



UW-Madison AutoML Decathlon Team
presentation:

PURE LEAF is all you need

Justin Kiefel, Kincannon Wilson, Ruiqi Geng, Sumanth Karnati

October 2, 2022



Datasets



- Input dimensions (Sequence, #channels, row_count, col_count)
- spherical** Input shape: (1, 3, 60, 60) Output shape: (100,) Dataset size: 40000 Dataset type: single-label
- ninapro** Input shape: (1, 16, 52, 1) Output shape: (18,) Dataset size: 2637 Dataset type: single-label
- ember** Input shape: (1, 2381, 1, 1) Output shape: (2,) Dataset size: 48000 Dataset type: single-label
- deepsea** Input shape: (1, 4, 1000, 1) Output shape: (36,) Dataset size: 59394 Dataset type: multi-label
- nottingham** Input shape: (1792, 88, 1, 1) Output shape: (88,) Dataset size: 693 Dataset type: multi-label
- navierstokes** Input shape: (20, 1, 64, 64) Output shape: (64, 64) Dataset size: 864 Dataset type: continuous
- crypto** Input shape: (3000, 13, 1, 1) Output shape: (600,) Dataset size: 1247 Dataset type: continuous

Auto-sklearn

```
def get_model(task_type:str, output_size: int):

    automl_model = None

    if (task_type == "single-label"): # Single-label classification

        automl_model = autosklearn.classification.AutoSklearnClassifier(
            time_left_for_this_task=120,
            per_run_time_limit=30,
            memory_limit=100000,
            tmp_folder="/tmp/autosklearn_classification_example_tmp",
        )

    if (task_type == "multi-label"): # Multi-label classification

        automl_model = autosklearn.classification.AutoSklearnClassifier(
            time_left_for_this_task=120,
            per_run_time_limit=30,
            memory_limit=100000,
        )

    if (task_type == "continuous"): # Regression

        automl_model = autosklearn.regression.AutoSklearnRegressor(
            time_left_for_this_task=120,
            per_run_time_limit=30,
            memory_limit=100000,
            tmp_folder="/tmp/autosklearn_regression_example_tmp",
        )

    return automl_model
```

```
# Installs
import os
os.system("pip install auto-sklearn")
os.system("pip install -U scikit-learn")
```

```
self.model.fit(
    x_train,
    y_train,
    dataset_name = "submission"
)
```

```
# get test predictions from the model
predictions = self.model.predict(x_test)
```

Our implementation & progress

- ❑ Errors on Colab
- ❑ Single-label with ember:
- ❑ Regression with crypto:

