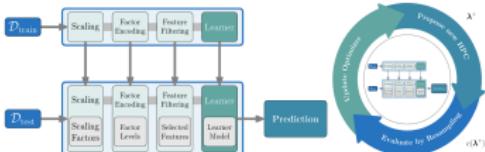


Introduction to Machine Learning

Hyperparameter Tuning Pipelines and AutoML



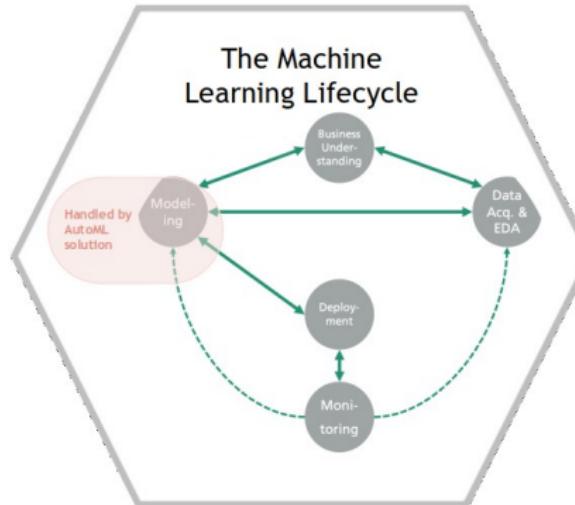
Learning goals



- Pipelines as connected steps of learnable operations
- Sequential pipeline
- Pipelines and DAGs

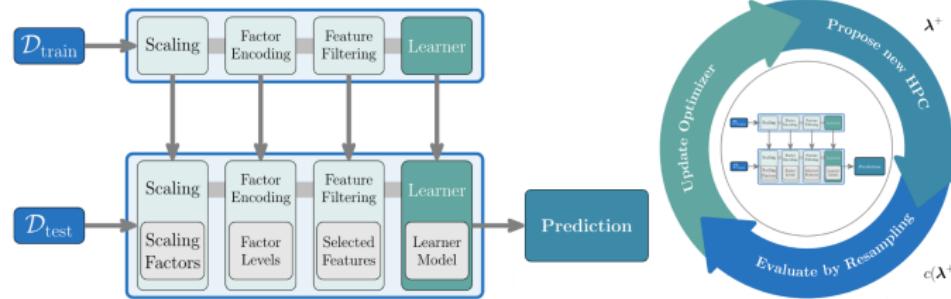
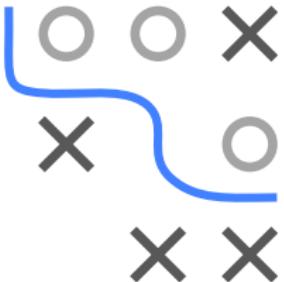
CASE FOR AUTOML

- More and more tasks are approached via data driven methods.
- Data scientists often rely on trial-and-error.
- The process is especially tedious for similar, recurring tasks.
- Not the entire machine learning lifecycle can be automated.



PIPELINES AND AUTOML

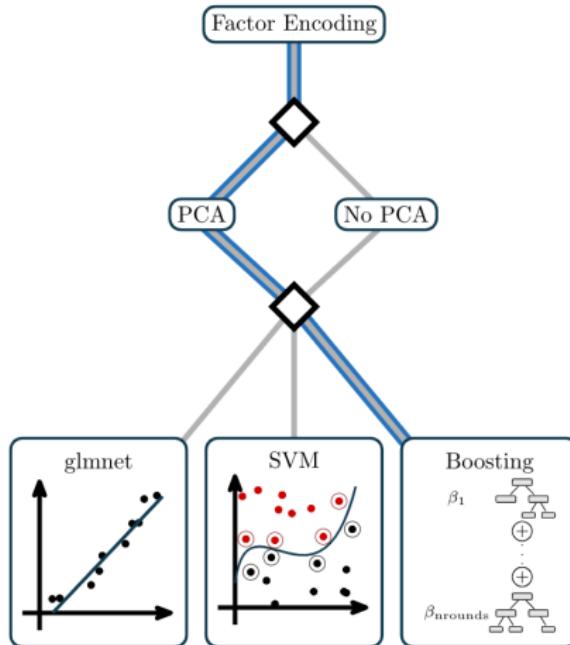
- ML typically has several data transformation steps before model fit
- If steps are in succession, data flows through sequential pipeline
- NB: Each node has a train and predict step and learns params
- And usually has HPs



Pipelines are required to embed full model building into CV to avoid overfitting and biased evaluation!

PIPELINES AND AUTOML

- Further flexibility by representing pipeline as DAG
- Single source accepts $\mathcal{D}_{\text{train}}$, single sink returns predictions
- Each node represents a preprocessing operation, a learner, a postprocessing operation or controls data flow
- Can be used to implement ensembles, operator selection,
...

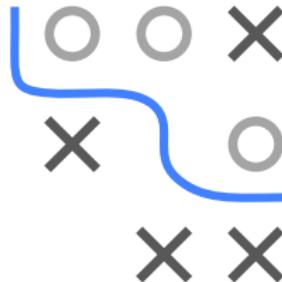


PIPELINES AND AUTOML

- HPs of pipeline are the joint set of all HPs of its contained nodes:

$$\tilde{\Lambda} = \tilde{\Lambda}_{\text{op},1} \times \cdots \times \tilde{\Lambda}_{\text{op},k} \times \tilde{\Lambda}_{\mathcal{I}}$$
 - HP space of a DAG is more complex:
 Depending on branching / selection
 different nodes and HPs are active
 → **hierarchical search space**

Search Space $\bar{\Lambda}$			
Name	Type	Bounds/Values	Trafo
encoding	C	one-hot, impact	
◊ pca	C	PCA, no PCA	
◊ learner	C	glmnet, SVM, Boosting	
<hr/>			
if learner = glmnet			
s	R	[−12, 12]	2^x
alpha	R	[0, 1]	—
<hr/>			
if learner = SVM			
cost	R	[−12, 12]	2^x
gamma	R	[−12, 12]	2^x
<hr/>			
if learner = Boosting			
eta	R	[−4, 0]	10^x
nrounds	I	{1, ..., 5000}	—
max.depth	I	{1, ..., 20}	—



A graph that includes many preprocessing steps and learner types can be flexible enough to work on a large number of data sets

Combining such graph with an efficient tuner is key in AutoML

AUTOML – CHALLENGES

- Most efficient approach?
- How to integrate human a-priori knowledge?
- How can we best (computationally) transfer “experience” into AutoML? Warmstarts, learned search spaces, etc.
- Multi-Objective goals, including model interpretability
- AutoML as a process is too much of a black-box, hurts adoption.

