AutoML in CapyMOA

Hamidreza Saffari
SDA Project

Professor: Dr. Della Valle

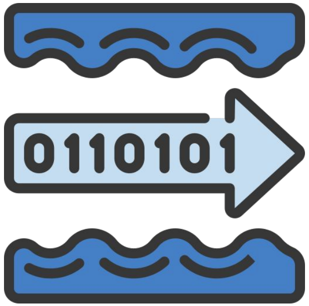
Supervisors: Lorenzo Iovine, Giacomo Ziffer

AGENDA

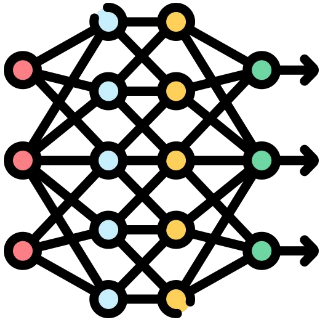
01

Introduction

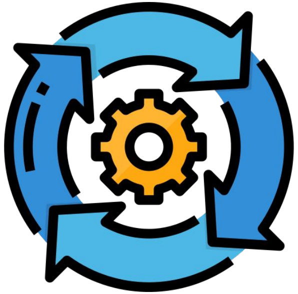
Problem: AutoML in **S**treaming **M**achine **L**earning



Streaming Data



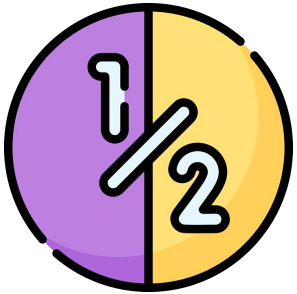
Machine Learning



Automatic Model Selection

Approaches: BanditClassifier and SuccessiveHalvingClassifier

Bandit Classifier



Successive Halving Classifier

02

Successive Halving Classifier

Approach: Successive Halving Classifier

- Efficiently selects models by **avoiding full training** on all data.
- Models are evaluated at stages called **"rungs"**.
- **Poor performers** are **discarded** early at each rung.
- **Promising models** receive **more resources** for further training.
- Designed to operate under a **fixed budget**.
- Aims to **maximize performance** with **minimal computation.**

Parameters of Successive Halving Classifier

- **Rungs:** A checkpoint where we evaluate and eliminate the worst performing models. It has a specific number of budget.
- **Budget:** The total amount of training data we're allowed to use across all models in all rungs.
- **Active Models:** The models that are still in the competition
- **Eta:** A number that tells us how many models to eliminate at each step (when eta=2 → Halving).

03

Bandit Classifier

Approach: Bandit Classifier

- Each model is linked to **an arm.**
- At each training step, the **policy** determines which arm/model to select.
- The reward is **based** on the **model's performance** on the given sample.

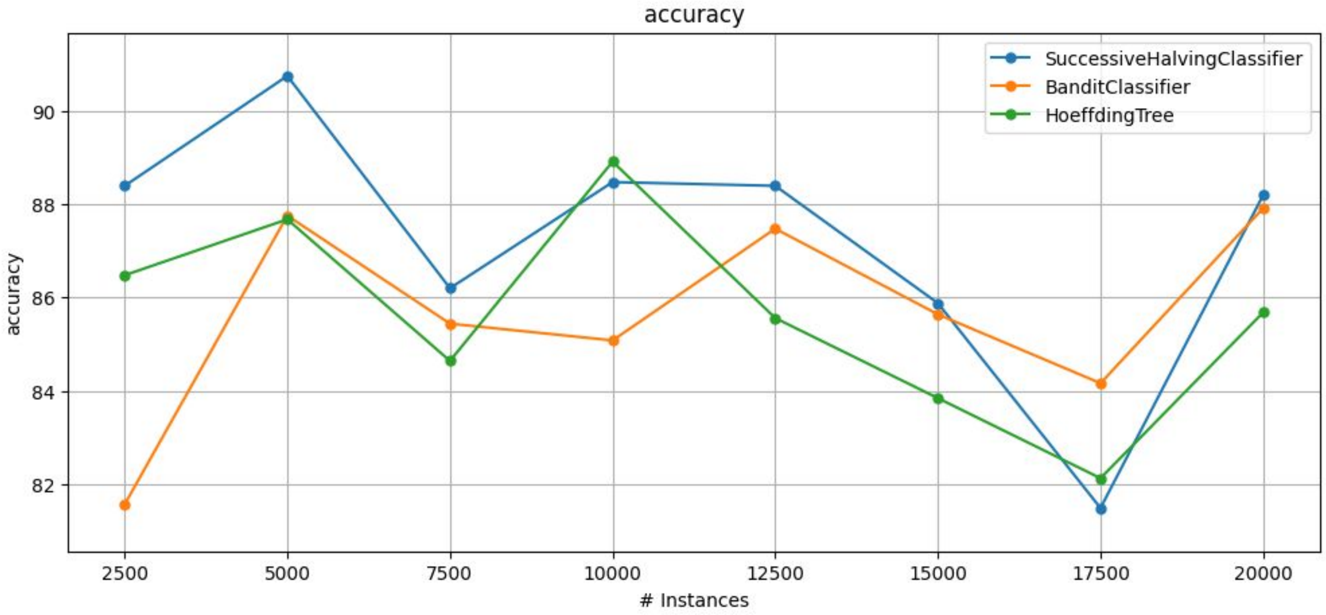
Epsilon Greedy Policy - Bandit Classifier