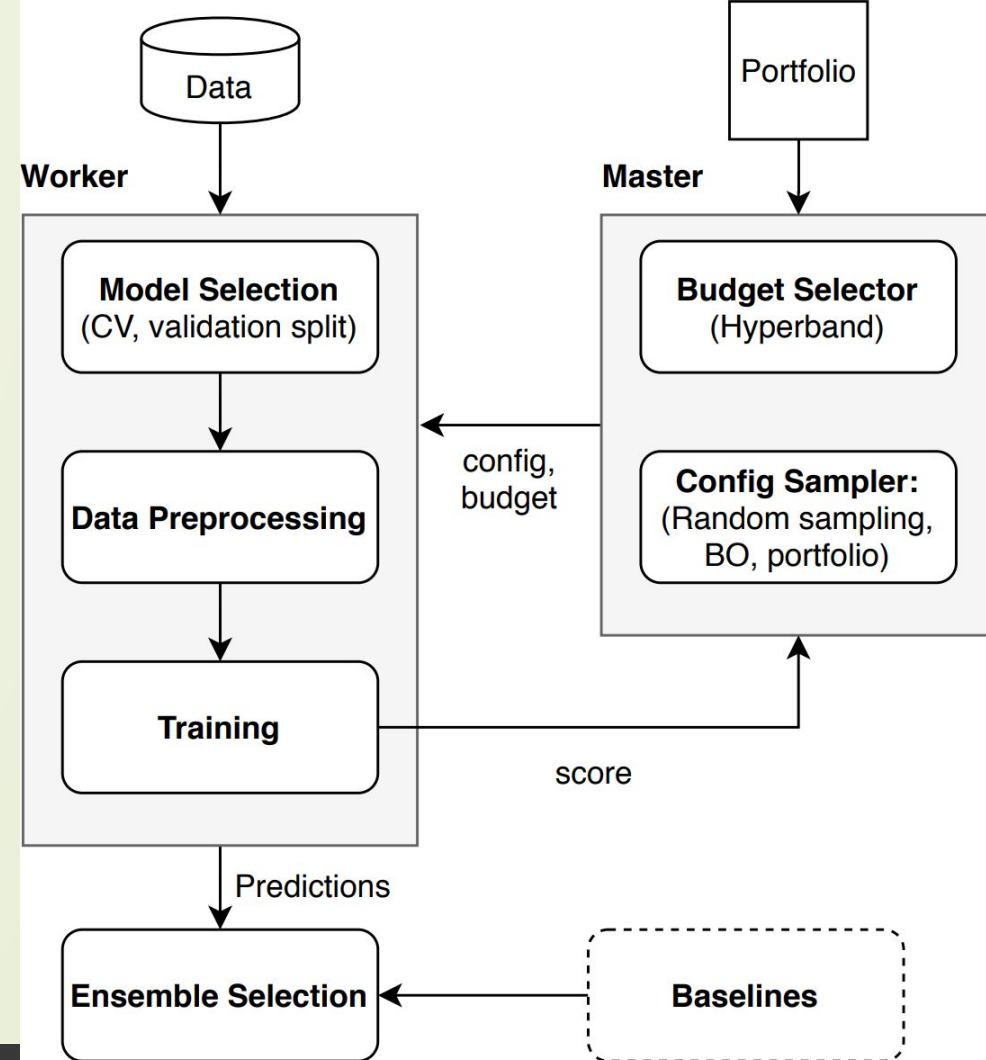
```
(48000, 2381) (48000,)  
[ERROR] [2022-10-02 16:22:41,391:Client-AutoML(1):submission] module 'sklearn' has no attribute 'utils'  
Traceback (most recent call last):  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/automl.py", line 658, in fit  
    X, y = self.InputValidator.transform(X, y)  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/data/validation.py", line 204, in transform  
    y_transformed = self.target_validator.transform(y)  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/data/target_validator.py", line 271, in transform  
    y_transformed = sklearn.utils.check_array(  
AttributeError: module 'sklearn' has no attribute 'utils'  
-----  
AttributeError                                Traceback (most recent call last)  
<ipython-input-15-7de54a2f3f09> in <module>  
    1 time_budget = 200  
----> 2 M.train(train_dataset, val_dataset=None, val_metadata=None, remaining_time_budget=time_budget)  
  
-----  
              ↳ 7 frames -----  
/usr/local/lib/python3.7/dist-packages/autosklearn/data/target_validator.py in transform(self, y)  
    269  
    270     # check array ensures correct numerical features for array  
--> 271     y_transformed = sklearn.utils.check_array(  
    272         y_transformed,  
    273         force_all_finite=True,  
  
AttributeError: module 'sklearn' has no attribute 'utils'
```

```
(1247, 39000) (1247, 600)  
[WARNING] [2022-10-02 16:41:20,599:Client-AutoMLSMBO(1)::submission] Could not find meta-data directory /usr  
[WARNING] [2022-10-02 16:41:22,770:Client-EnsembleBuilder] No runs were available to build an ensemble from  
Process pynisher function call:  
Traceback (most recent call last):  
  File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap  
    self.run()  
  File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run  
    self._target(*self._args, **self._kwargs)  
  File "/usr/local/lib/python3.7/dist-packages/pynisher/limit_function_call.py", line 133, in subprocess_fun  
    return_value = ((func(*args, **kwargs), 0))  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/ensemble_building/builder.py", line 583, in main  
