TECHNISCHE UNIVERSITÄT DARMSTADT

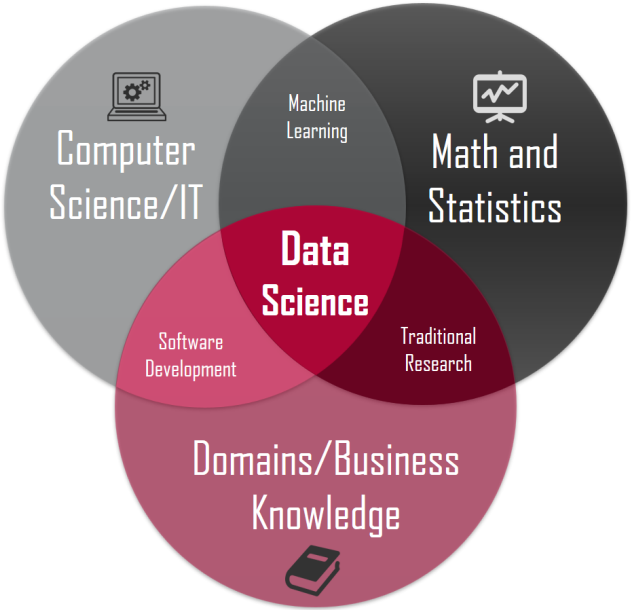
DSAIL
Data Systems and AI Lab

Democratizing Data Science through Interactive Curation of ML Pipelines

Zeyuan Shang, Emanuel Zgraggen, Benedetto Buratti, Ferdinand Kossmann, Philipp Eichmann, Yeounoh Chung, Carsten Binnig, Eli Upfal, Tim Kraska

Data Science is Difficult for Domain Experts

- Data science requires many skills: programming, statistical, machine learning
- Domain experts don't have such backgrounds while programmers or researchers don't have domain knowledge



DARPA Data-Driven Discovery of Model (D3M) AutoML Competition

	Solved Problems	% of Tasks Beat Baseline
Alpine Meadow	100%	80%
System 2	40%	27%
System 3	40%	13%
DARPA Baseline	100%	0%
System 4	20%	7%
System 5	87%	47%
System 6	27%	7%
System 7	60%	20%
System 8	87%	53%
System 9	60%	20%
System 10	60%	20%

- Goal: develop automated model discovery systems for domain experts
- Evaluation on real world datasets, including structured classification and regression task, image classification and measuring, audio transcription, among others
- System 2-10 are competing teams from UC Berkeley, Stanford, NYU, ….

We emulate a Data Scientist

Can you predict the sales next month for me?

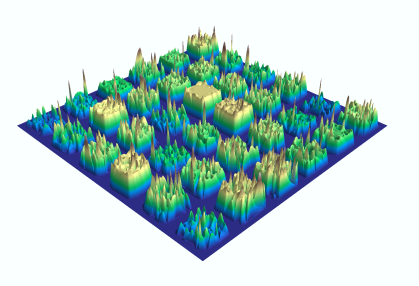
"What modeling options do I have?"

"What should I try first?"

How can I get some quick results?

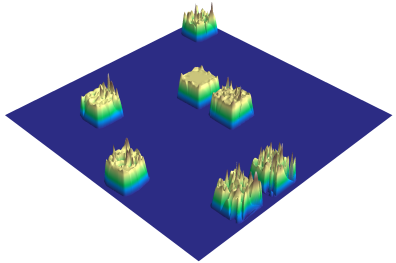
Overview of Alpine Meadow

"What modeling options do I have?"



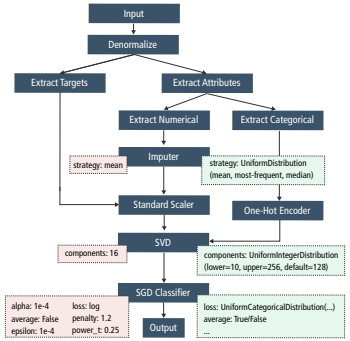
Rule-based Search Space Expansion

"What should I try first?"



Preselection Based On Past Experience (Learned Knowledge Base)

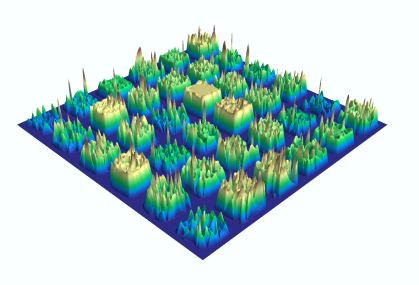
How can I get some quick results?



