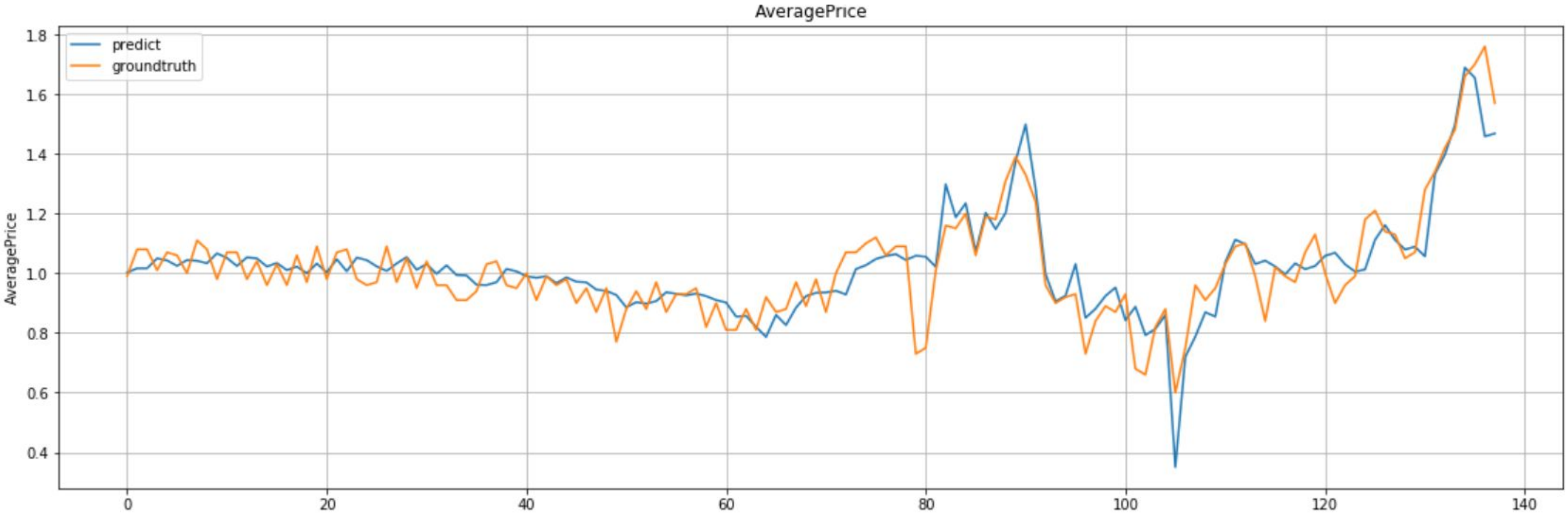
```
(-0.418896103896104, 0.741103896103896)
```

Choice of LSTM architecture

- **input_dim** = 1 - the number of input objects per time step
- **hidden_dim** = 150 - the size of the hidden state=the number of outputs that each LSTM cell produces at each time step
- **n_layers** = 3 - the number of hidden LSTM layers (hidden states in each cell) to use; typically a value between 1 and 3 *(got experimentally)*
- **epochs** = 300 - the number of training cycles *(the more epochs we have, the better trained the network becomes, and, accordingly, its result is better)*
- **lr** = 1e-3 - the learning rate=step *(the bigger the step is, the faster we learn, but here the loss can increase. The optimal one was got experimentally)*