    ensemble=ensemble, idx=iteration, seed=self.seed # type: ignore  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/automl_common/common/utils/backend.py", line 605,  
    pickle.dump(ensemble, fh)  
_pickle.PicklingError: Can't pickle <function r2_score at 0x7f8141601320>: it's not the same object as sklearn  
[WARNING] [2022-10-02 16:41:25,580:Client-EnsembleBuilder] No runs were available to build an ensemble from  
[WARNING] [2022-10-02 16:41:28,363:Client-EnsembleBuilder] No runs were available to build an ensemble from
```

```
[WARNING] [2022-10-02 16:43:01,255:Client-EnsembleBuilder] No runs were available to build an ensemble from  
[WARNING] [2022-10-02 16:43:03,084:Client-EnsembleBuilder] No runs were available to build an ensemble from  
[ERROR] [2022-10-02 16:43:04,985:Client-AutoML(1):submission] list index out of range  
Traceback (most recent call last):  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/automl.py", line 952, in fit  
    self._load_models()  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/automl.py", line 1584, in _load_models  
    self.ensemble_ = self._backend.load_ensemble(self._seed)  
  File "/usr/local/lib/python3.7/dist-packages/autosklearn/automl_common/common/utils/backend.py", line 590, in  
    with open(indices_files[-1], "rb") as fh:  
IndexError: list index out of range  
-----  
IndexError                                Traceback (most recent call last)  
<ipython-input-14-7de54a2f3f09> in <module>  
    1 time_budget = 200  
----> 2 M.train(train_dataset, val_dataset=None, val_metadata=None, remaining_time_budget=time_budget)  
  
-----  
              ↳ 7 frames -----  
/usr/local/lib/python3.7/dist-packages/autosklearn/automl_common/common/utils/backend.py in load_ensemble(self,  
seed)  
    588     indices_files.sort(key=lambda f: time.ctime(os.path.getmtime(f)))  
    589  
--> 590     with open(indices_files[-1], "rb") as fh:  
    591         ensemble_members_run_numbers = cast(AbstractEnsemble, pickle.load(fh))  
    592  
  
IndexError: list index out of range
```