- **Best model** → probability **1 - epsilon.**
- **Other models** → probability **epsilon**.
- During the **burn-in period**, the policy always explores to gather initial information about **all models**.

04

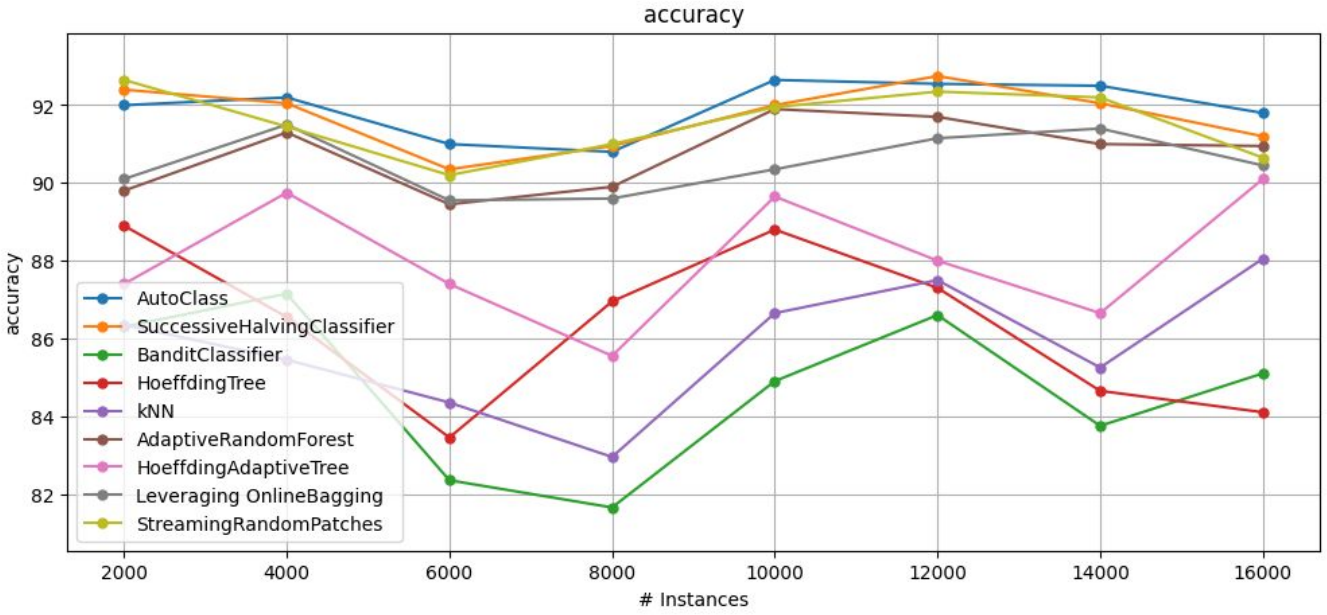
Comparison

Comparison of Models - Sanity Check



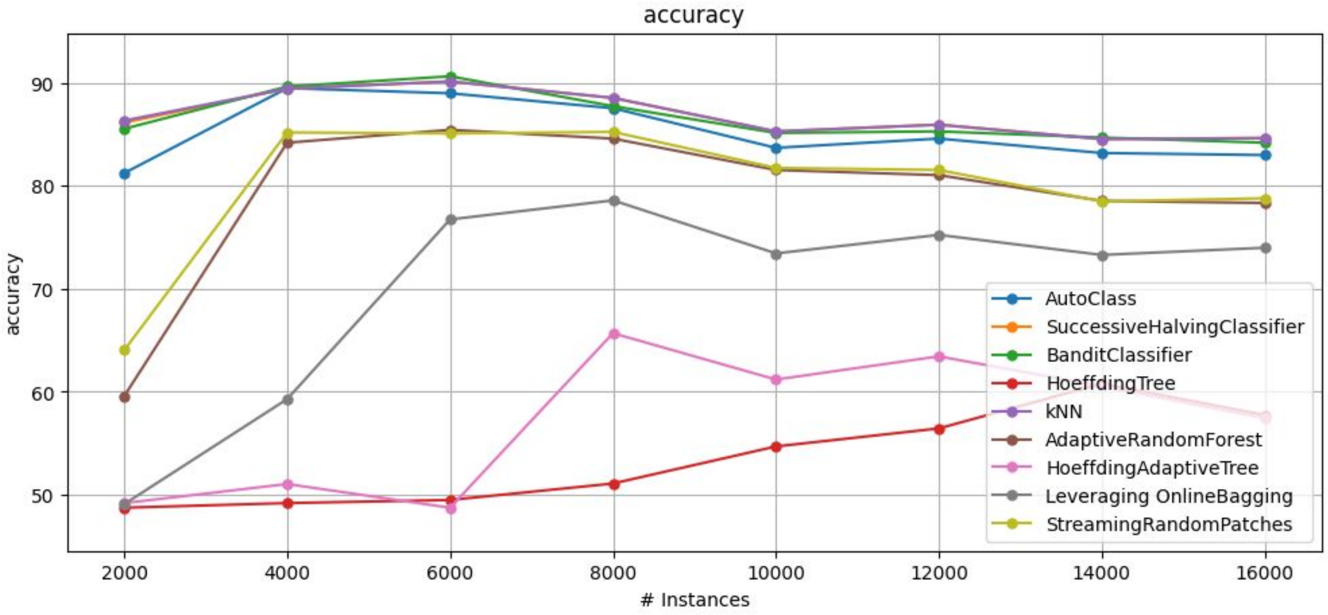

```
[SuccessiveHalving with 75 HT configs] Accuracy = 87.225, Time: 159.556s
