Detection of Bank Liquidity Stress using Recurrent Neural Networks

Ron Triepels[1] and Richard Heuver[2]

1 September, 2023

[1]Maastricht University, the Netherlands
[2]De Nederlandsche Bank, the Netherlands

Disclaimer

The author(s) of this paper is(are) member(s)/alternate(s) of one of the user groups with access to TARGET data in accordance with Article 3 of the Decision (EU) 2023/549 of the European Central Bank of 6 March 2023 on access to and use of certain TARGET data and repealing Decision ECB/2010/9 (ECB/2023/3). The Central Bank(s) of Author(s) and the MIB have checked the paper against the rules for guaranteeing the confidentiality of transaction-level data imposed by the MIB pursuant to Article 5 of the above-mentioned issue. The views expressed in the paper are solely those of the author(s) and do not necessarily represent the views of the Eurosystem.

Liquidity Stress Detection

Liquidity stress:

- Experienced by banks when faced with an unexpected liquidity shock.
- Short-term payment obligations can no longer be met.
- Can have severe consequences (e.g., a bank run, takeover, ...).

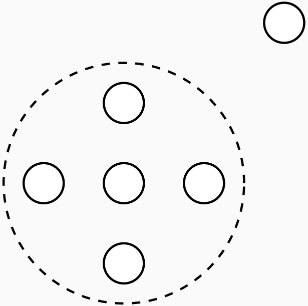
Data-driven stress detection:

- Financial market infrastructures generate a wealth of data.
- Analyze historic payment behavior using machine learning.
- Determine whether a bank likely faces liquidity stress.

Unsupervised Learning

General approach:

- Build a model from unlabeled data to learn the characteristics of normal payment behavior.
- Apply the model on new data and search for cases in which the model fails to describe behavior.



Type of models:

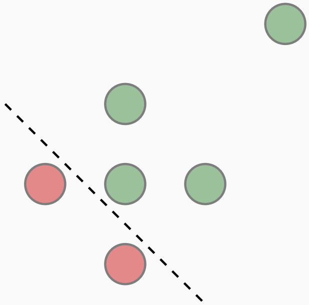
- Autoencoders [Triepels et al., 2017]
- Principal Component Analysis [León, 2020]

Drawback: it is hard to infer what these models learn. In addition, they potentially detect anomalies that are not the result of liquidity stress but noise.

Supervised Learning

General approach:

- Build a model from labeled data to learn the characteristics of a stressed and non-stressed bank.
- Apply the model on new data and identify stress periods.



Previous work:

- Liquidity stress detection [Heuver and Triepels, 2019]

Drawback: it is difficult to obtain high-quality labels of liquidity stress.

Research Question

Our work is an extension of the work by [Heuver and Triepels, 2019].

We use Recurrent Neural Networks:

- Operate on sequences of data.
- Learn patterns of a stressed and non-stressed bank over time.

Improved data:

- Based on data of 3 additional banks (10 in total).
- Better stress labels by applying active learning.

Do these improvements result in beter liquidity stress detection?

Problem Definition

Let $\mathcal{T} =< t_1, t_2, \cdots >$ be an ordered set of time intervals (i.e., days).

For each bank j, we have a sequence of vectors and labels:

- Feature vector $\boldsymbol{x}_j^{(i)}$ describes the payment behavior of the bank at t_i.
- Label $y_j^{(i)} \in \{0, 1\}$ indicates whether the bank faced liquidity stress at t_i.

Our goal is to build a model that estimates the probability:

$$P\left(y_j^{(i)} = 1 \middle| \boldsymbol{x}_j^{(i)}, \boldsymbol{x}_j^{(i-1)}, \dots\right) \quad (1)$$

Multi-Layer Perceptron (MLP)

Used in [Heuver and Triepels, 2019] which served as our benchmark model.