Auto-PyTorch

- Neural Architecture Search + Hyperparameter optimization
- Multi-Fidelity MetaLearning for Efficient and Robust AutoDL
- Warmstarting
- Bayesian optimization & Hyperband
- Parallel optimization



```
2022-10-02 18:11:43,068 INFO score.py: Start scoring for task: ember
2022-10-02 18:11:43,069 INFO score.py: solution shape=(12000, 2)
Traceback (most recent call last):
  File "/content/automl_decathlon_starter_kit/scoring/score.py", line 529, in <module>
    score, duration = scoring_main(args, task)
  File "/content/automl_decathlon_starter_kit/scoring/score.py", line 444, in scoring_main
    scoring_functions=scoring_functions,
  File "/content/automl_decathlon_starter_kit/scoring/score.py", line 347, in __init__
    self.fetch_ingestion_info()
  File "/content/automl_decathlon_starter_kit/scoring/score.py", line 372, in fetch_ingestion_info
    raise IngestionError("[-] Failed: scoring didn't detected start.txt")
__main__.IngestionError: [-] Failed: scoring didn't detected start.txt
```



AutoGluon



```
from autogluon.tabular import TabularDataset, TabularPredictor
```

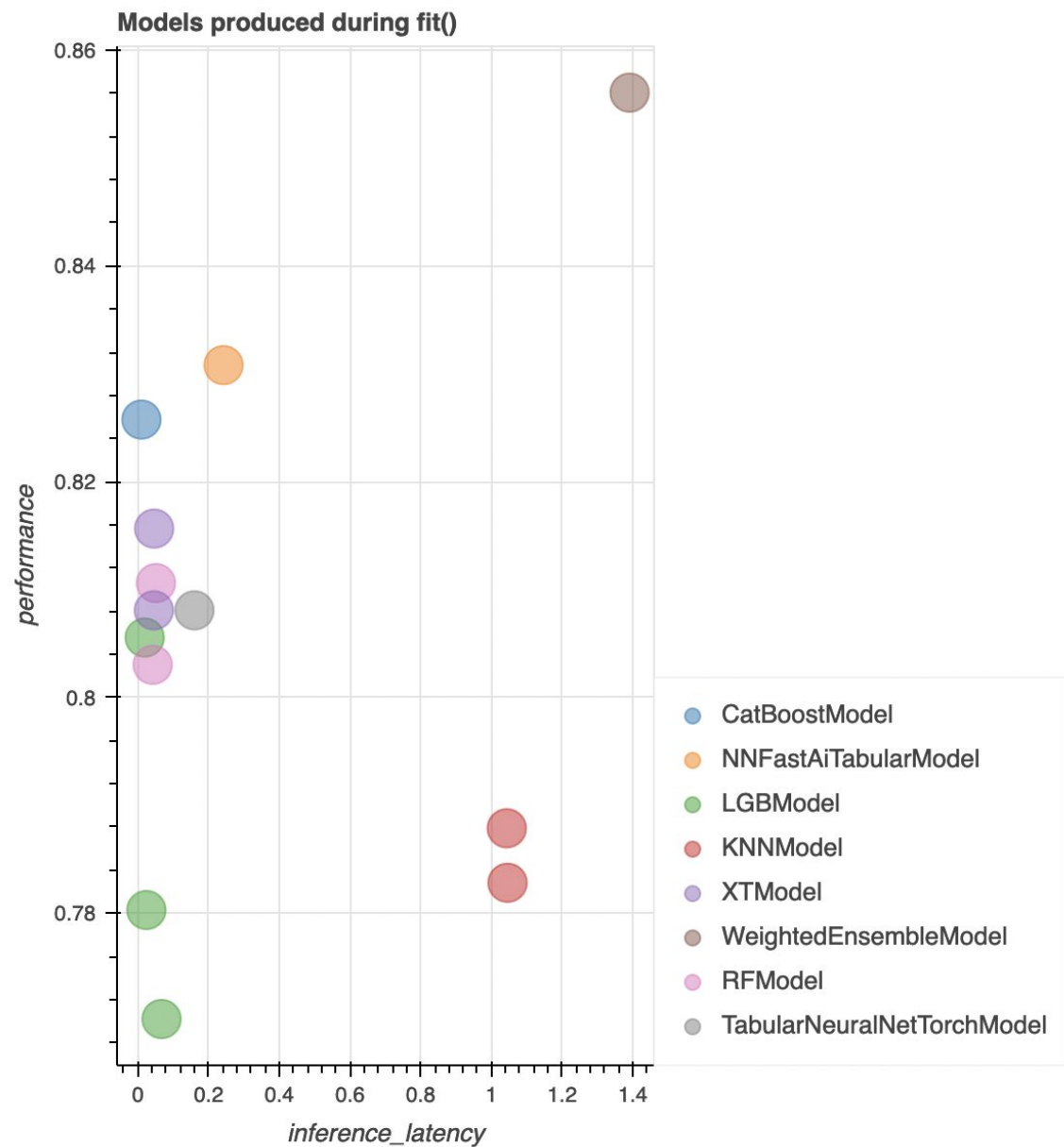
```
training_data = pandas.DataFrame(x_train)
training_data['y_labels'] = y_train
self.model = TabularPredictor(label = 'y_labels')
excluded_models_final = ['XGB']
excluded_models_test = ['XGB', 'NN_TORCH', 'GBM', 'CAT', 'FASTAI', 'XT', 'KNN']
self.model = self.model.fit(train_data = TabularDataset(training_data), excluded_model_types=excluded_models_test)
results = self.model.fit_summary(show_plot = True)
```

```
predictions = self.model.predict(TabularDataset(x_test))
```

