High-Performance Ensembles of Online Sequential Extreme Learning Machine for Regression and Time Series Forecasting

Luís Fernando L. Grim[1][2] and André L. S. Gradvohl[1]

lgrim@ifsp.edu.br
gradvohl@ft.unicamp.br

UNICAMP

INSTITUTO FEDERAL
São Paulo

[1]School of Technology – University of Campinas
[2]Federal Institute of São Paulo at Piracicaba
High Performance Inteligent Decision Systems Group

License and Contact Details

- Contact the author at

gradvohl@ft.unicamp.br

Agenda

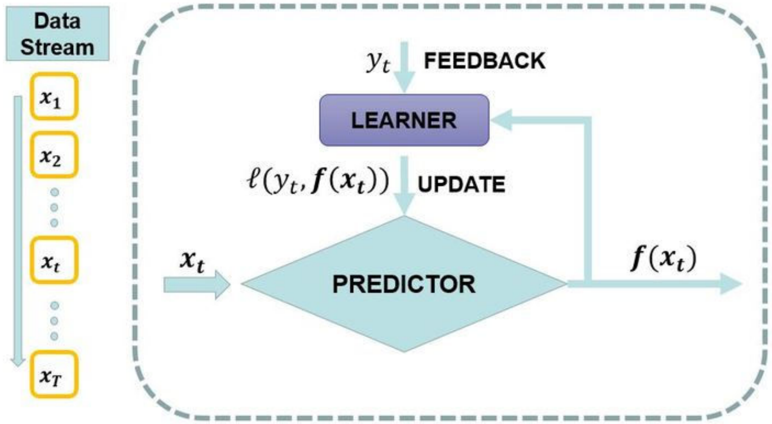
- Introduction;
- Extreme Learning Machines;
- Related Works;
- Implemented Approaches:
  - OS-ELM with Sliding Windows;
  - Ensembles of OS-ELMsw;
  - High-performance versus Serial ensembles;
  - Transfers and syncs in High-performance ensembles;
- Experiments Design:
  - The datasets and preprocessing steps;
  - Execution Environment;
  - Initial Configurations;
- Experimental Results:
  - RMSEs and Total Real Time of execution for the single model OS-ELMsw;
  - RMSEs and Total Real Time of execution for the Ensembles;
- Conclusions

Introduction

- Data Mining and Machine Learning:
  - Find implicit relationships in large datasets.
- Time Series forecasting:
  - Datasets can be considered as Data Streams;
  - and they may present Concept Drifts.
- Forecasting of Data Streams with Concept Drifts:
  - Time and modeling constraints.

Introduction

- Online learners:



Source: http://www.doyensahoo.com/introduction.html

- Proposal:
  - ✓ High-Performance Computing in Ensembles of Online Sequential Extreme Learning Machine;
- Objective:
  - ✓ Make improvements in execution time using High-Performance Computing techniques.

Extreme Learning Machine (ELM)

$$f_L(\mathbf{x}_t) = \sum_{j=1}^{L} \beta_j g(\mathbf{a}_j \cdot \mathbf{x}_t + b_j) = \mathbf{y}_t \qquad \text{for } t = 1, \ldots, T.$$

$$\mathbf{H}\beta = \mathbf{Y},$$

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{a}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(\mathbf{a}_1 \cdot \mathbf{x}_T + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_T + b_L) \end{bmatrix}_{T \times L},$$

$$\beta = \left[\beta_1^T, \cdots, \beta_L^T\right]_L \quad e \quad \mathbf{Y} = \left[\mathbf{y}_1^T, \cdots, \mathbf{y}_T^T\right]_T,$$

$$\beta = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{Y}$$

Online Sequential Extreme Learning Machine (OS-ELM)

- Initial training:

$$\mathbf{M}_0 = (\mathbf{H}_0^T\mathbf{H}_0)^{-1}$$

$$\beta_0 = (\mathbf{H}_0^T\mathbf{H}_0)^{-1}\mathbf{H}_0^T\mathbf{Y}_0$$

$$\mathbf{H}_0 = \begin{bmatrix} g(\mathbf{a}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(\mathbf{a}_1 \cdot \mathbf{x}_{N_0} + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_{N_0} + b_L) \end{bmatrix}_{N_0 \times L},$$

$$\beta_0 = \left[\beta_1^T, \cdots, \beta_L^T\right]_L \quad e \quad \mathbf{Y}_0 = \left[\mathbf{y}_1^T, \cdots, \mathbf{y}_T^T\right]_{N_0},$$