Each $\boldsymbol{x}_j^{(i)}$ is processed through a hidden layer with rectified units:

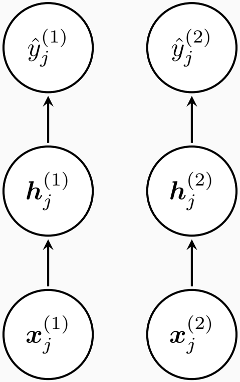
$$\boldsymbol{h}_j^{(i)} = f\big(\boldsymbol{W}_h \boldsymbol{x}_j^{(i)} + \boldsymbol{b}_h\big) \tag{2}$$

The output of the network is:

$$\hat{y}_j^{(i)} = \sigma\big(\boldsymbol{w}_o \boldsymbol{h}_j^{(i)} + b_o\big) \tag{3}$$

This is an estimate of:

$$P\big(y_j^{(i)} = 1 \big| \boldsymbol{x}_j^{(i)}\big) \tag{4}$$



Has a similar architecture as the MLP network but processes each $\boldsymbol{x}_j^{(i)}$ though a recurrent layer that maintains a hidden state over time.

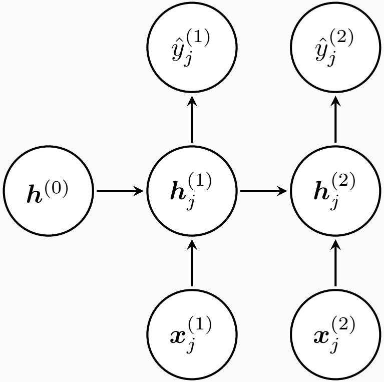
The output of the recurrent layer is:

$$\boldsymbol{h}_j^{(i)} = g\big(\boldsymbol{h}_j^{(i-1)}, \boldsymbol{x}_j^{(i)}\big) \qquad \text{for } i > 0 \tag{5}$$

The layer consists of a number of LSTM units [Hochreiter and Schmidhuber, 1997].

The output $\hat{y}_j^{(i)}$ is an estimate of:

$$P\big(y_j^{(i)} = 1 \big| \boldsymbol{x}_j^{(i)}, \boldsymbol{x}_j^{(i-1)}, \dots\big) \tag{6}$$



Dataset

Key properties:

- Includes 10 banks that are known to have faced liquidity stress.
- Includes 76 features that describe payment behavior.
- Data is aggregated on a daily basis.
- Spans over 14 years.

Compiled from three systems of the Eurosystem:

- TARGET2 (T2)
- Collateral Management System (CMS)
- Minimum Reserve System (MRS)

Data Sources and Features

Type	Feature	Source
Account Balance	• End-of-day account balance • Minimum account balance	T2
Payments	• Total net value of payments • Total net number of transactions • Net payment time within the day weighted by value • Net payment time within the day weighted by the number of transactions	T2
Money Market	• Total number of counterparties • HHI-index [Hirschman, 1945] of the number of money market counterparties • Difference between average interest rate and EONIA weighted by loan value	T2
Collateral	• Average haircut applied to the total collateral value • Total value of collateral before the haircut • Total value of collateral after the haircut	CMS
Minimum Reserve	• Difference between end-of-day balance and minimum reserve requirement	MRS

Stress Codes

For each bank, we searched online on Wikipedia and several national and international financial newspapers for evidence of liquidity stress.

We assigned a stress code to each day:

1. **No stress**: if no evidence of liquidity stress at the bank could be found.
2. **Possibly stress**: if we could find some evidence of liquidity stress at the bank but which was not that severe.
3. **Stress**: if we could find clear evidence of liquidity stress at the bank.
4. **Bankrupt**: if the bank is bankrupt or taken over by another institution.

Stress Labels

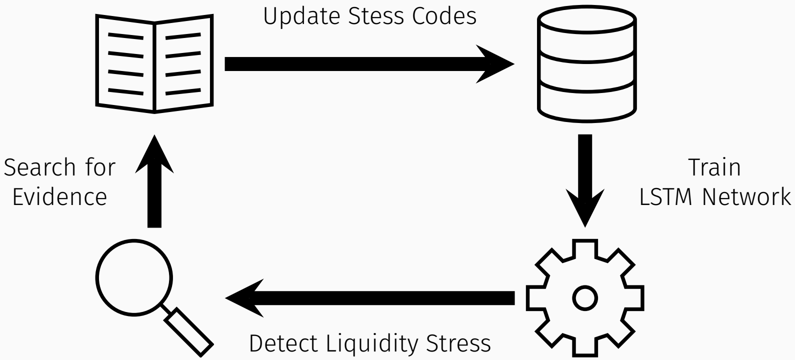
Accordingly, we labeled the data as follows:

$$y_j^{(i)} = \begin{cases} 1, & \text{if the stress code is 3 (stress)} \\ 0, & \text{if the stress code is 1 (no stress)} \end{cases} \tag{7}$$