Results: AutoGluon vs. Baseline

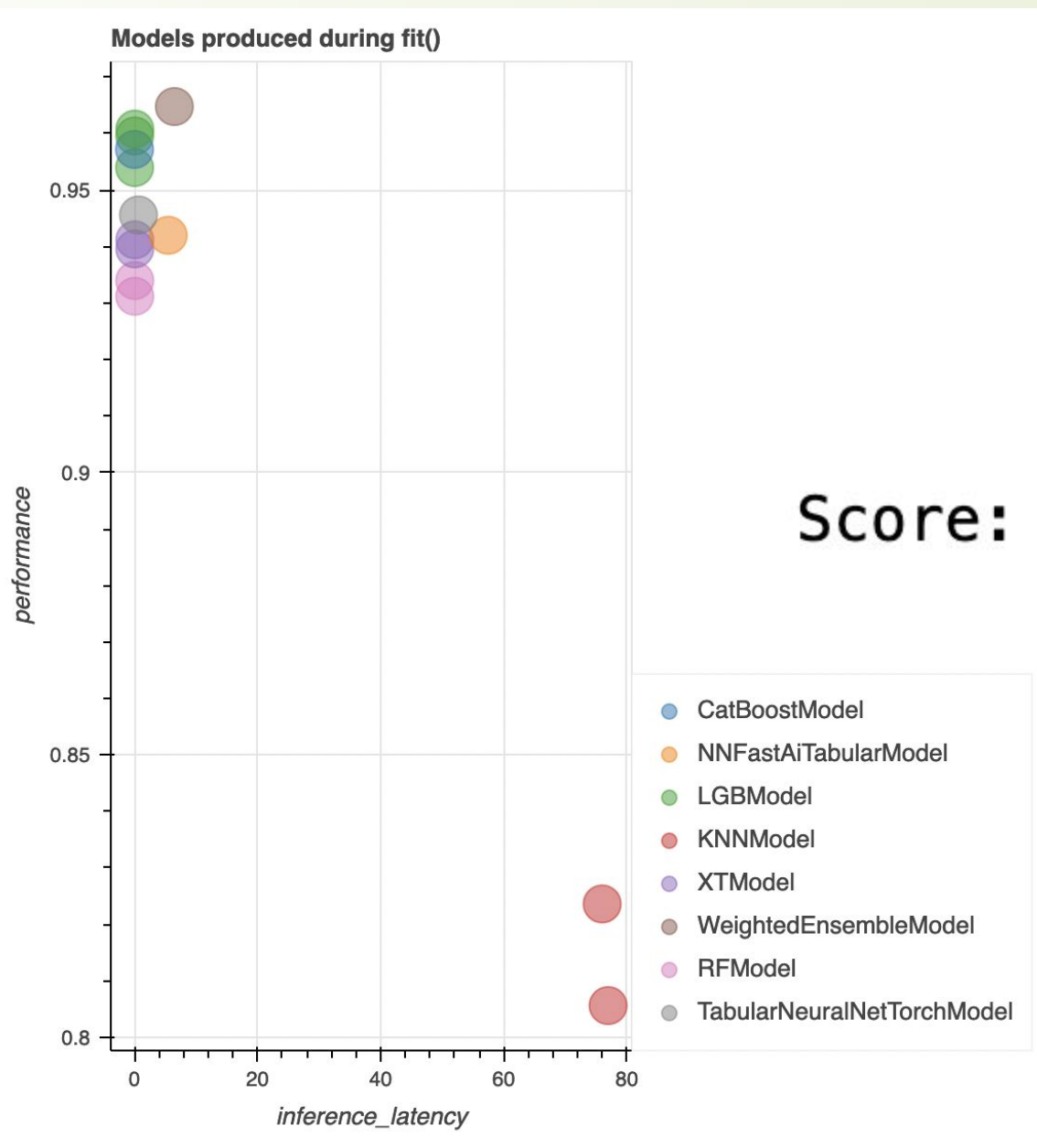
Datasets	AutoGluon	Baseline models			
		Combo (XGBoost- sequence, ResNet - others)	Wide ResNet	Linear	XGBoost
ninapro	0.1894	0.1409	0.1409	0.1864	0.2606
ember	0.0410	0.0756	-	0.4636	0.0756

ninapro results



Score: 0.18939393939393945

ember results

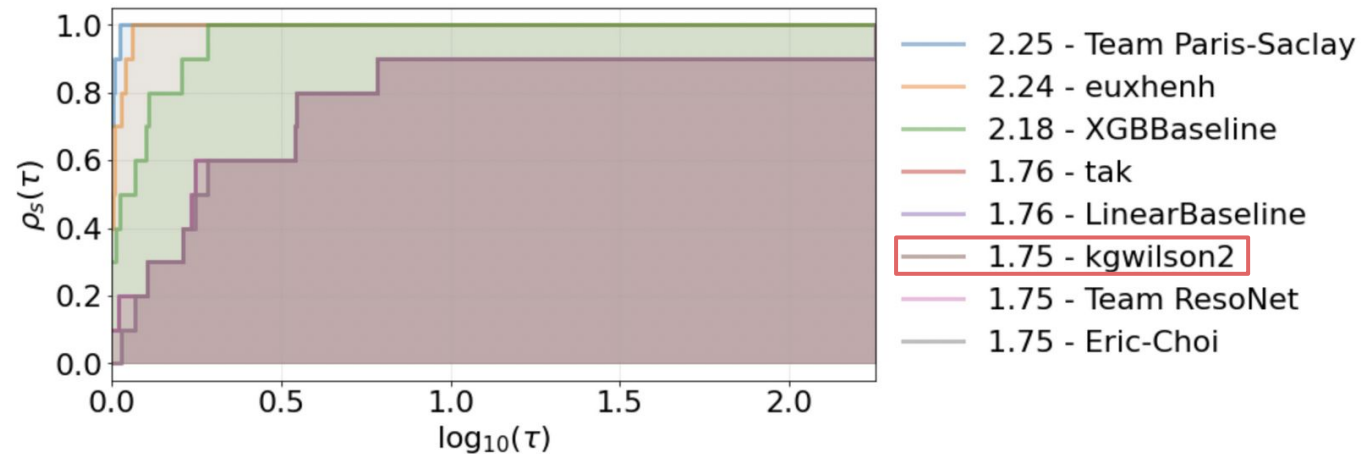


Score: 0.041000000000000036

Good news!

