Towards AutoML in the Presence of Drift*

Jorge G. Madrid, Hugo Jair Escalante & Eduardo Morales

Instituto Nacional de Astrofísica, Óptica y Electrónica

LXAI Workshop @ NeurIPS 2018, Montréal, Canada

* Jorge G Madrid, Hugo J Escalante, Eduardo Morales, Wei Wei Tu, Yang Yu, Lisheng Sun-Hosoya, Isabelle Guyon and Michele Sebag Towards AutoML in the presence of Drift: first results. AutoML@ICML2018 Workshop, Stockholm, Sweden, 2018

INAOE

National Institute of Astrophysics, Optics and Electronics (INAOE)

Introduction

AutoML

Traditional ML.

AutoML

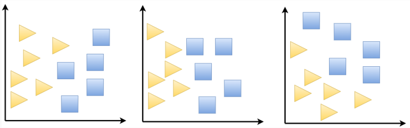
AutoML.

Concept Drift

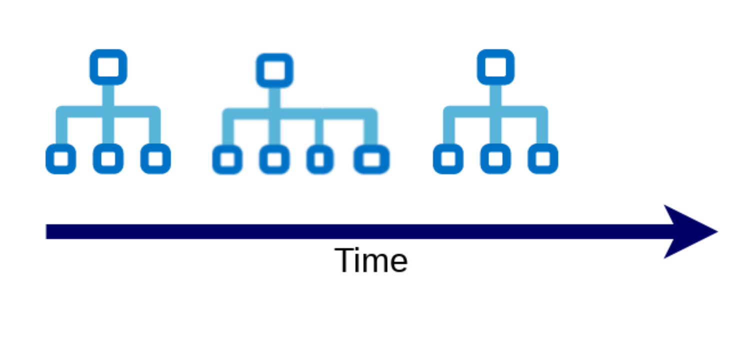
Batches of data in many real-world applications may be arriving daily, weekly...

Data distributions are changing over time:

- On-line advertising
- Recommendation systems
- Spam filtering
- Econometrics

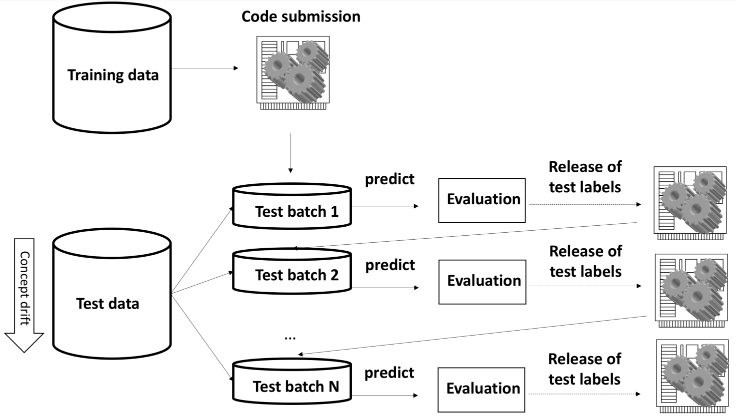


AutoML in the presence of drift



The model adapts autonomously over time.

AutoML in the presence of drift

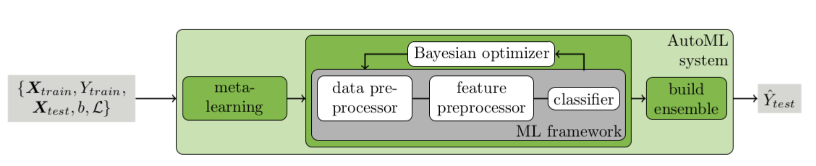


Evaluation scenario considered in the AutoML3 challenge.

Our proposal

Extending Autosklearn

- State-of-the art AutoML for supervised learning
- Winner of AutoML challenges
- Uses standard library (scikit-learn)
- Robust and open source



1

[1] Feurer, Matthias, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, and Frank Hutter. "Efficient and robust automated machine learning." In Advances in Neural Information Processing Systems, pp. 2962-2970. 2015.