Adaptive sampling-based pruning

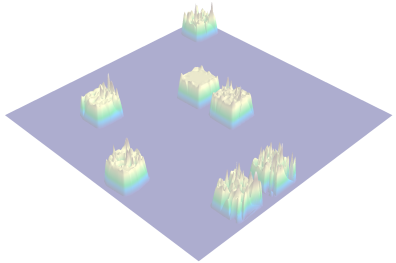
Overview of Alpine Meadow

"What modeling options do I have?"



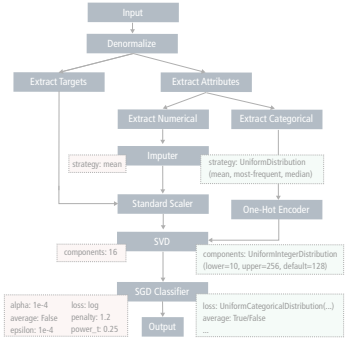
Rule-based Search Space Expansion

"What should I try first?"



Preselection Based On Past Experience (Learned Knowledge Base)

How can I get some quick results?



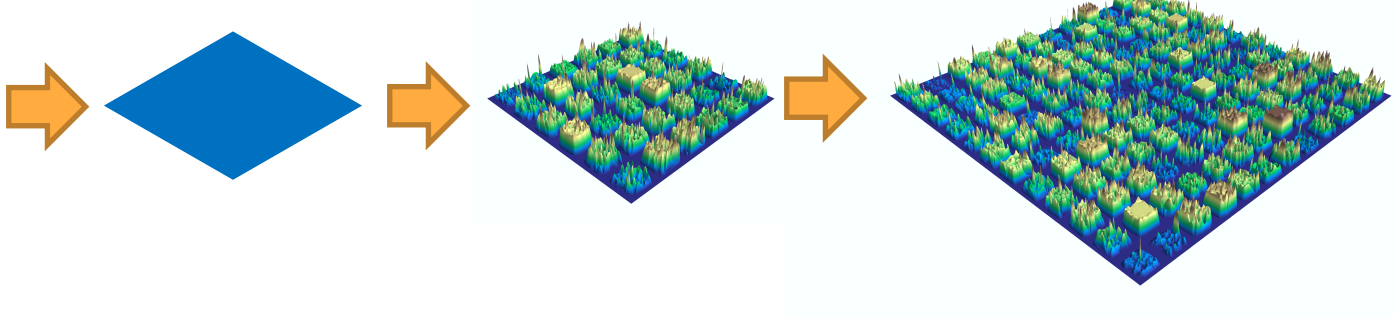
Adaptive sampling-based pruning

Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