Leaderboard

	rank	AUP	navierstokes	spherical	ninapro	cosmic	ecg	deepsea	nottingham	crypto	ember	fsd50k
Team Paris-Saclay	1	2.253576	0.284792	0.7718	0.110774	0.468815	0.420773	0.427120	0.260695	0.005631	0.066985	0.602274
euxhenh	2	2.242872	0.284792	0.7677	0.121396	0.470865	0.483353	0.428270	0.260695	0.005631	0.066985	0.611081
XGBBaseline	3	2.177617	0.284792	0.9660	0.210926	0.468936	0.539995	0.499239	0.260695	0.005529	0.068560	0.963624
tak	4	1.763196	0.999871	0.9771	0.189681	0.444158	0.742969	0.445968	1.581103	0.999022	0.232645	0.971508
LinearBaseline	5	1.763093	0.999871	0.9771	0.189681	0.444158	0.742969	0.445968	1.581103	0.999022	0.232645	0.973802
Eric-Choi	6	1.751047	0.999871	0.9771	0.210926	0.470865	0.742969	0.499239	1.581103	0.999022	0.232645	0.973802
kgwilson2	7	1.751047	0.999871	0.9771	0.210926	0.470865	0.742969	0.499239	1.581103	0.999022	0.232645	0.973802
Team ResoNet	8	1.751047	0.999871	0.9771	0.210926	0.470865	0.742969	0.499239	1.581103	0.999022	0.232645	0.973802



Here are your submissions to date (✓ indicates submission on leaderboard):

#	SCORE	FILENAME	SUBMISSION DATE	SIZE (BYTES)	STATUS	✓	
1	---	PureLeaf_Submission_1.zip	10/02/2022 21:12:57	312083	Finished	✓	+

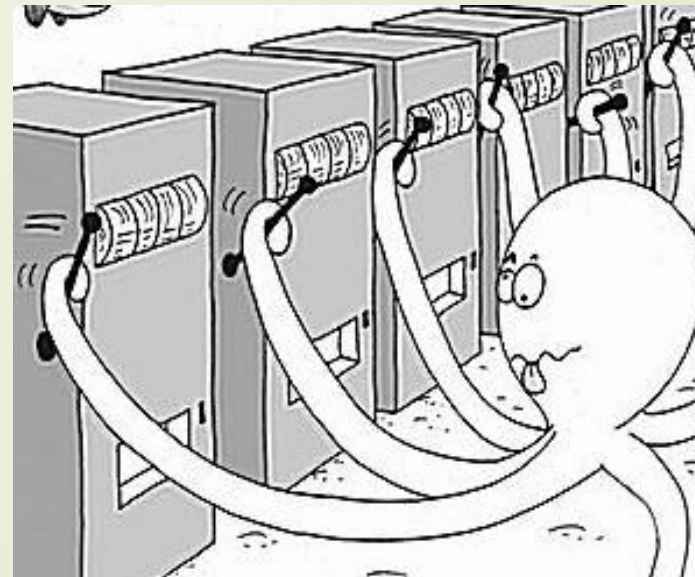
Sad news . . .

```
import os
os.system("pip install torch==1.12+cpu torchvision==0.13.0+cpu torchtex==0.13.0 -f https://download.pytorch.org/whl/cpu/torch_stable.html")
os.system("pip install autogluon")
os.system("pip install -U scikit-learn==1.0.2")
from autogluon.tabular import TabularDataset, TabularPredictor
```

```
ImportError: cannot import name '_check_feature_names_in' from 'sklearn.utils.validation' (/usr/local/lib/python3.8/dist-packages/sklearn/utils/validation.py)
2022-10-02 21:21:39,229 INFO ingestion.py: Ended ingestion for ember
```

Bandit Regulated Hyperparameter Tuning (BaRHT)