- Online step:

$$\mathbf{H}_{k+1} = \begin{bmatrix} g(\mathbf{a}_1 \cdot \mathbf{x}_{(\sum_{l=0}^{k} T_l)+1} + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_{(\sum_{l=0}^{k} T_l)+1} + b_L) \\ \vdots & \ddots & \vdots \\ g(\mathbf{a}_1 \cdot \mathbf{x}_{\sum_{l=0}^{k+1} T_l} + b_1) & \cdots & (\mathbf{a}_L \cdot \mathbf{x}_{\sum_{l=0}^{k+1} T_l} + b_L) \end{bmatrix}_{T_{k+1} \times L},$$

$$\mathbf{Y}_{k+1} = \left[\mathbf{y}^T_{(\sum_{l=0}^{k} T_l)+1}, \cdots, \mathbf{y}^T_{\sum_{l=0}^{k+1} T_l}\right]_{T_{k+1}},$$

$$\mathbf{M}_{k+1} = \mathbf{M}_k - \mathbf{H}_{k+1}^T(1 + \mathbf{H}_{k+1}\mathbf{M}_k\mathbf{H}_{k+1}^T)^{-1}\mathbf{H}_{k+1}\mathbf{M}_k,$$

$$\beta_{k+1} = \beta_k + \mathbf{M}_{k+1}\mathbf{H}_{k+1}^T(\mathbf{Y}_{k+1} - \mathbf{H}_{k+1}\beta_k).$$

Related Works

- Parallel Ensemble of GPUs' accelerated ELMs (Van Heeswijk et al., 2011):
  - Performed up to 3.3-fold faster than the conventional ELMs' Ensemble when using between 100 and 1,000 hidden neurons with the sigmoid function;
- GPU-accelerated OS-ELM (Krawczyk, 2016):
  - Performed up to 10-fold faster with 10 to 100 hidden neurons and update block of 2,500 samples.

Related Works

- Ensemble of Online Learners with Substitution of Models (EOS – Bueno; Coelho; Bertini, 2017):
  - Presented smaller and more stable RMSEs than static ensembles and single models in Particulate Matter (PM_{10}) datasets;
- Dynamic and On-line Ensemble Regression (DOER – Soares; Araújo, 2015):
  - Presented better results than other dynamic ensembles and single models in scenarios with Concept Drifts.

Implemented Approaches

- OS-ELM with Sliding Windows (OS-ELMsw) restricted to regression cases with sigmoid function $g(x) = \frac{1}{(1+e^{(-x)})}$:
  - Developed in C programming Language with Intel MKL library;
  - Used as models in all ensembles in this work;

- Ensemble of Online Sequential Extreme Learning Machine (EOS-ELM – Lan; Soh; Huang, 2009):
  - It is a static ensemble: Uses several OS-ELMsw models with the same parameters;
  - The randomly generated input parameters make each OS-ELM a distinct model;