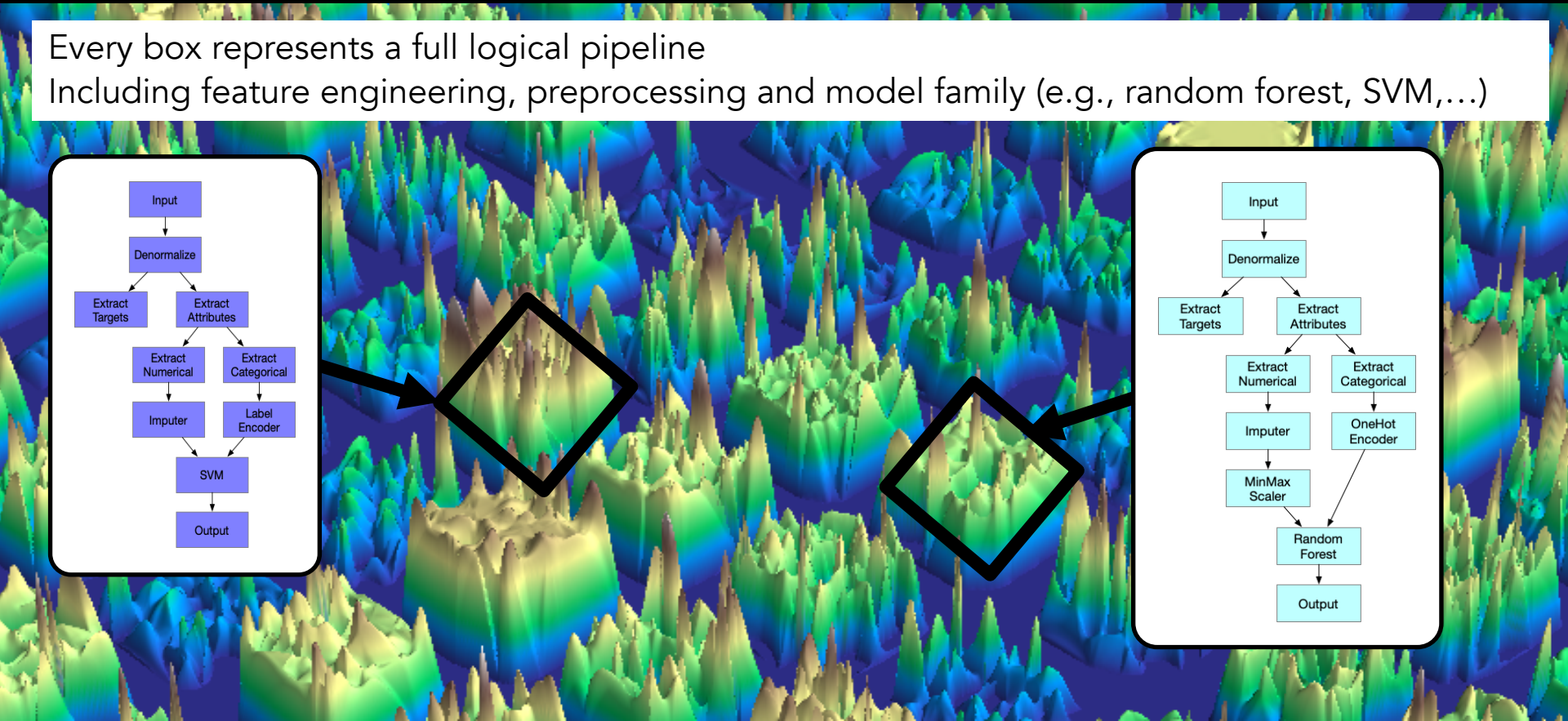
Rule-Based Search Space Expansion

- Rules added by Experts and learned from thousands of publicly available pipelines (Kaggle and OpenML)
- Example rules:
  - unscaled numeric feature → MinMaxScaler, Mean Normalizer
  - categorical feature → use encoder (label or one hot)
  - classification → SVM with default learning rate of 0.001 – 1.00
  - Image classification -> pre-trained neural network (transfer learning)