Note that:

- Days at which the stress code was 2 (possibly stress) were not labeled and only used for out-of-sample prediction.
- Days at which the stress code was 4 (bankrupt) were removed.

We used a form of active learning to narrow down the possibly stress periods.



Nested Cross Validation

	Bank A	Bank B	...	Bank I	Bank J
Itr. 1	Test	Val	...	Train	Train
Itr. 2	Val	Test	...	Train	Train
⋮	⋮	⋮	⋮	⋮	⋮
Itr. 9	Train	Train	...	Test	Train
Itr. 10	Train	Train	...	Val	Test

Results

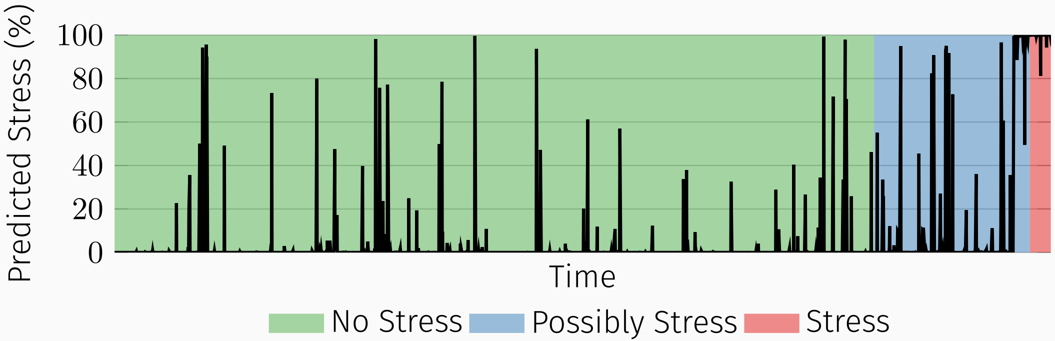
Key observations:

- The LSTM network detects liquidity stress better than the MLP network.
- It learns patterns of liquidity stress that generalize well to banks whose data the network has not seen before.
- It detects liquidity stress quite some time before the stress became known to the general public.

	Cross Entropy*	
Bank	MLP	LSTM
A	**0.020499**	**0.000022**
B	0.095966	0.085779
C	**0.045841**	**0.000058**
D	**0.000348**	**0.000008**
E	**0.028470**	**0.000018**
F	**0.535325**	**0.028626**
G	0.021893	0.045540
H	0.000370	0.071381
I	0.140433	0.523239
J	**0.014404**	**0.000011**
Mean	0.090355	0.075468
SD	0.162577	0.160614

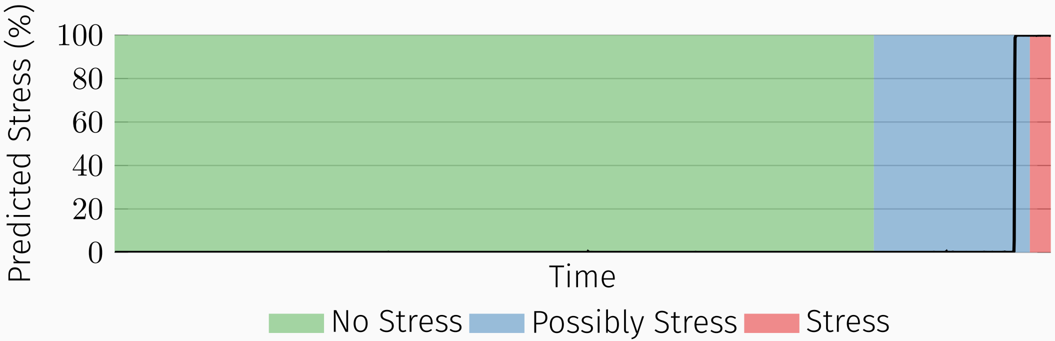
*A lower score indicates better performance.

