Online Hyperparameter Optimization for Streaming Neural Networks

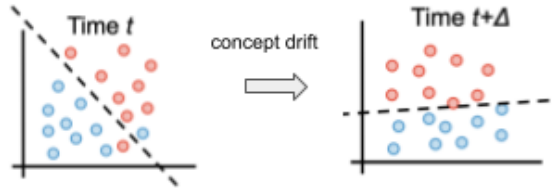
Nuwan Gunasekara, Heitor Gomes, Bernhard Pfahringer, and Albert Bifet

IJCNN 2022

Neural networks for evolving data streams

- Neural networks in batch learning
  - Good performance in NLP and image classification
  - Sensitive to hyper parameters
  - Requires mini-batch training
  - Demands high computing resources (memory and processing)

- Evolving data stream learning
  - Data is **not IID** (Independent and Identically Distributed)
  - Incremental testing and training
  - Data susceptible to **concept drifts**
  - Model must be able to predict at any moment
  - Model must use limited computing resources



Related work

- Autonomous Deep Learning (ADL): Continual learning approach for dynamic environments. (Ashfahani and Pratama 2019)
  - the network's depth and width are dynamically managed according to
    - drift detection (**adds a new hidden layer**)
    - level of mutual information between hidden layers (**merge hidden layers**).
    - network's generalization power which is estimated using **bias** and **variance** (**grow or prune hidden nodes**)
  - The network is structured differently than a normal MLP,
    - where each layer had a softmax layer, and the final output is obtained by **weighted voting**.
  - Extended to
    - MLPs (Pratama et al. 2019)
    - autoencoders (Ashfahani et al. 2020)
    - RNNs (Das et al. 2019)
  - Zhao et al. (2019) compared their SVM distribution-free one-pass learning method against Ashfahani and Pratama (2019)
    - data were **fed in batches**, and the models were trained for **multiple iterations** using a **given batch**.
    - Defining **robustness** as a relationship **between one algorithm's accuracy and the smallest accuracy among all algorithms**, the authors found the Ashfahani and Pratama (2019) method to be one of **the least robust** ones.

Related work

- Self Hyper-parameter Tuning for Stream Classification Algorithms (Veloso and Gama 2020)
  - requires a **double pass** over the data during the exploration phase.
  - It was further **improved** to use a **single pass** (Veloso et al. 2021)
    - During **exploration** phase which is **triggered by a concept drift**,
      - for ***n*** hyperparameters, the method creates ***n + 1*** **models + 7 experimental** models proposed by the Nelder–Mead algorithm, using shallow copies of the best model.
    - exploration starts by randomly selecting the *n + 1* models
    - it **stops** when the **best, good, and worst models converge**.
    - Compared against
      - a model with default parameters,
      - offline grid search,
      - offline random search
      - for recommendation, regression, and classification tasks.
    - According to the authors, results for **classification** and regression were **fairly similar** to the **model with the default hyperparameters**.

Continuously **A**daptive **N**eural Networks for **D**ata Streams (CAND)

- **Train a pool (*P*)**of small MLPs with different configurations:
  - optimizer, learning rate and network width
- For **prediction**, **choose the best** network using **estimated** loss
  - loss estimator: **ADWIN** (Bifet and Gavalda 2007)
    - Discards older distribution after a concept drift
- Two orthogonal methods to **accelerate training**:
  - **CAND$_{sub}$: Train a subset (*M*)** of the pool at a given moment.
    - half of the subset based on **estimated** loss, other half random.
  - **CAND^{SB} : Skip Backpropagation** when the loss is below a certain threshold *b*.
- **Online standardization**(Ikonomovska, Gama, and Dz˘eroski 2011) and **one-hot encoding** of input data.
- **Warm-up period (train *P*)**: either 1% of the dataset or 1000 instances, whichever is smaller.

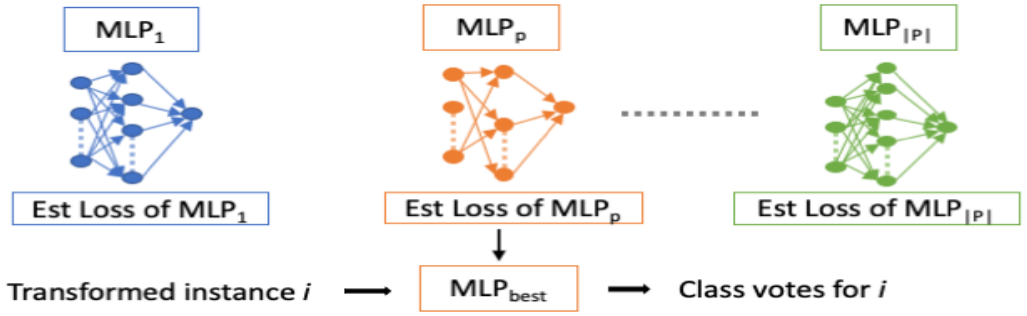


Figure 1: CAND prediction Algorithm 1

Algorithm 2 CAND$^{SB}_{sub}$ TRAIN

Input: instance i, pool P of MLPs, number of instances for warm-up W, $|M|$ pool size, skip backpropagation threshold b.

1: **if** the current instance count $< W$ **then**
2: M=P
3: **else**
4: M={Half of M is selected from P by lowest **estimated** loss, other half random}
5: **end if**
6: **for** all $m \in M$ **do**
7: TRAIN(i,m) ▷ invoked in $|M|$ separate threads
8: ▷ will skip backpropagation, if the loss $< b$
9: ▷ will update loss estimate for m
10: **end for**
11: Wait for all $|M|$ training threads to finish.