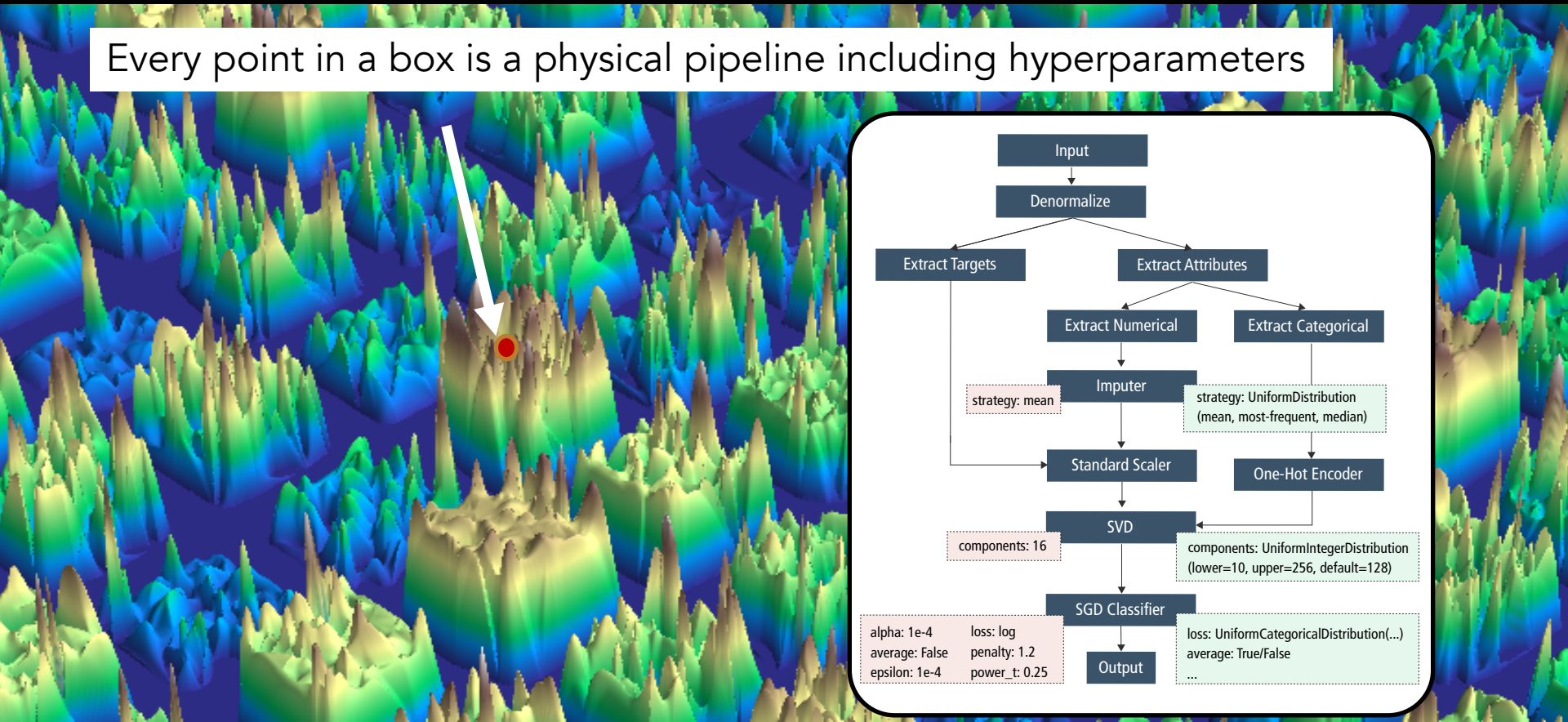
Looking into the Search Space

Every box represents a full logical pipeline
Including feature engineering, preprocessing and model family (e.g., random forest, SVM,…)



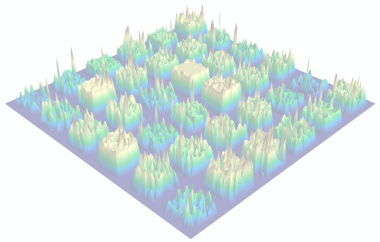
Looking into the Search Space

Every point in a box is a physical pipeline including hyperparameters



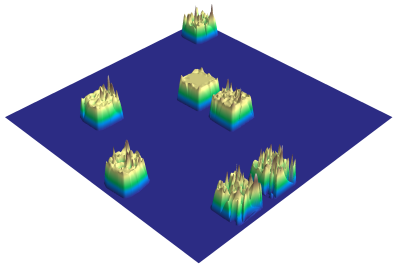
Overview of Alpine Meadow

"What modeling options do I have?"



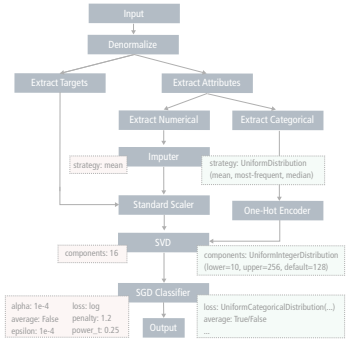
Rule-based Search Space Expansion

"What should I try first?"



Preselection Based On Past Experience (Learned Knowledge Base)

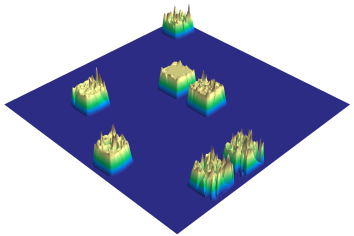
How can I get some quick results?



Adaptive sampling-based pruning

Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results



"What should I try first?"

