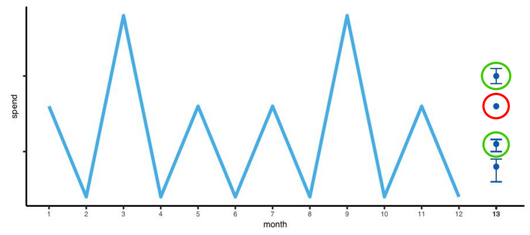
Evaluating uncertainty scores for deep regression networks in financial short time series forecasting

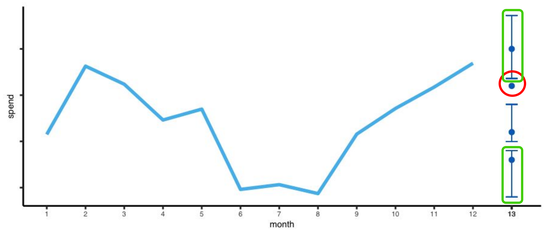
{mauricio.ciprian, leonardo.baldassini, luis.peinado, teresa.correas, roberto.maestre, joseantonio.rodriguez.serrano}@bbvadata.com
{oriol_pujol, jordi.vitria}@ub.edu

There's no such thing as a certain prediction

Time series exhibiting a pattern



Random time series



In the general context of prediction problems, some methods only produce pointwise estimates instead of a more desirable uncertainty score.

What if the best prediction accuracy comes from pointwise methods?

Why using deep networks on financial short time series?

Because other methods won't work as well.

In the analysis of personal financial transactions, monthly aggregation offers the right granularity tradeoff to compare frequent and relatively noisy expenses (e.g. grocery shopping) with more sporadic or periodic ones (e.g. theatre tickets, utilities).

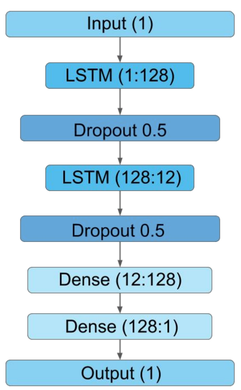
Furthermore, it is often the case that business constraints dictate the need for short histories, for example to prevent recent clients from being excluded from a study.

In this scenario of short-and-coarse time series, classic prediction methods such as Holt-Winters or ARIMA tend to produce poor results **[6]**.

In order to produce reliable predictions out of only 12-point series, we have adopted a ML approach developing of a deep-network based approach, whose output is a single point estimate.

While it outperformed the alternative methods we evaluated, how sure could we be of its predictions?

Network architecture and prediction results



	MARE	MASE	RMSE	u10%	u20%
Mean	570	748	608.2	18.0	29.2
Last	707	867	648.1	33.6	40.2
Zero	952	2.247	742.9	2.0	5.2
1-NN	1.095	1.057	13337.1	36.4	46.1
100-NN	806	748	871.5	29.6	42.1
Act 1-NN	935	925	6764.9	**35.7**	45.2
Act 100-NN	823	785	1069.9	28.7	41.1
Random Forest	844	783	828.4	29.1	40.2
LSTM	**521**	**586**	**552.4**	29.9	**47.2**

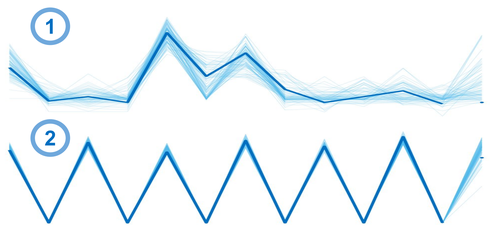
In order to carry out prediction on short-and-coarse time series, we included LSTM layers in our network with the aim of capturing as much historical information as possible. **[3, 7]**

We evaluated performance against standard the following metrics:
MARE: Mean Absolute Relative Error
MASE: Mean Absolute Scaled Error **[5]**
RMSE: Root Mean Squared Error
uP%: Percentage of series below the P-th percentile in the MASE range.

LSTM-based regression outperforms alternative methods in 4 out of 5 metrics.