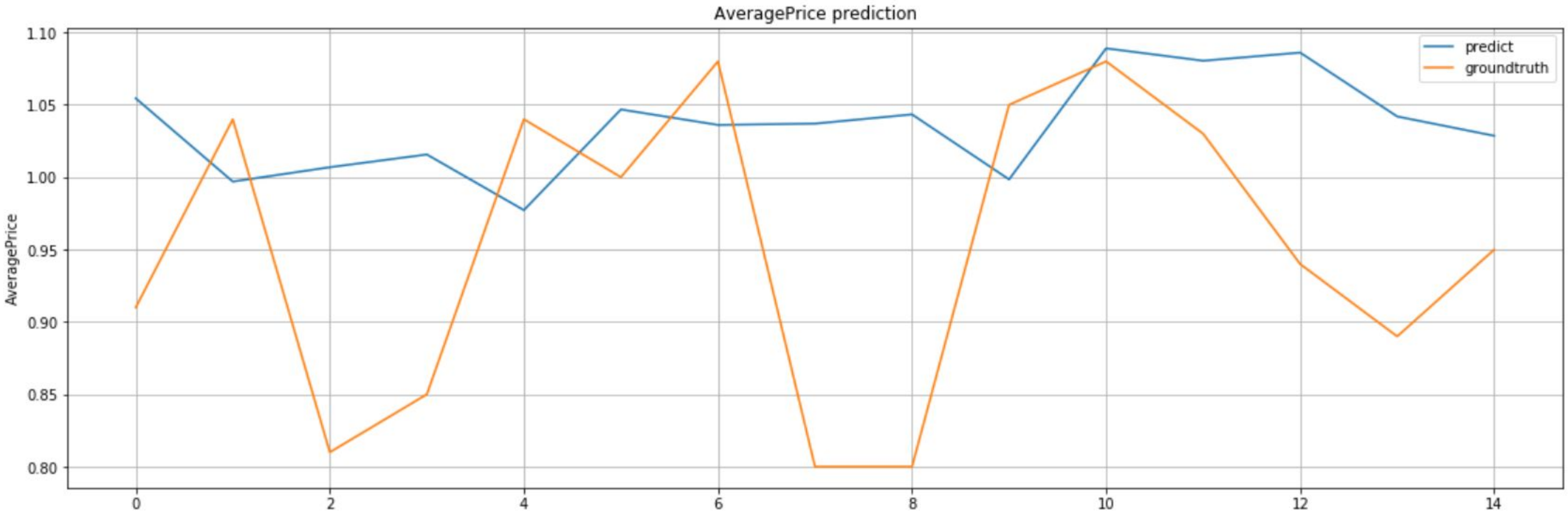
LSTM: results - training

Train loss: 0.0867 (RMSE)



LSTM: results - testing

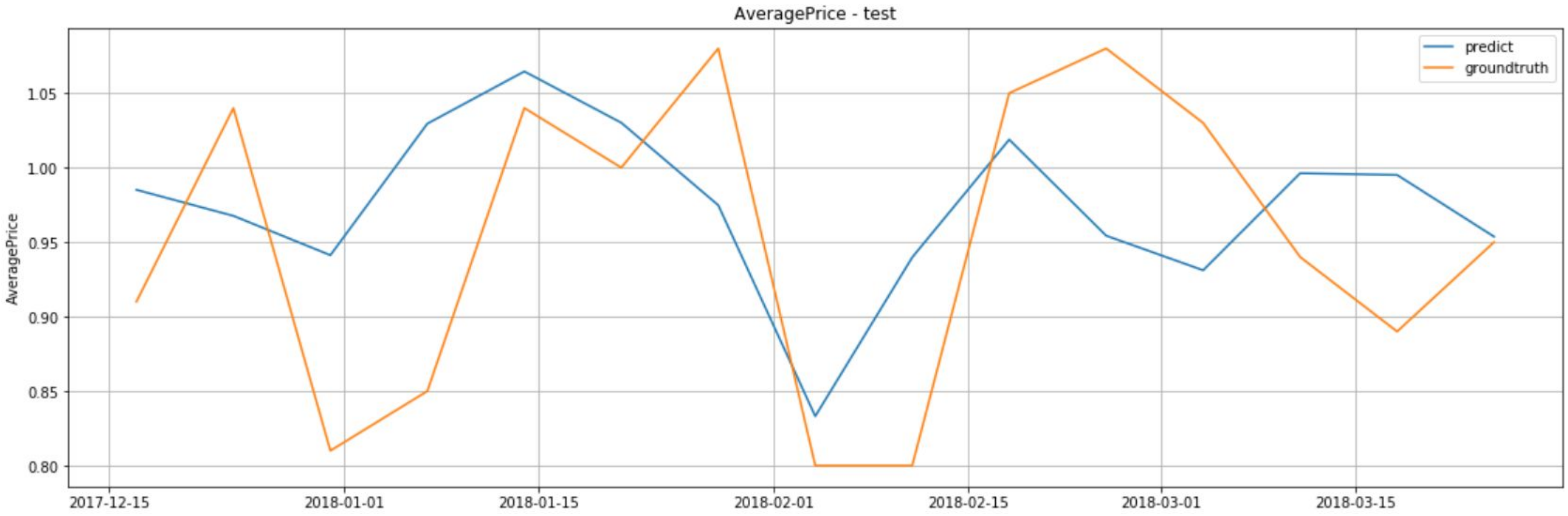
Test loss: 0.1337 (RMSE)



