

## ergebnisse für normalize and one hot encoding:

Stream	Type	ADL	Simple
Electricity	Full	91,62%	85.25%
Agrawal	no drift	57.26%	60.50%
Agrawal	one drift	64.12%	60.47%
Agrawal	three drifts	65.90%	52.48%
Agrawal	four drifts	64.30%	60.75%
SEA	no drift	65.27%	63.40%
SEA	one drift	88.44%	87.29%
SEA	three drifts	75.45%	87.23%
SEA	four drifts	74.65%	86.73%

## vergleich zum letzten mal:

### adl

	Type Agrawal	SEA
no drift	53.3%	54.06%
one drift	53.54%	<b>82.00%</b>
three drifts	56.68%	<b>80.95%</b>
drift back and forth	65.71%	<b>81.1%</b>

### simple

	Type Agrawal	SEA
no drift	58,19%	57.57%
one drift	<b>54.53%</b>	84.25%
three drifts	<b>64.31%</b>	83.94%
drift back and forth	<b>65.58%</b>	83.61%

**problem: aus irgendeinem grund waren die ergebnisse beim ersten lauf mit co2 plötzlich viel schlechter?**

**benutzt ray tune als search alg bereits hyper opt ?**

**hyperparameter search: im stream learning mit mehr instanzen**

**syntetische streams müssen auch größere anzahl instanzen providen**

**instanzen one hot encoden und normalisieren**

agrawal one hot encoden noch nicht in capymoa, schema -> wertebereich -> is nominal, und > 2,

**und normalisieren: bereits implementiert in capymoa feature standardisation:**

[vgl chapter 2: Using preprocessing from MOA \(filters\)](#)

jetzt ist es schon mit sklearn implementiert -.-

**größere grace periods auf mehr instanzen**

**Ergebnisse des disablen von hidden layern:**

keine zeit mehr für runs gehabt

**Accuracy Changes**

**Emission Changes**