Confidence calculation: Proximity-based methods

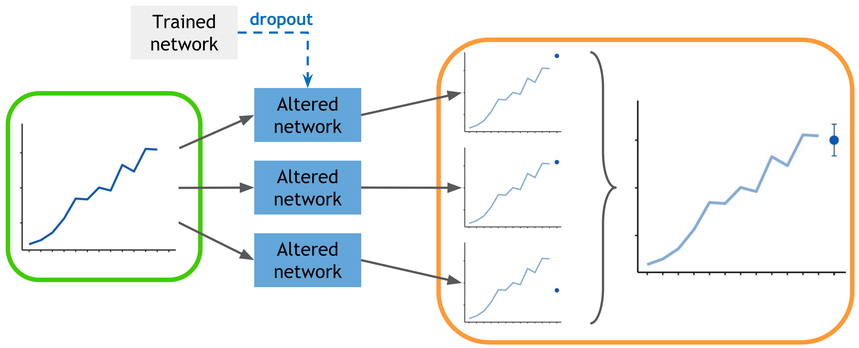
The image shows two series and their 100 most immediate neighbours. Series like 2, with many similar neighbours, highlight a frequent pattern in the data, thus allowing for more accurate predictions. We tried the following approaches:



NN density: Euclidean distance to the nearest neighbour in the training set.
NN density (emb): Likewise, but in the space of embeddings generated by the last fully-connected layer of the network evaluated on the series.
RF regression: A random forest that learns the prediction error of the LSTM, with cross validation over the training set and feature preprocessing via z-scores and PCA.
autoenc: Marginalised denoising autoencoder **[1]**. Frequently seen patterns yield small reconstruction errors.

Confidence calculation: Bootstrapping over regression networks

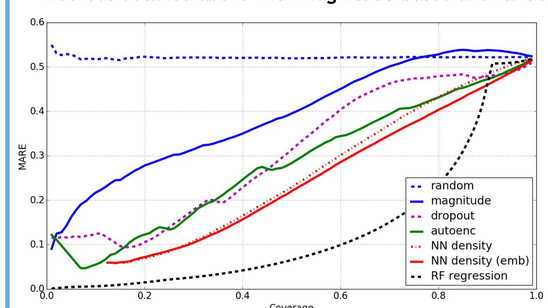
Recent research has highlighted how Gaussian processes and variational inference provide a theoretical framework for deep learning **[2]**, within which dropout has been related to the estimation of prediction uncertainty. In our method input series are fed to versions of the trained network where dropout has been applied. The predictions of the networks are then combined in 95% confidence intervals.



The rationale behind this approach is that predictions that are robuster to the noise introduced in networks produce tighter intervals.

Confidence calculation: Benchmarking

The benchmark of confidence calculation methods uses *MARE vs coverage* (fraction of non-rejected samples) curves: Methods producing better estimates on the confidence of predictions keep the MARE low for larger proportions of the test set. We compare the methods detailed above with **magnitude**-based and **random** prediction rejection baselines.



Perhaps unsurprisingly, given its supervised nature, random forest regression achieves the best confidence estimation. Among unsupervised methods, the best estimator we evaluated is the distance to the *n*-th neighbour in the space of embeddings produced by the last dense layer of the regression network.

References

[1] Chen, Xu, Weinberger, and Sha. *Marginalized denoising autoencoders for domain adaptation*. In ICML, 2012.
[2] Gal and Ghahramani. *Dropout as a Bayesian approximation: Representing model uncertainty in deep learning*. In ICML, 2016.
[3] Gers, Schmidhuber, and Cummins. *Learning to forget: Continual prediction with LSTM*. Neural Comput., 12(10), 2000.
[5] Hyndman and Koehler. *Another look at measures of forecast accuracy*. International Journal of Forecasting, pages 679-688, 2006.
[6] Hyndman and Kostenko. *Minimum sample size requirements for seasonal forecasting models*. Foresight, 6(Spring):12-15, 2007.
[7] Lipton, Berkowitz, and Elkan. *A critical review of recurrent neural networks for sequence learning*. arXiv preprint arXiv:1506.00019, 2015.

BBVA DATA & ANALYTICS

UNIVERSITAT DE BARCELONA