Preselection Based On Past Experience

- Expected quality/time trade-off (reliable fast pipelines first, high-risk expensive pipelines later)
- Learned from past experience
- Finally, translate pipeline to python code

For example:

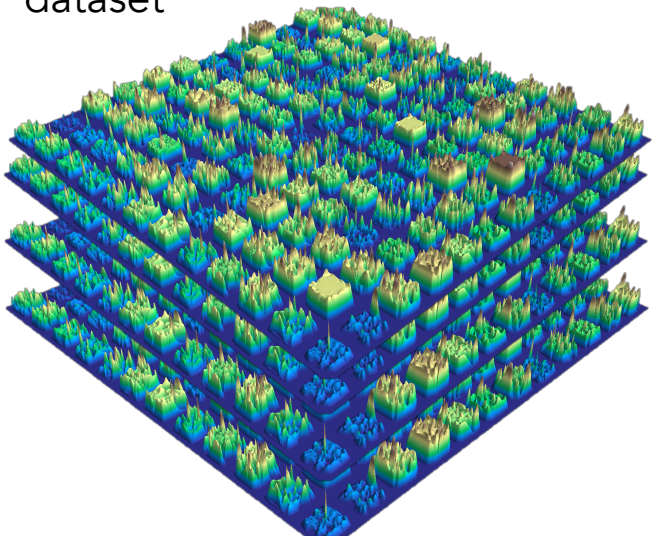
- Gradient Boosting Trees are most-likely a good starting point for the given dataset
- Given the data size, don't even try to use slow models, e.g., SVM and neural nets

Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

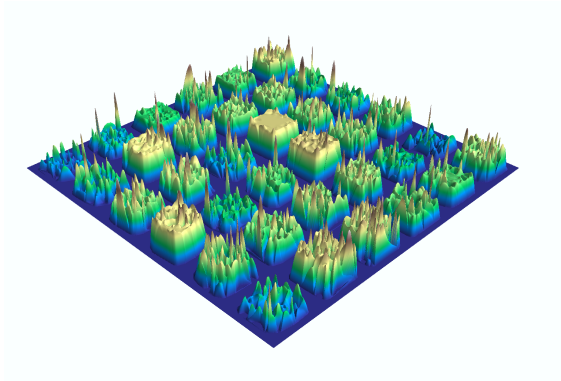
Warm-starting: build the knowledge

- Run Alpine Meadow on lots of datasets and collect all the pipeline traces
- For a new dataset, find some "similar" datasets we have seen before (**meta-learning**) and transfer knowledge from them
- Similarity is based on some trained model to predict similarity based on meta-features of a dataset



Transfer

using the cost model

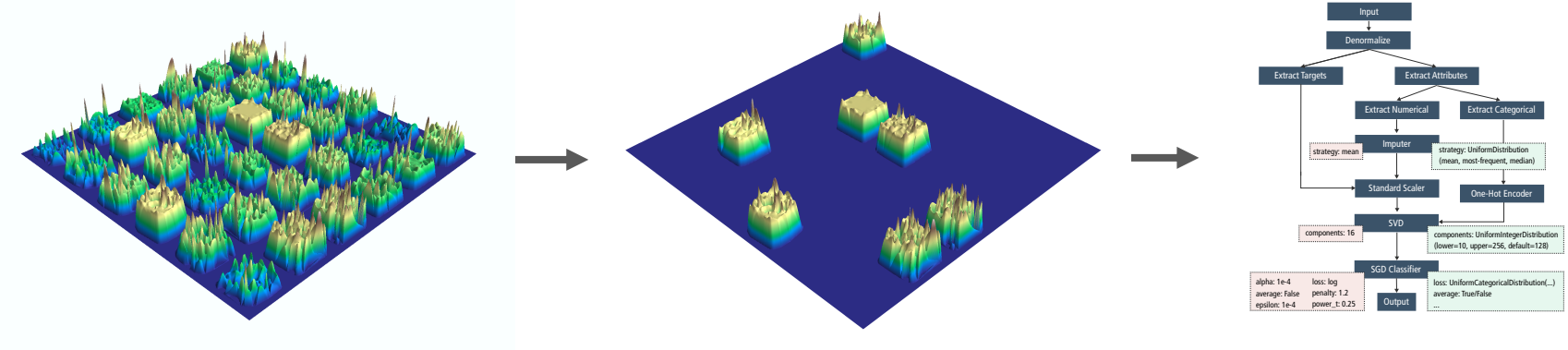


Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

Pipeline Selection

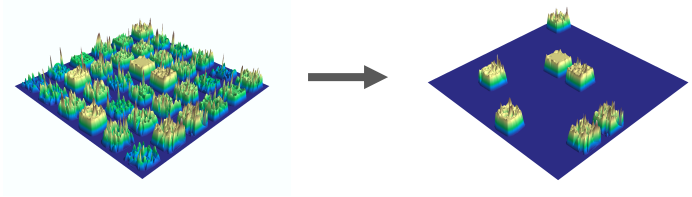
- Search space -> Logical -> Physical
- Combination of multi-armed bandit and Bayesian Optimization, while previous methods only use one of them