Experiment setup

- Datasets (17)
  - low-dimensional(< 2000 features)
    - Includes data sets with different concept drifts
      - **abrupt** (AGR_a, LED_a)
      - **gradual** (AGR_g, LED_g)
      - **fast incremental changes** (RBF_f)
      - **moderate incremental changes** (RBF m)
      - AGR_a, AGR_g, LED_a and LED_g
        - New Concept after every 250000
    - From Gomes, Read, and Bifet (2019)
  - high-dimensional(≥ 2000 features)

name	type	instances	features before one-hot	features after one-hot	# classes	class distribution max(%)	class distribution min(%)
airlines	LR^d	539382	7	614	2	55.46	44.54
electricity	LR^d	45310	8	14	2	57.55	42.45
kdd99	LR^d	4898430	41	122	23	56.24	0.00*
WISDM	LR^d	5417	45	80	6	38.43	4.53
covtype	LR^d	581010	54	54	7	48.76	0.47
nomao	LR^d	34464	118	172	2	71.44	28.56
AGR_a	LDS^d	1000000	9	40	2	52.83	47.17
AGR_g	LDS^d	1000000	9	40	2	52.83	47.17
RBF_f	LDS^d	1000000	10	10	5	30.01	9.27
RBF_m	LDS^d	1000000	10	10	5	30.01	9.27
LED_a	LDS^s	1000000	24	24	10	10.08	9.94
LED_g	LDS^s	1000000	24	24	10	10.08	9.94
epsilon	HR^d	100000	2000	2000	2	50.05	49.95
SVHN	HR^d	26032	3072	3072	10	19.59	6.13
gisette	HR^d	6000	5000	5000	2	50.00	50.00
spam	HR^s	9323	39916	39916	2	74.40	25.60
sector	HR^s	6412	55197	55197	105	1.25	0.14

Table 1: Dataset properties and data type: (**L**)ow dimensional, has (**D**)rifts, (**H**)igh dimensional, (**R**)eal world, (**S**)ynthetic, dense(^{d}), sparse(^{s}). * 1.00E-04.

Experiment setup

- Comparison methods
  - Current state-of-the-art streaming methods
    - Streaming Random Patches (**SRP**) (Gomes, Read, and Bifet 2019)
    - Adaptive Random Forest (**ARF**) (Gomes et al. 2017b)
    - Are ensemble methods, which use efficient base learners and drift detectors
    - Ensemble sizes: **10** and **30**
    - Same hyper parameters as in Gomes, Read, and Bifet (2019)
  - Autonomous Deep Learning (**ADL**) (Ashfahani and Pratama 2019)
- CAND configurations
  - Larger ***P*** pool configurations (|P|= 30)
    - optimizers: Adam and SGD
    - 5 learning rates: 5e-1, 5e-2, 5e-3, 5e-4,and 5e-5.
    - All the MLPs were **single-layer** ones with either 2^8 , 2^9 , or 2^{10} neurons in the **hidden layer**
  - Smaller ***M*** pool size 10 (to match ensemble size 10)
  - Cross entropy loss for loss calculation
- All experiments were run on MOA

Effect of $|P|$ and CAND prediction method

dataset	ADL	Ensemble learners				CAND $\|P\|$=10			
		ARF 10, 60%	SRP 10, 60%	ARF 10, 10%	SRP 10, 10%	Min Estd Loss	Majority Vote	Best MLP*!	At Least One*#
Low dimensional									
Avg Acc	79.51	88.76	**88.95**	79.87	86.40	84.16	83.05	84.64	94.32
Avg Rank	4.92	2.25	**1.42**	5.67	3.08	4.83	5.83		
Data with drifts									
Avg Acc	58.29	78.13	**80.04**	50.25	56.28	79.51	78.20	79.47	91.12
Avg Rank	5.67	2.83	3.00	6.33	5.67	**1.67**	2.83		
High dimensional									
Avg Acc	55.00	49.80	49.95	53.59	55.46	**80.58**	71.28	80.45	89.22
Avg Rank	4.20	6.20	6.00	4.20	4.00	**1.20**	2.20		
Overall Acc	64.81	73.55	74.34	61.69	66.67	**81.46**	77.88	81.58	91.69
Overall Rank	4.97	3.62	3.32	5.47	4.26	**2.65**	3.71		

Table 2: CAND($|P|$=10) against ADL and ensemble learners with 10 base learners. * hypothetical MLP selection criteria. ! single best MLP for the given dataset. # at least one MLP predicted the correct label.

dataset	ARF 30, 60%	SRP 30, 60%	ARF 30, 10%	SRP 30, 10%	CAND $\|P\|$=30			
					Min Estd Loss	Majority Vote	Best MLP*!	At Least One*#
Low dimensional								
Avg Acc	89.10	**89.47**	80.23	87.64	88.86	86.08	89.47	96.98
Avg Rank	3.25	**1.83**	5.92	3.00	2.42	4.58		
Data with drifts								
Avg Acc	79.69	**81.72**	52.13	63.50	79.50	77.67	79.49	93.83
Avg Rank	2.83	**1.83**	5.67	5.33	**1.83**	3.50		
High dimensional								
Avg Acc	38.00	19.30	54.82	57.79	**82.21**	43.80	82.25	93.87
Avg Rank	5.20	5.00	3.00	2.40	**1.00**	4.40		
Avg Acc	70.75	66.10	62.84	70.34	**83.60**	70.68	83.82	94.95
Avg Rank	3.68	2.76	4.97	3.65	**1.79**	4.15		

Table 3: CAND($|P|$=30) against ensemble learners with 30 base learners. * hypothetical MLP selection criteria. ! single best MLP for the given dataset. # at least one MLP predicted the correct label. *underline* value is worse (smaller accuracy) than the 10 learner counterpart in table 2 (ranks are not considered in this comparison).

Effect of $|P|$ and CAND prediction method

TABLE II

CAND($|P|$=10) AGAINST ADL AND ENSEMBLE LEARNERS WITH 10 BASE LEARNERS. * HYPOTHETICAL MLP SELECTION CRITERIA. ! SINGLE BEST MLP FOR THE GIVEN DATASET. # AT LEAST ONE MLP PREDICTED THE CORRECT LABEL.