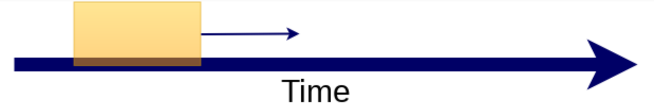
Proposed Extension

1. With first batch: full model selection with Autosklearn
2. For each batch until end:
   - if Concept Drift is detected
     2.1 Adapt model
     2.2 Reset drift *detector*
   - else
     2.1 Go to next batch with current model

Drift Detection

FHDDM

- Based on Hoeffding's inequality
- State-of-the-art drift detector
- Uses a sliding window
- What size for the window?



2

[2] Pesaranghader, Ali, and Herna L. Viktor. "Fast hoeffding drift detection method for evolving data streams." Joint European conference on machine learning and knowledge discovery in databases. Springer, Cham, 2016.

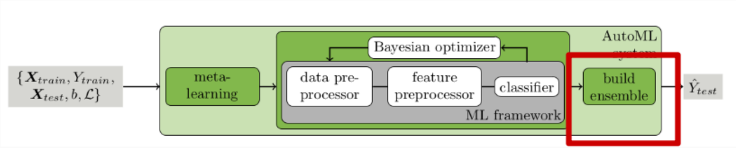
Adaption mechanisms

Model replacement

- Replace full model with a new one
- Uses data from every batch

Model management

- Ensemble weight update
  - **WU-latest:** Using latest batch
  - **WU-all:** Using all batches available
- Add new
  - Augments the ensemble with a new classifier



Results

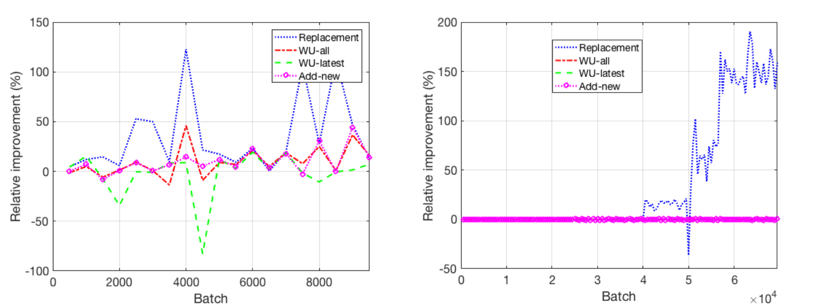
Datasets

Table 1: Datasets considered for experimentation.

Concept drift datasets		
Dataset	instances	attributes
Chess	503	8
Poker	100,000	10
Electricity	45,312	8
Stagger	70,000	3
AutoML2 challenge data sets		
Dataset	instances	attributes
PM	49,964	89
RH	60,042	76
RI	57,306	113
RL	56,209	22
RM	55,239	89

Relative Improvement

Performance of drift aware AutoML variants: Chess, Stagger



Benchmark Results

Table 2: Results on benchmark data.

Method	Electricity	Poker	Chess	Stagger	Rank
Base	67.15	67.97	38.23	54.09	4.5
Replacement	**76.44**	**90.38**	**58.13**	**78.81**	1
WU-all	70.23	76.89	52.62	54.09	2.75
WU-latest	67.95	67.49	53.24	54.09	3.5
Add new model	67.47	74.98	47.28	54.14	3.25

AutoML2 Results

Table 3: Results on data from the AutoML2 challenge.

Method	PM*	RH	RI	RL	RM*
Base	0.433	0.192	0.299	0.340	0.264
Replacement	0.433	0.197	0.092	**0.478**	0.264
WU-all	0.433	**0.370**	0.199	0.212	0.264
WU-latest	0.433	0.270	**0.450**	0.405	0.264
Add new	0.433	0.298	0.184	0.277	0.264

Conclusions

- Promising results for these intuitive mechanisms
- Different types of drift require different *adaptive* mechanisms
- How to scale these mechanisms to work with millions of samples?