Exponential Smoothing: results

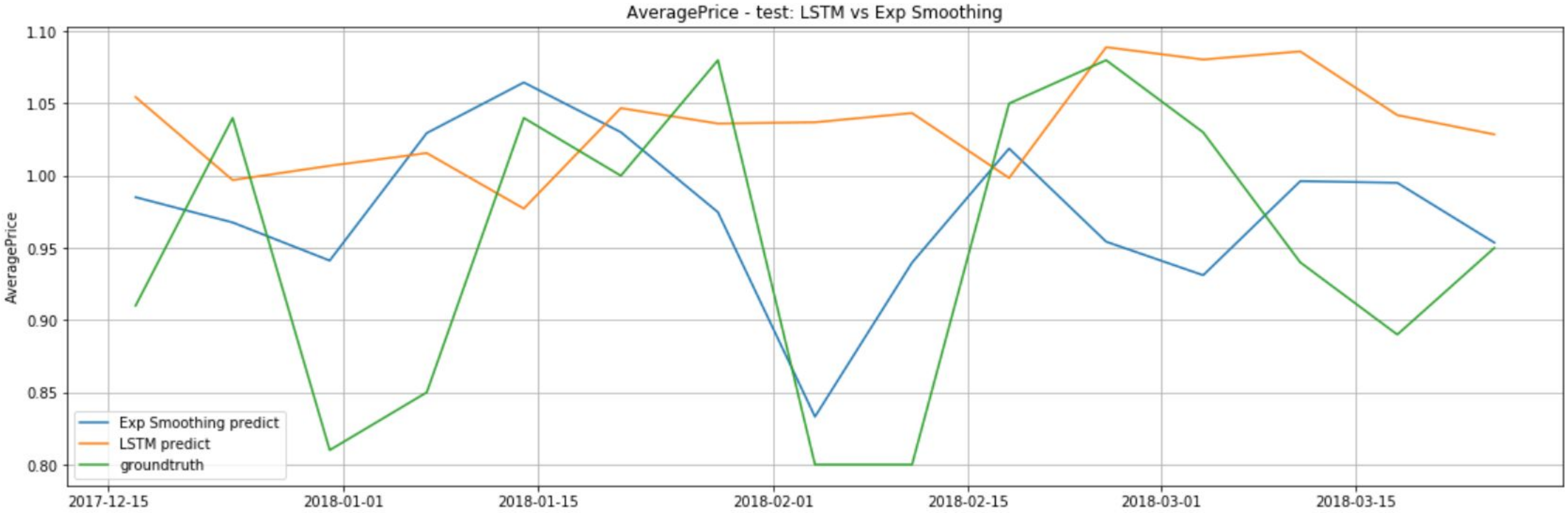
- Additive and multiplicative trends and seasonality were tested, but additive seasonality and trend showed better results.

Test loss: 0.0946



Conclusions

So, we compared two models on time series dataset: Exponential Smoothing and LSTM. The obtained results showed that <u>Exponential Smoothing method was more accurate</u>: 0.0946 vs 0.1337 loss (RMSE) on the test sample.



Our team

Sergei Raudik

Sergei.Raudik@skoltech.ru

Tested Exponential smoothing method

Liliya Mironova

Liliya.Mironova@skoltech.ru

Tested LSTM method

https://colab.research.google.com/drive/1sBkSlkJrdVjwqS-uC0dcu3d-C9J1hnJL

Thank you for your attention!