dataset	ADL	Ensemble learners				CAND $\|P\|$=10			
dataset		ARF 10, 60%	SRP 10, 60%	ARF 10, 10%	SRP 10, 10%	Min Estd Loss	Majo-rity Vote	Best MLP*!	At Least One*#
airlines	61.06	65.86	**66.74**	61.18	64.90	61.14	61.24	61.14	83.27
electricity	74.20	**89.87**	89.14	58.00	83.47	84.98	82.48	85.24	95.59
kdd99	99.96	99.96	**99.97**	99.94	**99.97**	99.92	99.91	99.91	99.97
WISDM	56.37	85.28	**85.36**	77.77	83.54	72.96	71.71	74.35	90.70
covtype	87.91	94.49	**95.28**	85.34	89.46	88.91	86.01	89.90	97.41
nomao	**97.58**	97.08	97.23	97.00	97.06	97.02	96.96	97.27	98.99
Avg Acc	79.51	88.76	**88.95**	79.87	86.40	84.16	83.05	84.64	94.32
Avg Rank	4.92	2.25	**1.42**	5.67	3.08	4.83	5.83		
AGR_a	63.56	85.91	**92.44**	64.45	77.21	89.59	87.34	89.40	97.66
AGR_g	61.30	79.99	**87.55**	62.98	76.19	87.14	85.20	86.98	96.60
RBF_f	31.93	**71.20**	70.84	30.01	46.88	67.26	64.76	67.67	88.43
RBF_m	46.26	84.70	83.19	30.01	62.07	**85.75**	84.75	85.62	95.91
LED_a	73.66	73.92	73.49	56.24	37.24	**74.02**	73.91	73.89	84.38
LED_g	73.02	73.06	72.73	57.80	38.09	**73.28**	73.22	73.23	83.76
Avg Acc	58.29	78.13	**80.04**	50.25	56.28	79.51	78.20	79.47	91.12
Avg Rank	5.67	2.83	3.00	6.33	5.67	**1.67**	2.83		
epsilon	77.54	50.26	50.21	58.06	55.87	**85.89**	79.69	85.73	96.41
SVHN	22.59	19.52	20.35	20.57	22.25	**55.68**	38.58	55.94	76.02
gisette	76.27	82.36	82.17	88.04	89.64	**96.23**	93.74	96.25	99.48
spam	**98.34**	96.12	96.22	96.87	96.80	97.99	97.56	98.13	99.74
sector	0.25	0.74	0.80	4.39	12.73	**67.13**	46.84	66.21	74.47
Avg Acc	55.00	49.80	49.95	53.59	55.46	**80.58**	71.28	80.45	89.22
Avg Rank	4.20	6.20	6.00	4.20	4.00	**1.20**	2.20		
Overall Acc	64.81	73.55	74.34	61.69	66.67	**81.46**	77.88	81.58	91.69
Overall Rank	4.97	3.62	3.32	5.47	4.26	**2.65**	3.71		

TABLE III

CAND($|P|$=30) AGAINST ENSEMBLE LEARNERS WITH 30 BASE LEARNERS. * HYPOTHETICAL MLP SELECTION CRITERIA. ! SINGLE BEST MLP FOR THE GIVEN DATASET. # AT LEAST ONE MLP PREDICTED THE CORRECT LABEL. $\underline{underline}$ VALUE IS WORSE (SMALLER ACCURACY) THAN THE 10 LEARNER COUNTERPART IN TABLE II (RANKS ARE NOT CONSIDERED IN THIS COMPARISON).

dataset	ARF 30, 60%	SRP 30, 60%	ARF 30, 10%	SRP 30, 10%	CAND $\|P\|$=30 Min Estd Loss	Majo-rity Vote	Best MLP*!	At Least One*#
airlines	66.46	**67.97**	61.18	66.77	61.44	61.87	61.63	86.77
electricity	90.57	89.48	58.09	84.57	**91.55**	88.15	92.16	98.64
kdd99	99.96	99.97	99.94	**99.98**	99.96	99.94	99.97	99.99
WISDM	85.59	86.43	78.90	85.86	**88.85**	80.26	90.81	97.96
covtype	94.79	**95.60**	86.09	91.35	93.77	89.00	94.54	99.14
nomao	97.20	97.37	97.16	97.33	**97.56**	97.27	97.72	99.37
Avg Acc	89.10	**89.47**	80.23	87.64	88.86	86.08	**89.47**	96.98
Avg Rank	3.25	**1.83**	5.92	3.00	2.42	4.58		
AGR_a	87.48	**92.96**	64.45	79.71	89.70	87.37	89.38	98.77
AGR_g	81.96	**89.14**	62.98	75.99	87.23	84.84	87.03	98.31
RBF_f	75.02	**75.43**	30.01	52.17	66.98	62.34	67.67	93.02
RBF_m	**86.63**	85.68	30.01	68.07	85.78	84.31	85.63	97.80
LED_a	73.96	73.97	62.55	53.26	**74.02**	73.94	73.99	87.71
LED_g	73.10	73.14	62.79	51.81	**73.30**	73.22	73.26	87.35
Avg Acc	79.69	**81.72**	52.13	63.50	79.50	77.67	79.49	93.83
Avg Rank	2.83	**1.83**	5.67	5.33	**1.83**	3.50		
epsilon	NA	NA	61.21	60.20	**85.89**	50.04	85.76	99.76
SVHN	19.89	20.72	21.76	23.59	**57.02**	18.79	57.38	87.12
gisette	74.72	75.80	89.25	90.93	**96.26**	50.84	96.36	99.93
spam	95.40	NA	97.15	97.29	**98.30**	97.60	98.54	99.90
sector	NA	NA	4.72	16.95	**73.56**	1.71	73.19	82.62
Avg Acc	38.00	19.30	54.82	57.79	**82.21**	43.80	82.25	93.87
Avg Rank	5.20	5.00	3.00	2.40	**1.00**	4.40		
Overall Acc	70.75	66.10	62.84	70.34	**83.60**	70.68	83.82	94.95
Overall Rank	3.68	2.76	4.97	3.65	**1.79**	4.15		

Different variants of CAND against best ensembles

Accuracy

dataset	ARF 10, 60%	SRP 10, 60%	ARF 30, 10%	SRP 30, 10%	C $\|P\|$=10	C_{sub} $\|P\|$=30 $\|M\|$=10	C^{SB}_{sub} $\|P\|$=30 $\|M\|$=10	C $\|P\|$=30
Low dimensional								
Avg Acc	88.76	**88.95**	80.23	87.64	84.13	88.87	88.85	88.86
Avg Rank	4.58	3.17	6.83	3.67	7.00	**3.08**	4.42	3.25
Data with drifts								
Avg Acc	78.13	**80.04**	52.13	63.50	79.52	79.34	78.92	79.50
Avg Rank	4.33	3.67	7.67	7.33	2.58	3.08	5.00	**2.33**
High dimensional								
Avg Acc	49.80	49.95	54.82	57.79	80.68	81.46	80.29	**82.21**
Avg Rank	7.60	7.40	5.80	5.20	2.60	2.10	3.60	**1.70**
Avg Acc	73.55	74.34	62.84	70.34	81.49	83.33	82.83	**83.60**
Avg Rank	5.38	4.59	6.82	5.41	4.15	2.79	4.38	**2.47**