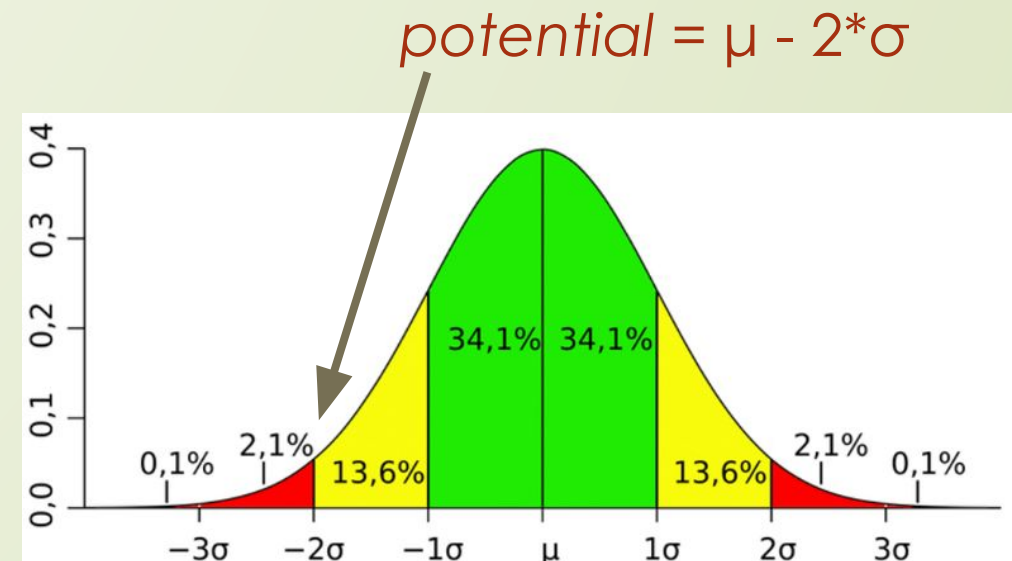
- ❑ **Challenge:** Optimize over machine learning algorithms **AND** hyperparameters with limited time
- ❑ **Approach:** Use a **multi-armed bandit algorithm** to allocate hyperparameter tuning time to more promising algorithms



Predicting Algorithm Potential


- **Algorithm Potential:** The 95th Percentile of performance for hyperparameter configurations
- Assume hyperparameter performances are **normally distributed**. (Fair for random search)
- Keep a running estimate of the mean, variance, and algorithm potential

$$\bar{x}_n = \frac{(n-1)\bar{x}_{n-1} + x_n}{n} = \bar{x}_{n-1} + \frac{x_n - \bar{x}_{n-1}}{n}$$
$$\sigma_n^2 = \frac{(n-1)\sigma_{n-1}^2 + (x_n - \bar{x}_{n-1})(x_n - \bar{x}_n)}{n}$$





BaHRT Algorithm

1. Choose ML Algorithm
 - a. Highest potential with probability $(1-\epsilon)$
 - b. Random with probability ϵ
 2. Train next hyperparameter configuration for algorithm
 3. Calculate validation metrics
 4. Update algorithm's mean, variance, and potential
 5. GOTO 1
- 

In Practice

XGBoost

```
[I 2022-10-02 16:26:44,031] Trial 0 finished with value: 0.06637001476489361 and parameters: {'booster': 'gblinear', 'lambda': 0.021347611909650784, 'alpha': 1.317144193674571e-06}. Best is trial 0 with value: 0.06637001476489361.
[I 2022-10-02 16:26:48,713] Trial 0 finished with value: 0.04846564488456466 and parameters: {'fit_intercept': True, 'alpha': 639.3896969010427}. Best is trial 0 with value: 0.04846564488456466.
[I 2022-10-02 16:26:54,188] Trial 1 finished with value: 0.04897043775779135 and parameters: {'fit_intercept': True, 'alpha': 759.562524026799}. Best is trial 0 with value: 0.04846564488456466.
[I 2022-10-02 16:26:59,369] Trial 2 finished with value: 0.048799219687273604 and parameters: {'fit_intercept': False, 'alpha': 717.6298820643271}. Best is trial 0 with value: 0.04846564488456466.
[I 2022-10-02 16:27:05,496] Trial 3 finished with value: 0.046112496745545156 and parameters: {'fit_intercept': True, 'alpha': 165.17225432004798}. Best is trial 3 with value: 0.046112496745545156.
[I 2022-10-02 16:27:10,125] Trial 4 finished with value: 0.046346089439874776 and parameters: {'fit_intercept': False, 'alpha': 211.99048485891643}. Best is trial 3 with value: 0.046112496745545156.
[I 2022-10-02 16:27:14,247] Trial 5 finished with value: 0.04583921056648525 and parameters: {'fit_intercept': False, 'alpha': 30.03289476656597}. Best is trial 5 with value: 0.04583921056648525.
```