Estimates of Bank E by the MLP Network



The background colors represent the stress codes obtained from the news analysis. The network has not seen this bank before and does not have access to the stress codes.

Estimates of Bank E by the LSTM Network



The background colors represent the stress codes obtained from the news analysis. The network has not seen this bank before and does not have access to the stress codes.

Results (Cont.)

Key observations:

- The MLP network performed better than the LSTM network on banks G and H.
- These banks had very short and abrupt periods of stress.

	Cross Entropy*	
Bank	MLP	LSTM
A	0.020499	0.000022
B	0.095966	0.085779
C	0.045841	0.000058
D	0.000348	0.000008
E	0.028470	0.000018
F	0.535325	0.028626
G	**0.021893**	**0.045540**
H	**0.000370**	**0.071381**
I	0.140433	0.523239
J	0.014404	0.000011
Mean	0.090355	0.075468
SD	0.162577	0.160614

*A lower score indicates better performance.

Results (Cont.)

Key observations:

- The liquidity stress of banks B and I could not be reliably detected.
- We suspect the quality of stress labels of these banks is insufficient.
- Or, the stress is different from the stress faced by other banks in the dataset.

	Cross Entropy*	
Bank	MLP	LSTM
A	0.020499	0.000022
B	**0.095966**	**0.085779**
C	0.045841	0.000058
D	0.000348	0.000008
E	0.028470	0.000018
F	0.535325	0.028626
G	0.021893	0.045540
H	0.000370	0.071381
I	**0.140433**	**0.523239**
J	0.014404	0.000011
Mean	0.090355	0.075468
SD	0.162577	0.160614

*A lower score indicates better performance.

Conclusions

An LSTM network seems well suited to detect liquidity stress.

Our approach could be usefull to detect liquidity stress at an early stage.

Future work:

- Expand dataset with more (healthy and non-healthy) banks.
- Improve quality of stress labels by natural language processing.
- Generate explanations for stress predictions.

References

- Heuver, R. and Triepels, R. (2019).
  Liquidity stress detection in the european banking sector.
  In Proceedings of the 11th International Conference on Agents and Artificial Intelligence-Volume 2, pages 266–274. Scitepress-Science And Technology Publications.
- Hirschman, A. O. (1945).
  National power and the structure of foreign trade.
  University of California Press.
- Hochreiter, S. and Schmidhuber, J. (1997).
  Long short-term memory.
  Neural computation, 9(8):1735–1780.

References (Cont.)

León, C. (2020).
Detecting anomalous payments networks: A dimensionality-reduction approach.
Latin American Journal of Central Banking, 1(1-4):100001.

Triepels, R., Daniels, H., and Heijmans, R. (2017).
Anomaly detection in real-time gross settlement systems.
In International Conference on Enterprise Information Systems, volume 2, pages 433–441. SCITEPRESS.

Questions?

Cross Entropy vs AUPRC

Cross Entropy:

- The average distance between each estimated $\hat{y}_j^{(i)}$ and target $y_j^{(i)}$.

AUPRC:

- Precision of a model averaged over all possible decision thresholds.

Liquidity stress is so rare that the AUPRC yields biased results for some banks.

	Cross Entropy		AUPRC	
Bank	MLP	LSTM	MLP	LSTM
A	0.020499	0.000022	0.84	1.00
B	0.095966	0.085779	0.72	0.72
C	0.045841	0.000058	0.93	1.00
D	0.000348	0.000008	1.00	1.00
E	**0.028470**	**0.000018**	**0.99**	**1.00**
F	0.535325	0.028626	0.05	0.16
G	0.021893	0.045540	0.64	0.40
H	0.000370	0.071381	1.00	0.30
I	0.140433	0.523239	0.08	0.02
J	0.014404	0.000011	0.66	1.00
Mean	0.090355	0.075468	0.69	0.66
SD	0.162577	0.160614	0.36	0.40