Wall time

dataset	ARF 10, 60%	SRP 10, 60%	ARF 30, 10%	SRP 30, 10%	C $\|P\|$=10	C_{sub} $\|P\|$=30 $\|M\|$=10	C^{SB}_{sub} $\|P\|$=30 $\|M\|$=10	C $\|P\|$=30
Low dimensional								
Avg	56.69	102.05	**43.70**	153.19	3472.79	2317.48	1590.30	6652.62
Avg Rank	1.83	3.33	**1.17**	3.67	6.83	6.17	5.00	8.00
Data with drifts								
Avg	99.18	226.30	**78.72**	1308.47	3813.80	2901.09	2070.31	6377.17
Avg Rank	1.67	3.00	**1.33**	4.33	7.00	6.00	4.67	8.00
High dimensional								
Avg	23735.15	19390.41	9110.72	4940.42	6749.59	5353.29	**3122.52**	14980.53
Avg Rank	4.00	5.20	3.00	**2.20**	5.80	5.40	3.00	7.40
Avg	7035.94	5818.95	2722.83	**1968.94**	4556.91	3416.34	2210.37	9004.78
Avg Rank	2.41	3.76	**1.76**	3.47	6.59	5.88	4.29	7.82

$CAND_{sub}(|M|=10, |P|=30)$ on LED_a

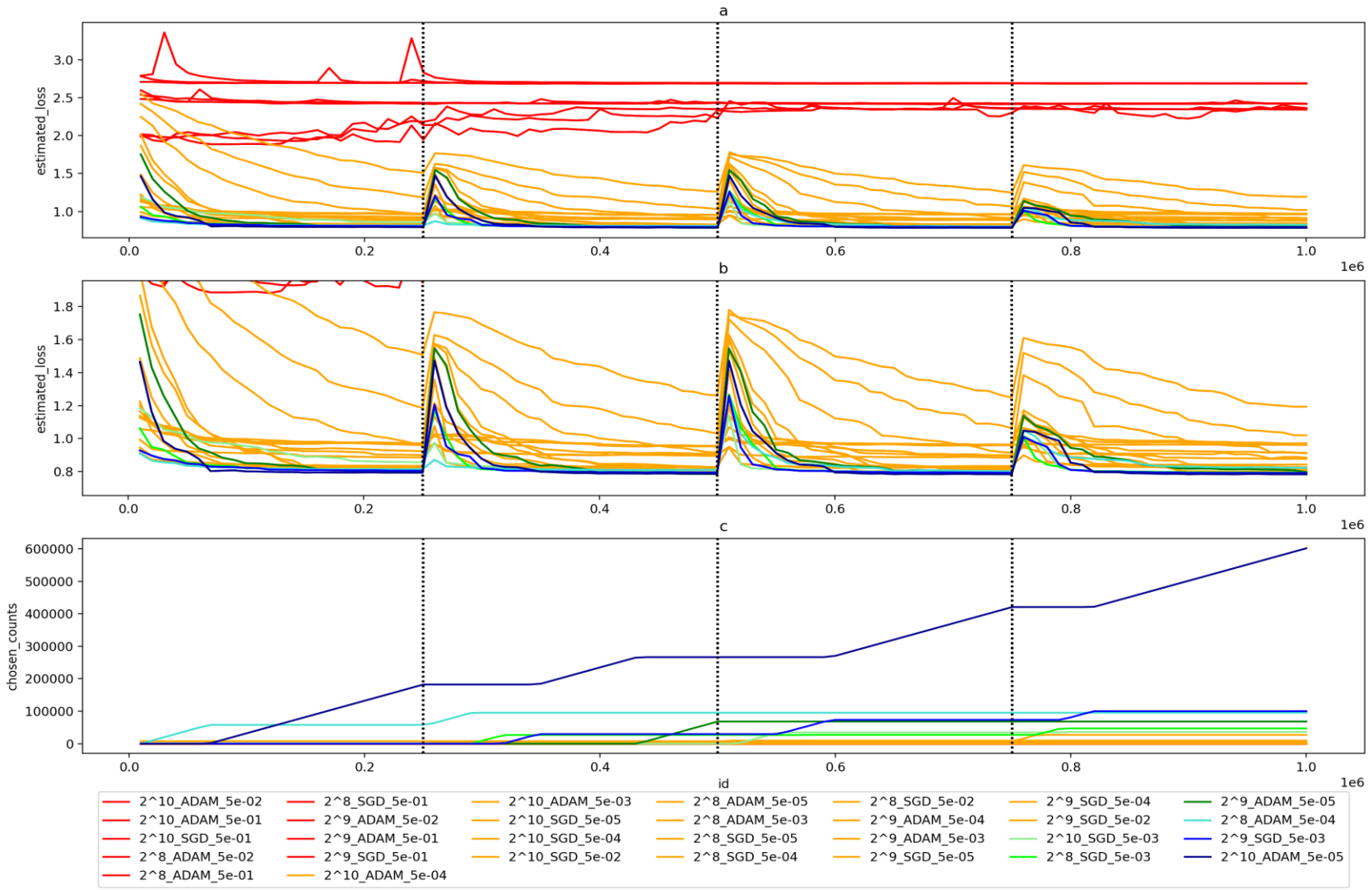


Figure 2: a): MLPs estimated losses, b): MLPs estimated losses (zoomed-in), c): chosen counts of MLPs. MLP colour scheme: cooler colours for the most chosen, and warmer colours for the least chosen MLPs. Vertical dotted lines: start of concept drift.