[BanditClassifier with 75 HT configs] Accuracy = 85.800, Time: 100.866s
[Default HoeffdingTree] Accuracy = 85.615, Time: 0.139s
```

Comparison of Models – Electricity Stream



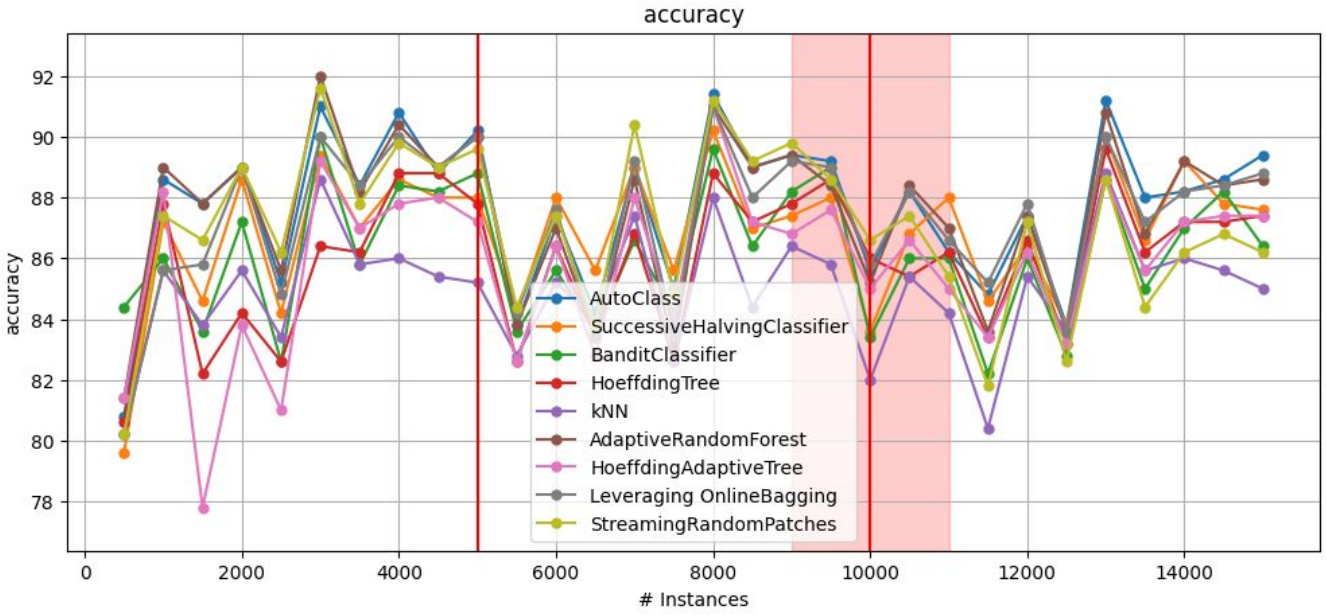

```
[AutoClass] Accuracy = 91.833, Time: 438.069s
[SuccessiveHalving] Accuracy = 91.600, Time: 264.252s
[BanditClassifier] Accuracy = 84.493, Time: 75.419s
```