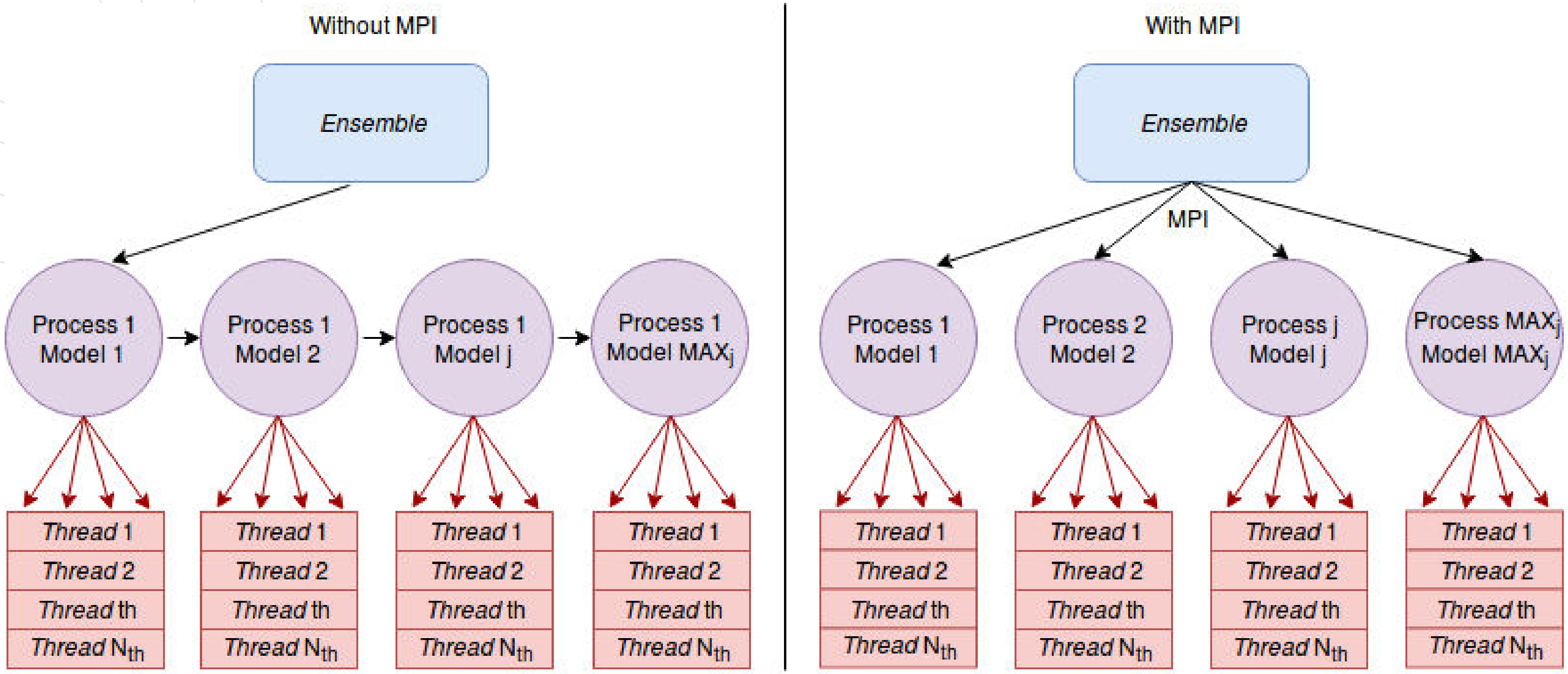
Implemented Approaches

- Dynamic and On-line Ensemble Regression (DOER):
  - Dynamic inclusion/replacement of models and adaptation of models' weights;
  - Modifications adopted in this work, called "Simplified C DOER" (SCDOER):
    - Operates also with sample blocks and calculates the Mean Squared Error (MSE) for each model in a simplified way;
    - Considers a window size equal to N_0 for the initial training in process of inclusion or replacement of models.

Implemented Approaches

- *Ensemble of Online Learners with Substitution of Models* (EOS):
  - In this work behaves similar to SCDOER, with the following exceptions:
    - Uses arithmetic mean, as in EOS-ELM;
    - Uses a fixed rate (proposed by Street and Kim (2001)) equal to N_0 to include or replace a model.

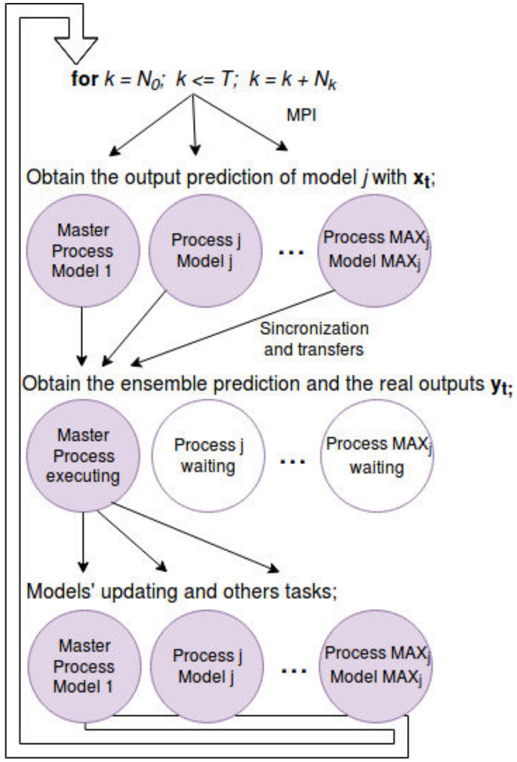
High-performance ensembles versus serial ensembles