Effect of mini-batch size and GPU training

	Acc					Wall Time				
device	CPU	GPU				CPU	GPU			
batch size	1	1	4	16	32	1	1	4	16	32
Low dimensional										
Avg	88.87	88.97	85.62	81.53	80.23	2317.48	2930.49	748.65	325.35	254.94
Avg rank	1.67	1.33	3	4.17	4.83	4	4.83	3.17	2	1
Data with drifts										
Avg	79.34	79.35	79.41	78.42	78.2	2901.09	9544.88	2197.75	812.16	592.73
Avg rank	3.5	2.33	3	2.5	3.67	3.83	5	3.17	2	1
High dimensional										
Avg	81.46	80.85	81.48	80.17	80.52	5353.29	388.25	194.9	141.02	130.66
Avg rank	2	2.8	2.6	3.8	3.8	5	4	3	2	1
Overall	83.33	83.19	82.21	80.03	79.6	3416.34	4517.26	1097.23	442.95	337.61
Overall rank	2.41	2.12	2.88	3.47	4.12	4.24	4.65	3.12	2	1

Table 8: Effect of mini-batch size on accuracy and wall time (s) for CAND$_{sub}$($|M|$=10, $|P|$=30).

Effect of smaller pool size $|M|$

	Acc					Wall Time				
$\|M\|$	2	4	6	8	10	2	4	6	8	10
Low dimensional										
Avg	88.33	88.91	88.94	**88.98**	88.97	**2159.31**	2384.58	2631.2	2825.19	2930.49
Avg rank	5	3.5	2.83	**1.83**	**1.83**	**1.33**	2	3.5	3.83	4.33
Data with drifts										
Avg	78.68	78.95	79.12	79.26	**79.35**	**5649.33**	6834.95	7939.9	9018.68	9544.88
Avg rank	4.83	4.17	2.5	2	**1.5**	**1**	2	3.17	4.17	4.67
High dimensional										
Avg	76.38	77.55	79.38	80.77	**80.85**	**264.77**	282.72	313.27	339.26	388.25
Avg rank	4.6	4.2	2.8	1.8	**1.6**	**1**	2	3	4	5
Overall	81.41	82.06	82.67	83.13	**83.19**	**2833.87**	3337.1	3823.12	4279.97	4517.26
Overall rank	4.82	3.94	2.71	1.88	**1.65**	**1.12**	2	3.24	4	4.65

Table 9: Effect of $|M|$ on accuracy and wall time (s) for CAND$_{sub}$(M, $|P|$=30, mini-batch size=1, GPU trained)

Effect of skip backpropagation threshold

	Acc				Wall Time			
SB	0	0.3	0.6	0.9	0	0.3	0.6	0.9
Low dimensional								
Avg	**88.97**	88.97	88.76	86.15	2930.49	2267.76	2070.6	**1612.83**
Avg rank	1.83	**1.67**	2.67	3.83	4	3	1.83	**1.17**
Data with drifts								
Avg	79.35	**79.45**	78.85	69.93	9544.88	8373.63	7425.06	**6774.56**
Avg rank	**1.5**	**1.5**	3	4	4	3	**1.5**	**1.5**
High dimensional								
Avg	80.85	**81.23**	79.66	76.08	388.25	327.2	323.47	**294.98**
Avg rank	2	**1.4**	3	3.6	4	2.8	2.2	**1**
Overall	83.19	**83.33**	82.59	77.46	4517.26	3852.02	3446.55	**3047.01**
Overall rank	1.76	**1.53**	2.88	3.82	4	2.94	1.82	**1.24**

Table 10: Effect of skip backpropagation threshold on accuracy and wall time (s) for $\text{CAND}^{SB}_{sub}(|M|=10, |P|=30$, mini-batch size=1, GPU trained)

Efficient CAND variants

	Acc				Wall Time			
$\|M\|$	10	10	8	6	10	10	8	6
batch size	1	4	4	4	1	4	4	4
SB	0	0.3	0.3	0.3	0	0.3	0.3	0.3
Low dimensional								
Avg	**88.97**	85.62	81.53	80.23	2930.49	748.65	325.35	**254.94**
Avg rank	**1**	2	3.17	3.83	4	3	2	**1**
Data with drifts								
Avg	79.35	**79.41**	78.42	78.2	9544.88	2197.75	812.16	**592.73**
Avg rank	**2**	2.67	2.17	3.17	4	3	2	**1**
High dimensional								
Avg	80.85	**81.48**	80.17	80.52	388.25	194.9	141.02	**130.66**
Avg rank	**2**	**2**	3	3	4	3	2	**1**
Overall	**83.19**	82.21	80.03	79.6	4517.26	1097.23	442.95	**337.61**
Overall rank	**1.65**	2.24	2.76	3.35	4	3	2	**1**

Table 11: Accuracy and wall time (s) for efficient $\text{CAND}^{SB}_{sub}(M, |P|\text{=30, GPU trained})$ variants.

Conclusion and Future directions

- Conclusions
  - CAND yields **good performance** compared to current state-of-the-art stream learning methods.
    - Performs well on **high-dimensional** data
  - **small mini-batches** yield similar accuracy to single-instance fully incremental training
- CAND with GPU support is available on latest MOA (Massive Online Analysis) https://github.com/Waikato/moa
- Future directions
  - During GPU training, load only the smaller *M* pool to the GPU memory.
  - Experiment with different NN architectures

Q&A

Effect of $|P|$ and CAND prediction method

TABLE II

CAND($|P|$=10) AGAINST ADL AND ENSEMBLE LEARNERS WITH 10 BASE LEARNERS. * HYPOTHETICAL MLP SELECTION CRITERIA. ! SINGLE BEST MLP FOR THE GIVEN DATASET. # AT LEAST ONE MLP PREDICTED THE CORRECT LABEL.