Comparison of Models – RBFm Stream



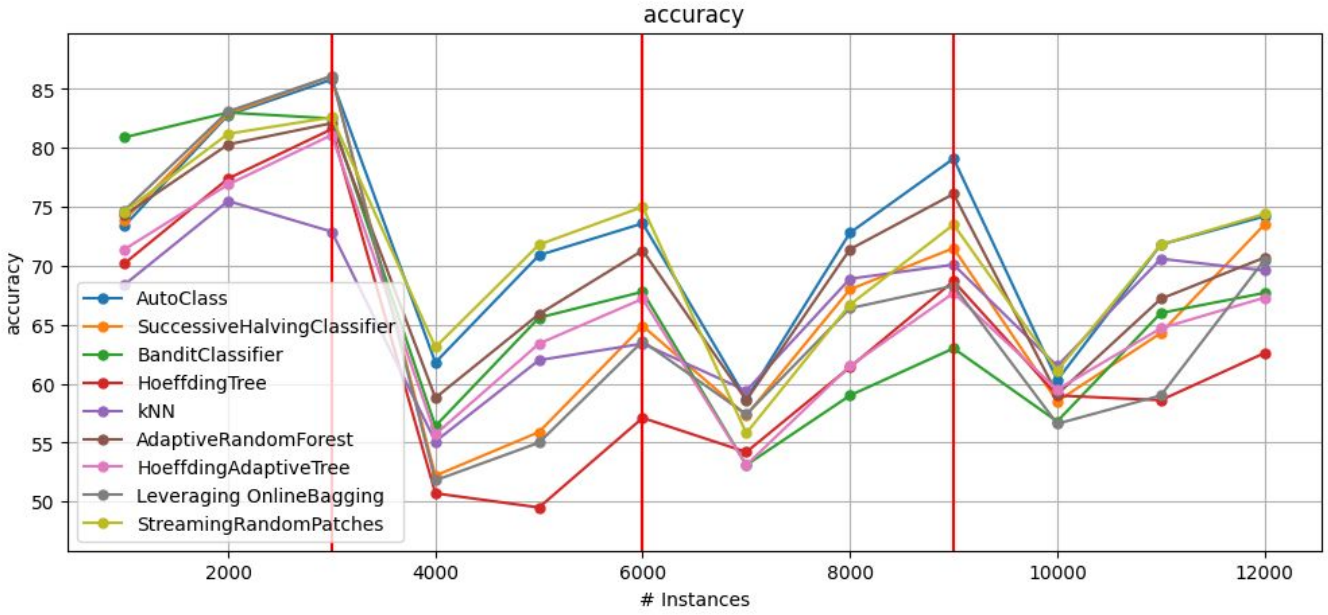

```
[AutoClass] Accuracy = 85.427, Time: 494.321s
[SuccessiveHalving] Accuracy = 87.013, Time: 250.490s
[BanditClassifier] Accuracy = 86.787, Time: 126.094s
```

Comparison of Models – SEA Stream with Drift




```
[AutoClass] Accuracy = 87.680, Time: 226.214s
[SuccessiveHalving] Accuracy = 86.820, Time: 178.488s
[BanditClassifier] Accuracy = 86.113, Time: 83.838s
```

Comparison of Models – Custom Stream with Concept Drift




```
[AutoClass] Accuracy = 72.100, Time: 776.176s
[SuccessiveHalving] Accuracy = 67.417, Time: 213.327s
[BanditClassifier] Accuracy = 66.817, Time: 105.817s
```

Comparison of Models

Stream	Model	Accuracy	Time (s)	Approximate Data Used (%)
Electricity	Bandit Classifier	84.49	**75**	**5**
	Successive Halving	91.6	264	10
	AutoClass	**91.83**	438	100
RBF	Bandit Classifier	86.78	**126**	**5**
	Successive Halving	**87.01**	250	10
	AutoClass	85.42	494	100
SEA - DataDrift	Bandit Classifier	86.11	**83**	**5**
	Successive Halving	86.82	178	20
	AutoClass	**87.68**	226	100
Custom Drift Stream	Bandit Classifier	66.81	**105**	**5**
	Successive Halving	67.41	213	20
	AutoClass	**72.1**	776	100

Time Calculation

- Bandit: burn_in*n_models + (max_instances-burn_in)
- Successive: budget=max_instances*K
- AutoClass: max_instances *n_models/2

Thank you!