Thread-level parallelism with Intel MKL and OpenMP

Transfers and synchronizations in High-performance ensembles

Ensembles	Data transfers	Master-process tasks
All	Models' prediction output	Make the prediction of the ensemble
EOS and SCDOER	MSE of each model; The worst model located.	Locate the worst model when required.
SCDOER	Weight of each model; Abs. rel. error of the ensemble; Median MSE of the ensemble; Sum of models' weights.	Calculate the median; MSE of the ensemble; Calculate the abs. rel. error of the ensemble; Calculate the sum of models' weights.



Experiments Design: The datasets

- Synthetic: Hyperplane (HYP):
  - Known Benchmark of Data Stream with Concept Drifts;
  - Data Stream with 175 thousand instances;
  - Four distinct concepts with 1/4 of the samples, 10 input variables and an output variable on [0:1] interval.
  - No preprocessing steps were required;
- Real:
  - Hourly samples of Inhalable Particles (PM10) in the air, between 01/01/1998 and 11/23/2017 provided by QUALAR:
  - Automatic station Cubatão - Vila Parisi: Total of 174,397 samples;
  - Min. value: zero;
  - Max. value: 1,470 $\mu m/m^3$;
  - 8,011 missing samples.

Preprocessing steps in the real dataset

- Inclusion of missing samples:
  - The most probable value with the Amelia II software and linear interpolation;
- Outliers analysis:
  - Pruning values above 350 µm/m^3; Hampel filter on the remaining outliers;
- Data normalization: min-max between [0,1];
- Organization of samples in time series instances:
  - Inputs → Current instant plus the last five: $x_n = [s_{n-5} \ldots, s_{n-1}, s_n]$
  - Outputs (target variable) → Next instant: $y = [s_{n+1}]$

Execution environment

- 25 nodes Quanta QSSC-S4R with the following configurations:
  - 4 Intel Xeon E7-4870 @ 2.40GHz CPUs with 20 cores;
  - 1024 GiB of RAM; 8 SSD disks with 450 GiB;
  - 2 HBAs Intel Corporation 82599EB 10-Gigabit.
- Head Node: SUSE Linux Enterprise Server 11 SP4.
- Other nodes: Ubuntu 16.04.3 LTS;
- All nodes have the Slurm 17.02.6 Workload Manager, Intel MKL 2018.2.199 and MPICH 3.2 libraries installed.

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Initial Configurations

- OS-ELMsw:
  - Operating with 100 hidden neurons;
  - And with updating the model with each new instance;
- Initial training set: 5,000 instances;
- Simulated stream with approximately 170,000 instances;
- SCDOER alpha factor: 0.04;
- Each case was repeated for 20 trials. Evaluation of means and standard deviations:
  - Prediction Error: Root Mean Square Error (RMSE);
  - Execution time: Total Real time (TRT) of execution for all algorithms, in seconds.

Experimental Results

TABLE II

RMSEs FOR THE SINGLE MODEL OS-ELMSW.

SW	HYP dataset	PM_{10} dataset
1	0.14545 ± 0.00001	0.10799 ± 0.00004
50	0.14400 ± 0.00002	0.10830 ± 0.00005
100	0.14402 ± 0.00002	0.10836 ± 0.00006

TABLE III

TRT OF EXECUTION (IN SECONDS) FOR THE SINGLE MODEL OS-ELMSW.

Dataset	SW	# of threads			
		1	*2*	*4*	*8*
Hyperplane	50	142.69 ± 0.76	160.97 ± 1.72	133.40 ± 1.04	442.63 ± 19.97
PM_{10}	1	4.89 ± 0.06	7.98 ± 0.14	9.59 ± 0.28	9.67 ± 0.20

Experimental Results

TABLE IV
RMSEs FOR THE ENSEMBLES.