dataset	ADL	Ensemble learners				CAND $\|P\|$=10			
dataset		ARF 10, 60%	SRP 10, 60%	ARF 10, 10%	SRP 10, 10%	Min Estd Loss	Majo-rity Vote	Best MLP*!	At Least One*#
airlines	61.06	65.86	**66.74**	61.18	64.90	61.14	61.24	61.14	83.27
electricity	74.20	**89.87**	89.14	58.00	83.47	84.98	82.48	85.24	95.59
kdd99	99.96	99.96	**99.97**	99.94	**99.97**	99.92	99.91	99.91	99.97
WISDM	56.37	85.28	**85.36**	77.77	83.54	72.96	71.71	74.35	90.70
covtype	87.91	94.49	**95.28**	85.34	89.46	88.91	86.01	89.90	97.41
nomao	**97.58**	97.08	97.23	97.00	97.06	97.02	96.96	97.27	98.99
Avg Acc	79.51	88.76	**88.95**	79.87	86.40	84.16	83.05	84.64	94.32
Avg Rank	4.92	2.25	**1.42**	5.67	3.08	4.83	5.83		
AGR_a	63.56	85.91	**92.44**	64.45	77.21	89.59	87.34	89.40	97.66
AGR_g	61.30	79.99	**87.55**	62.98	76.19	87.14	85.20	86.98	96.60
RBF_f	31.93	**71.20**	70.84	30.01	46.88	67.26	64.76	67.67	88.43
RBF_m	46.26	84.70	83.19	30.01	62.07	**85.75**	84.75	85.62	95.91
LED_a	73.66	73.92	73.49	56.24	37.24	**74.02**	73.91	73.89	84.38
LED_g	73.02	73.06	72.73	57.80	38.09	**73.28**	73.22	73.23	83.76
Avg Acc	58.29	78.13	**80.04**	50.25	56.28	79.51	78.20	79.47	91.12
Avg Rank	5.67	2.83	3.00	6.33	5.67	**1.67**	2.83		
epsilon	77.54	50.26	50.21	58.06	55.87	**85.89**	79.69	85.73	96.41
SVHN	22.59	19.52	20.35	20.57	22.25	**55.68**	38.58	55.94	76.02
gisette	76.27	82.36	82.17	88.04	89.64	**96.23**	93.74	96.25	99.48
spam	**98.34**	96.12	96.22	96.87	96.80	97.99	97.56	98.13	99.74
sector	0.25	0.74	0.80	4.39	12.73	**67.13**	46.84	66.21	74.47
Avg Acc	55.00	49.80	49.95	53.59	55.46	**80.58**	71.28	80.45	89.22
Avg Rank	4.20	6.20	6.00	4.20	4.00	**1.20**	2.20		
Overall Acc	64.81	73.55	74.34	61.69	66.67	**81.46**	77.88	81.58	91.69
Overall Rank	4.97	3.62	3.32	5.47	4.26	**2.65**	3.71		

TABLE III

CAND($|P|$=30) AGAINST ENSEMBLE LEARNERS WITH 30 BASE LEARNERS. * HYPOTHETICAL MLP SELECTION CRITERIA. ! SINGLE BEST MLP FOR THE GIVEN DATASET. # AT LEAST ONE MLP PREDICTED THE CORRECT LABEL. $\underline{underline}$ VALUE IS WORSE (SMALLER ACCURACY) THAN THE 10 LEARNER COUNTERPART IN TABLE II (RANKS ARE NOT CONSIDERED IN THIS COMPARISON).

dataset	ARF 30, 60%	SRP 30, 60%	ARF 30, 10%	SRP 30, 10%	CAND $\|P\|$=30 Min Estd Loss	Majo-rity Vote	Best MLP*!	At Least One*#
airlines	66.46	**67.97**	61.18	66.77	61.44	61.87	61.63	86.77
electricity	90.57	89.48	58.09	84.57	**91.55**	88.15	92.16	98.64
kdd99	99.96	99.97	99.94	**99.98**	99.96	99.94	99.97	99.99
WISDM	85.59	86.43	78.90	85.86	**88.85**	80.26	90.81	97.96
covtype	94.79	**95.60**	86.09	91.35	93.77	89.00	94.54	99.14
nomao	97.20	97.37	97.16	97.33	**97.56**	97.27	97.72	99.37
Avg Acc	89.10	**89.47**	80.23	87.64	88.86	86.08	**89.47**	96.98
Avg Rank	3.25	**1.83**	5.92	3.00	2.42	4.58		
AGR_a	87.48	**92.96**	64.45	79.71	89.70	87.37	89.38	98.77
AGR_g	81.96	**89.14**	62.98	75.99	87.23	84.84	87.03	98.31
RBF_f	75.02	**75.43**	30.01	52.17	66.98	62.34	67.67	93.02
RBF_m	**86.63**	85.68	30.01	68.07	85.78	84.31	85.63	97.80
LED_a	73.96	73.97	62.55	53.26	**74.02**	73.94	73.99	87.71
LED_g	73.10	73.14	62.79	51.81	**73.30**	73.22	73.26	87.35
Avg Acc	79.69	**81.72**	52.13	63.50	79.50	77.67	79.49	93.83
Avg Rank	2.83	**1.83**	5.67	5.33	**1.83**	3.50		
epsilon	NA	NA	61.21	60.20	**85.89**	50.04	85.76	99.76
SVHN	19.89	20.72	21.76	23.59	**57.02**	18.79	57.38	87.12
gisette	74.72	75.80	89.25	90.93	**96.26**	50.84	96.36	99.93
spam	95.40	NA	97.15	97.29	**98.30**	97.60	98.54	99.90
sector	NA	NA	4.72	16.95	**73.56**	1.71	73.19	82.62
Avg Acc	38.00	19.30	54.82	57.79	**82.21**	43.80	82.25	93.87
Avg Rank	5.20	5.00	3.00	2.40	**1.00**	4.40		
Overall Acc	70.75	66.10	62.84	70.34	**83.60**	70.68	83.82	94.95
Overall Rank	3.68	2.76	4.97	3.65	**1.79**	4.15		

Different variants of CAND against best ensembles

TABLE IV
ACCURACY (%) FOR SELECTED ENSEMBLE SETTINGS AGAINST CAND VARIANTS. C=CAND(CPU TRAINED).