Linear Regression

Using Different models

- ❑ Tried using PASA_NJU which was ranked 3rd in 2019 Auto DL competition.
- ❑ It did not work, PASA_NJU made their submission in 2019 Version TensorFlow and when I tried using that, it ran into multiple errors

#	User	Entries	Date of Last Entry	<Rank> ▲	Dataset 1 ▲	Dataset 2 ▲	Dataset 3 ▲	Dataset 4 ▲	Dataset 5 ▲	Compute Time	Detailed Results
1	DeepWisdom	10	03/14/20	1.2000	0.7987(1)	0.6426(2)	0.8980(1)	0.9072(1)	0.7777(1)	3:11:52	Learning Curve
2	DeepBlueAI	8	03/16/20	2.8000	0.7353(2)	0.5995(3)	0.8720(4)	0.8885(2)	0.7685(3)	1:36:20	Learning Curve
3	PASA_NJU	7	03/14/20	3.2000	0.7298(3)	0.5627(4)	0.8765(3)	0.8686(4)	0.7761(2)	1:46:32	Learning Curve

⏮ 1 frames ⏭

```
/content/automl\_decathlon\_starter\_kit/AutoDL/AutoDL\_sample\_code\_submission/model.py in infer_domain(metadata)
    100     metadata: an AutoDLMetadata object.
    101     """
--> 102     row_count, col_count = metadata.get_matrix_size(0)
    103     sequence_size = metadata.get_sequence_size()
    104     channel_to_index_map = metadata.get_channel_to_index_map()

AttributeError: 'DecathlonMetadata' object has no attribute 'get_matrix_size'
```


SVM

- ❑ Built SVM model both for continuous and single label .
- ❑ Tried training on the data, the single label data got into errors, as the data has multi-class, the svm classification can handle binary classification, which is included in the scikit learn package, so it did not work
- ❑ When I ran the model on continuous crypto data, its ran for more than 40 minutes without producing any results, So I discontinued the model.

```
def svm_model(task_type: str, output_size: int, random_state=None):  
    #  
  
    # Common model params  
    model_params = {  
        "C": 1.0,  
        "kernel": 'rbf',  
        "degree": 3,  
        "gamma": 'scale',  
        "coef0": 0.0,  
        "shrinking": True,  
        "probability": False,  
        "tol": 0.001,  
        # "cache_siz": 200,  
        "class_weight": None,  
        "verbose": False,  
        "max_iter": -1,  
        "decision_function_shape": 'ovr',  
        "break_ties": False,  
    }  
    if random_state:  
        model_params["random_state"] = random_state  
  
    # Cases
```

Changing parameters

- ❑ Results obtained by changing parameters in models and baseline models
- ❑ On Ninapro dataset

```
score = decathlon_scorer(solution, prediction, dataset)
print ("Score: ", score)
```

Score: 0.15000000000000002

```
score = decathlon_scorer(solution, prediction, dataset)
print ("Score: ", score)
```

Score: 0.14242424242424245

```
score = decathlon_scorer(solution, prediction, dataset)
print ("Score: ", score)
```

Score: 0.27575757575757576

```
score = decathlon_scorer(solution, prediction, dataset)
print ("Score: ", score)
```



Score: 0.28181818181818186



Future Directions (MetaLeaf)

- ❑ MetaLeaf: aggregate results from:
- ❑ - PyCaret
- ❑ - H2O AutoML
- ❑ - TPOT
- ❑ - Auto-sklearn
- ❑ - FLAML
- ❑ - EvalML
- ❑ - AutoKeras
- ❑ - Auto-ViML
- ❑ - AutoGluon
- ❑ - MLBox

Future Directions (Transformers)

GPT-3 can predict stock prices, classifications, and generate images.





Future Directions

GPT-4 == MetaLeaf-1





Thank you!

Justin Kiefel, Kincannon Wilson, Ruiqi Geng, Sumanth Karnati

October 2, 2022