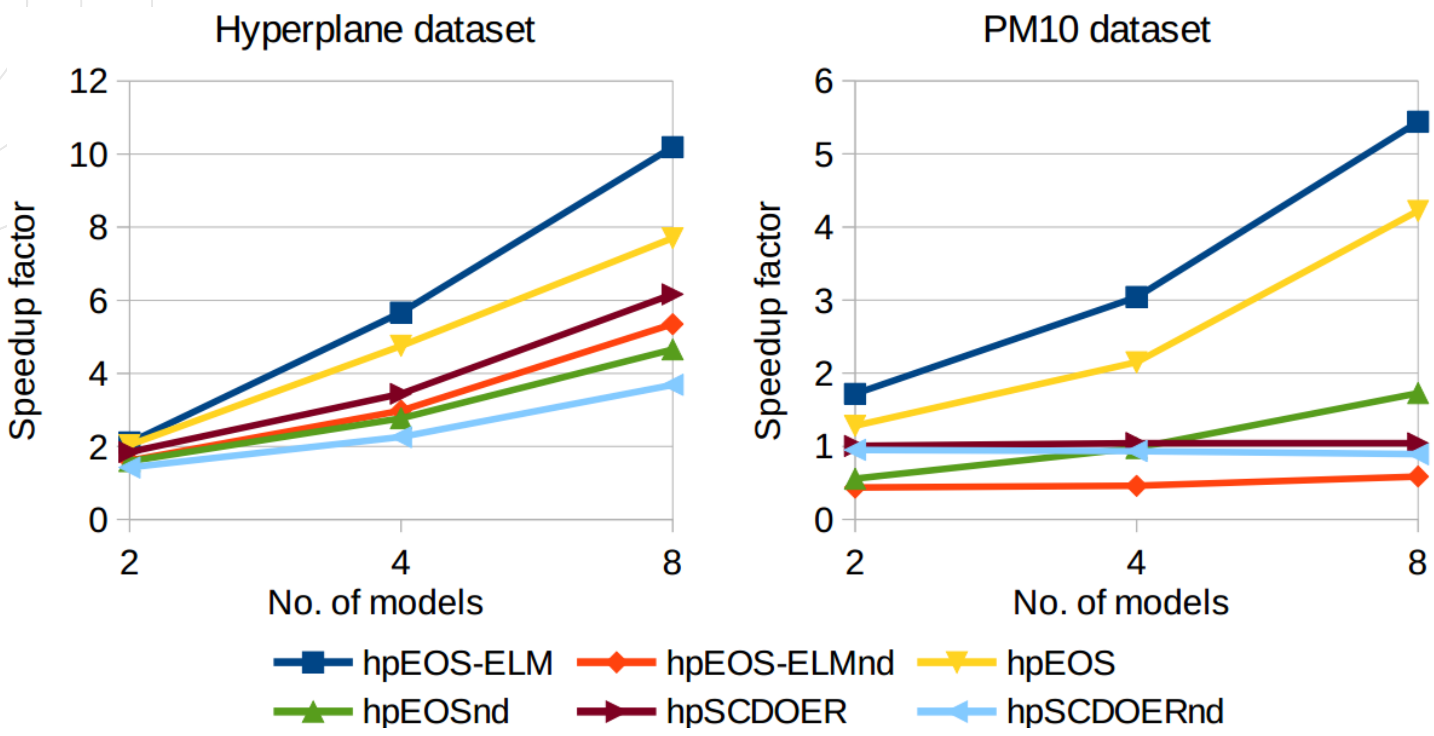
Dataset	Ensemble	# of models		
		2	*4*	*8*
HYP	EOS-ELM	0.14396 ± 0.00001	0.14395 ± 0.00000	0.14394 ± 0.00000
	EOS	0.07227 ± 0.00009	0.08266 ± 0.00001	0.09276 ± 0.00001
	SCDOER	0.06377 ± 0.00051	0.06674 ± 0.00054	0.07065 ± 0.00035
PM_{10}	EOS-ELM	0.10792 ± 0.00003	0.10789 ± 0.00001	0.10787 ± 0.00002
	EOS	0.10816 ± 0.00003	0.10788 ± 0.00002	0.10776 ± 0.00002
	SCDOER	0.10931 ± 0.00004	0.10919 ± 0.00003	0.10910 ± 0.00002

Experimental Results

TABLE V
TRT OF EXECUTION (IN SECONDS) FOR THE ENSEMBLES.