dataset	ARF 10, 60%	SRP 10, 60%	ARF 30, 10%	SRP 30, 10%	C $\lvert P\rvert$=10	C_{sub} $\lvert P\rvert$=30 $\lvert M\rvert$=10	$C_{sub}^{SB=0.6}$ $\lvert P\rvert$=30 $\lvert M\rvert$=10	C $\lvert P\rvert$=30
airlines	65.86	66.74	61.18	**66.77**	61.13	61.40	61.10	61.44
electricity	89.87	89.14	58.09	84.57	84.97	90.21	90.59	**91.55**
kdd99	99.96	99.97	99.94	**99.98**	99.92	99.96	99.96	99.96
WISDM	85.28	85.36	78.90	85.86	72.72	89.68	**90.54**	88.85
covtype	94.49	**95.28**	86.09	91.35	88.91	94.37	93.79	93.77
nomao	97.08	97.23	97.16	97.33	97.15	**97.61**	97.13	97.56
Avg Acc	88.76	**88.95**	80.23	87.64	84.13	88.87	88.85	88.86
Avg Rank	4.58	3.17	6.83	3.67	7.00	**3.08**	4.42	3.25
AGR_a	85.91	**92.44**	64.45	79.71	89.64	89.31	88.16	89.70
AGR_g	79.99	**87.55**	62.98	75.99	87.19	87.17	86.83	87.23
RBF_f	**71.20**	70.84	30.01	52.17	67.20	67.06	66.84	66.98
RBF_m	84.70	83.19	30.01	68.07	85.77	85.18	84.69	**85.78**
LED_a	73.92	73.49	62.55	53.26	**74.03**	**74.03**	73.89	74.02
LED_g	73.06	72.73	62.79	51.81	73.27	73.28	73.10	**73.30**
Avg Acc	78.13	**80.04**	52.13	63.50	79.52	79.34	78.92	79.50
Avg Rank	4.33	3.67	7.67	7.33	2.58	3.08	5.00	**2.33**
epsilon	50.26	50.21	61.21	60.20	85.86	**85.89**	83.11	**85.89**
SVHN	19.52	20.35	21.76	23.59	55.60	55.08	54.12	**57.02**
gisette	82.36	82.17	89.25	90.93	**96.38**	96.28	94.40	96.26
spam	96.12	96.22	97.15	97.29	98.03	**98.42**	97.64	98.30
sector	0.74	0.80	4.72	16.95	67.51	71.61	72.20	**73.56**
Avg Acc	49.80	49.95	54.82	57.79	80.68	81.46	80.29	**82.21**
Avg Rank	7.60	7.40	5.80	5.20	2.60	2.10	3.60	**1.70**
Overall Acc	73.55	74.34	62.84	70.34	81.49	83.33	82.83	**83.60**
Overall Rank	5.38	4.59	6.82	5.41	4.15	2.79	4.38	**2.47**

TABLE V
WALL TIME (S) FOR SELECTED ENSEMBLE SETTINGS AGAINST CAND VARIANTS. C=CAND(CPU TRAINED).

dataset	ARF 10, 60%	SRP 10, 60%	ARF 30, 10%	SRP 30, 10%	C $\lvert P\rvert$=10	C_{sub} $\lvert P\rvert$=30 $\lvert M\rvert$=10	$C_{sub}^{SB=0.6}$ $\lvert P\rvert$=30 $\lvert M\rvert$=10	C $\lvert P\rvert$=30
airlines	127.56	208.50	**22.46**	381.44	15888.01	9784.03	7626.94	28999.15
electricity	6.10	9.95	**3.47**	24.17	120.99	81.10	67.75	196.99
kdd99	**49.67**	111.36	100.27	151.76	1853.02	2201.20	844.08	5311.52
WISDM	3.07	3.92	**2.77**	3.57	20.21	15.76	11.17	41.64
covtype	140.12	252.40	**123.21**	339.30	2769.45	1682.24	949.38	4981.91
nomao	13.59	26.15	**10.01**	18.89	185.05	140.53	42.49	384.48
Avg	56.69	102.05	**43.70**	153.19	3472.79	2317.48	1590.30	6652.62
Avg Rank	1.83	3.33	**1.17**	3.67	6.83	6.17	5.00	8.00
AGR_a	113.89	253.63	**45.16**	3137.58	4165.67	3286.07	2025.02	7680.91
AGR_g	125.14	298.59	**46.13**	2688.19	4161.60	3410.06	2131.33	7679.61
RBF_f	105.93	249.15	**54.87**	363.33	2852.29	2331.26	1992.29	4850.71
RBF_m	102.48	216.62	**54.08**	352.31	2855.15	2121.74	1691.46	4858.58
LED_a	**73.50**	168.49	135.07	675.34	4395.22	3213.00	2335.20	6599.98
LED_g	**74.15**	171.34	137.01	634.07	4452.85	3044.41	2246.57	6593.23
Avg	99.18	226.30	**78.72**	1308.47	3813.80	2901.09	2070.31	6377.17
Avg Rank	1.67	3.00	**1.33**	4.33	7.00	6.00	4.67	8.00
epsilon	2006.44	2298.30	**1170.11**	1464.35	4987.39	5059.79	2481.79	11576.26
SVHN	1033.51	2392.79	**403.31**	1053.32	2739.18	2044.38	1787.48	5549.00
gisette	246.95	272.70	178.46	**153.06**	560.66	678.26	187.56	1332.19
spam	3144.16	3742.61	5163.25	**1759.17**	10414.31	8410.42	2174.42	25601.24
sector	112244.68	88245.67	38638.48	20272.19	15046.42	10573.59	**8981.35**	30843.94
Avg	23735.15	19390.41	9110.72	4940.42	6749.59	5353.29	**3122.52**	14980.53
Avg Rank	4.00	5.20	3.00	**2.20**	5.80	5.40	3.00	7.40
Overall	7035.94	5818.95	2722.83	**1968.94**	4556.91	3416.34	2210.37	9004.78
Overall Rank	2.41	3.76	**1.76**	3.47	6.59	5.88	4.29	7.82

Effect of mini-batch size and GPU training

TABLE VII
EFFECT OF MINI-BATCH SIZE ON ACCURACY AND WALL TIME (S) FOR $\text{CAND}_{sub}(|M|=10, |P|=30)$.

	Acc					Wall Time				
device	CPU	GPU				CPU	GPU			
batch size	1	1	4	16	32	1	1	4	16	32
airlines	61.4	**61.44**	61.21	61.21	61	9784.03	5090.33	1104.13	405.24	**291.46**
electricity	90.21	**91.51**	86.94	79.28	76.11	81.1	341.76	95.15	37.43	**27.69**
kdd99	99.96	**99.96**	99.93	99.85	99.78	2201.2	6833.39	1950.46	979.91	**816.73**
WISDM	**89.68**	89.55	78.64	64.36	60.39	15.76	41.55	14.89	7.8	**6.57**
covtype	**94.37**	93.74	89.52	88.15	88.26	1682.24	5027.07	1254.86	491.59	**364.15**
nomao	97.61	**97.62**	97.46	96.31	95.85	140.53	248.85	72.4	30.15	**23.02**
Avg	88.87	**88.97**	85.62	81.53	80.23	2317.48	2930.49	748.65	325.35	**254.94**
Avg rank	1.67	**1.33**	3	4.17	4.83	4	4.83	3.17	2	**1**
AGR_a	89.31	**89.35**	89.13	89.05	88.91	3286.07	8428.79	2178.76	749.77	**534.47**
AGR_g	87.17	87.11	87.11	**87.31**	87.23	3410.06	9227.59	2213.58	761.32	**537.64**
RBF_f	67.06	67.05	**67.28**	60.71	59.71	2331.26	9412.32	2176.72	802.27	**584.56**
RBF_m	85.18	85.22	85.61	**86.13**	86.03	2121.74	10264.88	2226.64	800.08	**582.6**
LED_a	74.03	**74.07**	74.03	74.03	74	3213	10514.25	2192.45	878.36	**656.89**
LED_g	73.28	**73.3**	73.29	73.3	73.3	3044.41	9421.42	2198.36	881.17	**660.23**
Avg	79.34	79.35	**79.41**	78.42	78.2	2901.09	9544.88	2197.75	812.16	**592.73**
Avg rank	3.5	**2.33**	3	2.5	3.67	3.83	5	3.17	2	**1**
epsilon	85.89	85.74	**85.92**	85.52	85.62	5059.79	807.52	332	205.65	**180.77**
SVHN	55.08	**55.34**	54.6	49.54	53.11	2044.38	241.18	109.96	75.3	**69.67**
gisette	**96.28**	96.1	96.09	96.14	95.66	678.26	61.72	34.53	26.3	**24.75**
spam	**98.42**	98.38	97.52	96.93	94.85	8410.42	401.02	284.44	244.99	**237.41**
sector	71.61	68.69	73.29	72.72	**73.37**	10573.59	429.81	213.59	152.86	**140.67**
Avg	81.46	80.85	**81.48**	80.17	80.52	5353.29	388.25	194.9	141.02	**130.66**
Avg rank	**2**	2.8	2.6	3.8	3.8	5	4	3	2	**1**
Overall	**83.33**	83.19	82.21	80.03	79.6	3416.34	4517.26	1097.23	442.95	**337.61**
Overall rank	2.41	**2.12**	2.88	3.47	4.12	4.24	4.65	3.12	2	**1**

References

Ashfahani, A.; and Pratama, M. 2019. Autonomous deep learning: Continual learning approach for dynamic environments. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, 666–674. SIAM.

Ashfahani, A.; Pratama, M.; Lughofer, E.; and Ong, Y.-S. 2020. DEVDAN: Deep evolving denoising autoencoder. *Neurocomputing*, 390: 297–314.

Bifet, A.; and Gavalda, R. 2007. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*, 443–448. SIAM.

Bifet, A.; Gavaldà, R.; Holmes, G.; and Pfahringer, B. 2018. *Machine learning for data streams: with practical examples in MOA*. MIT Press.

Bifet, A.; Holmes, G.; Kirkby, R.; and Pfahringer, B. 2010. MOA: Massive Online Analysis. *J. Mach. Learn. Res.*, 11: 1601–1604.

Cassales, G.; Gomes, H.; Bifet, A.; Pfahringer, B.; and Senger, H. 2020. Improving Parallel Performance of Ensemble Learners for Streaming Data Through Data Locality with Mini-Batching. In *IEEE International Conference on High Performance Computing and Communications (HPCC)*.

Das, M.; Pratama, M.; Savitri, S.; and Zhang, J. 2019. Muse-rnn: A multilayer self-evolving recurrent neural network for data stream classification. In *2019 IEEE International Conference on Data Mining (ICDM)*, 110–119. IEEE.

Gomes, H. M.; Barddal, J. P.; Enembreck, F.; and Bifet, A. 2017a. A survey on ensemble learning for data stream classification. *ACM Computing Surveys (CSUR)*, 50(2): 1–36.

Gomes, H. M.; Bifet, A.; Read, J.; Barddal, J. P.; Enembreck, F.; Pfahringer, B.; Holmes, G.; and Abdessalem, T. 2017b. Adaptive random forests for evolving data stream classification. *Machine Learning*, 106(9): 1469–1495.

Gomes, H. M.; Read, J.; and Bifet, A. 2019. Streaming random patches for evolving data stream classification. In *2019 IEEE International Conference on Data Mining (ICDM)*, 240–249. IEEE.

Gomes, H. M.; Read, J.; Bifet, A.; Barddal, J. P.; and Gama, J. 2019. Machine learning for streaming data: state of the art, challenges, and opportunities. *ACM SIGKDD Explorations Newsletter*, 21(2): 6–22.

Ikonomovska, E.; Gama, J.; and Džeroski, S. 2011. Learning model trees from evolving data streams. *Data mining and knowledge discovery*, 23(1): 128–168.

Pratama, M.; Za'in, C.; Ashfahani, A.; Ong, Y. S.; and Ding, W. 2019. Automatic construction of multi-layer perceptron network from streaming examples. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1171–1180.

Veloso, B.; and Gama, J. 2020. Self Hyper-parameter Tuning for Stream Classification Algorithms. In *IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning*, 3–13. Springer.

Veloso, B.; Gama, J.; Malheiro, B.; and Vinagre, J. 2021. Hyperparameter self-tuning for data streams. *Information Fusion*, 76: 75–86.

Zhao, P.; Wang, X.; Xie, S.; Guo, L.; and Zhou, Z.-H. 2019. Distribution-free one-pass learning. *IEEE Transactions on Knowledge and Data Engineering*.