Dataset	Ensemble	# of models		
		2	4	8
Hyperplane	EOS-ELM	266.70 ± 2.92	535.73 ± 4.42	1069.36 ± 7.49
	hpEOS-ELM	126.49 ± 3.32	94.56 ± 4.80	104.94 ± 8.16
	hpEOS-ELMnd	165.73 ± 5.17	179.74 ± 5.07	200.00 ± 5.77
	EOS	291.25 ± 1.96	569.58 ± 2.71	1091.72 ± 5.83
	hpEOS	142.80 ± 5.29	119.93 ± 5.75	141.73 ± 3.76
	hpEOSnd	182.07 ± 4.66	205.70 ± 5.76	234.76 ± 3.95
	SCDOER	328.12 ± 2.14	639.30 ± 4.80	1254.89 ± 6.75
	hpSCDOER	177.66 ± 4.57	185.95 ± 4.46	203.58 ± 5.94
	hpSCDOERnd	230.66 ± 2.69	283.21 ± 5.73	340.72 ± 7.58
PM_{10}	EOS-ELM	10.44 ± 0.06	20.03 ± 0.21	38.44 ± 0.36
	hpEOS-ELM	6.07 ± 0.09	6.58 ± 0.07	7.07 ± 0.10
	hpEOS-ELMnd	23.92 ± 1.94	43.52 ± 5.10	65.44 ± 3.56
	EOS	23.79 ± 0.15	65.44 ± 0.71	166.30 ± 1.35
	hpEOS	18.56 ± 0.36	30.44 ± 0.15	39.41 ± 0.15
	hpEOSnd	42.67 ± 3.42	66.92 ± 2.31	96.25 ± 2.65
	SCDOER	848.90 ± 2.47	875.35 ± 4.77	883.20 ± 3.02
	hpSCDOER	844.10 ± 2.28	837.65 ± 1.18	847.81 ± 1.14
	hpSCDOERnd	893.33 ± 3.52	935.56 ± 7.07	989.76 ± 7.16

Experimental Results



Conclusions

- The high-performance ensembles perform better when compared with their corresponding serial version in most cases.
- The hpEOS-ELM presented the highest accelerations because of its simplicity;
- Multi-node approaches: necessary to be tested in environments with high-performance networks, e. g., RDMA/InfiniBand.
- The high-performance techniques presented in this work may also be suitable for other types of ensembles and online learning models.
- An interesting challenge would be the application in ensembles composed of models with adaptive sliding window.

Main References

- BUENO, A.; COELHO, G. P.; BERTINI, J. R. Online Sequential Learning based on Extreme Learning Machines for Particulate Matter Forecasting. In: Brazilian Conference on Intelligent Systems (BRACIS). Uberlandia: IEEE, 2017. p. 169–174. ISBN 9781538624074.
- HUANG, G.-B. et al. Extreme learning machine: Theory and applications. Neurocomputing, v. 70, n. 1-3, p. 489–501, 2006. ISSN 09252312.
- KRAWCZYK, B. GPU-accelerated extreme learning machines for imbalanced data streams with Concept Drift. Procedia Computer Science, Elsevier Masson SAS, v. 80, p. 1692–1701, 2016. ISSN 18770509.
- LAN, Y.; SOH, Y. C.; HUANG, G. B. Ensemble of online sequential extreme learning machine. Neurocomputing, Elsevier, v. 72, n. 13-15, p. 3391–3395, 2009. ISSN 09252312.
- LIANG, N.-Y. et al. A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks. IEEE Transactions on Neural Networks, v. 17, n. 6, p. 1411–1423,2006. ISSN 1045-9227.
- SOARES, S. G.; ARAÚJO, R. A dynamic and on-line ensemble regression for changing environments. Expert Systems with Applications, Elsevier Ltd, v. 42, n. 6, p. 2935–2948, 2015. ISSN 09574174.
- VAN HEESWIJK, M. et al. GPU-accelerated and parallelized ELM ensembles for large-scale regression. Neurocomputing, Elsevier, v. 74, n. 16, p. 2430–2437, 2011. ISSN 09252312.

Questions?

Info about the authors:

- Luís F. L. Grim, ORCID 0000-0002-1221-4095
- **André L. S. Gradvohl**, ORCID 0000-0002-6520-9740

We would like to acknowledge the following supporters:

- *Hasso Plattner Institute – Future SOC Lab*
- *School of Technology at University of Campinas*

Special Thanks to:

Inria inventors for the digital world & Overleaf for the generous financial support.

This presentation is available at:
https://doi.org/10.5281/zenodo.1406087