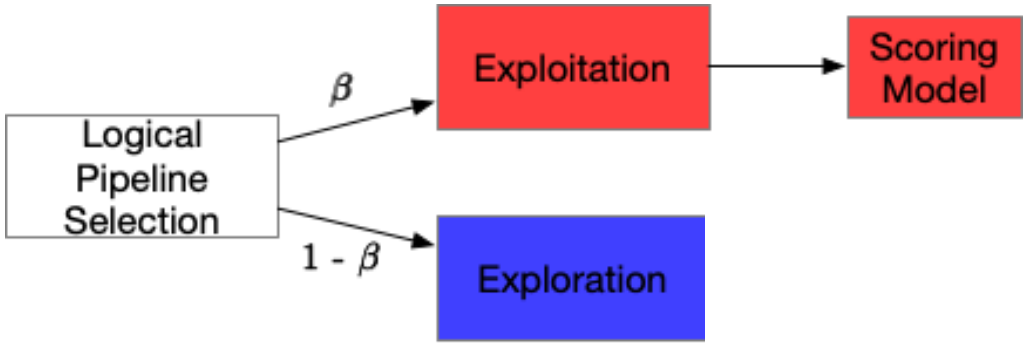
Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

Logical Pipeline Selection



- Balance between exploitation and exploration
- Exploitation: exploiting good boxes
- Exploration: avoid being trapped in a local optimum



Alpine Meadow

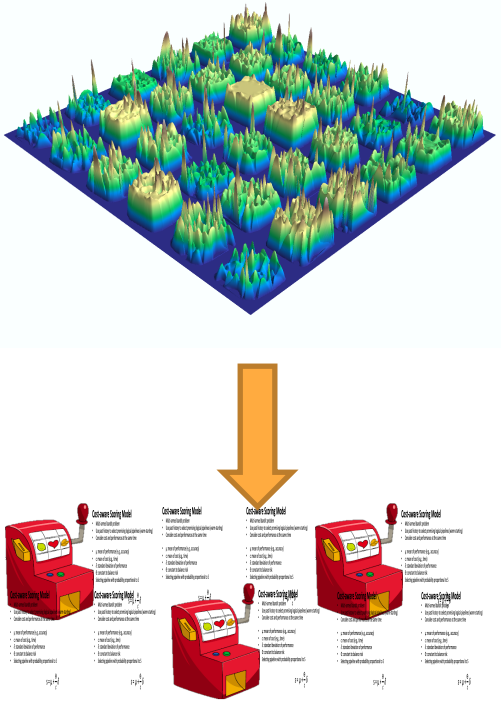
Not Your Normal AutoML–Tool: Built For Interactive Results

Cost-aware Scoring Model

- Multi-armed bandit problem
- Use past history to select promising logical pipelines (warm-starting from the knowledge bases)
- Consider cost and performance at the same time

- μ: mean of performance (e.g., accuracy)
- c: mean of cost (e.g., time)
- δ: standard deviation of performance
- **Θ**: constant to balance risk
- Selecting pipeline with probability proportional to S

$$s = \mu + \frac{\Theta}{c}\delta$$

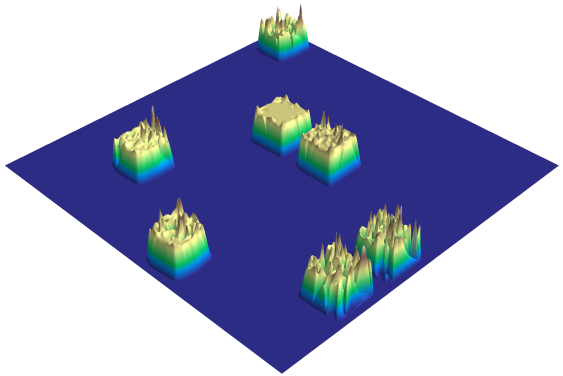


Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

Physical Pipeline Selection

- Hyper-parameter tuning: Bayesian Optimization
- Efficient method for black-box function optimization
- Model the function behavior and select the next promising one

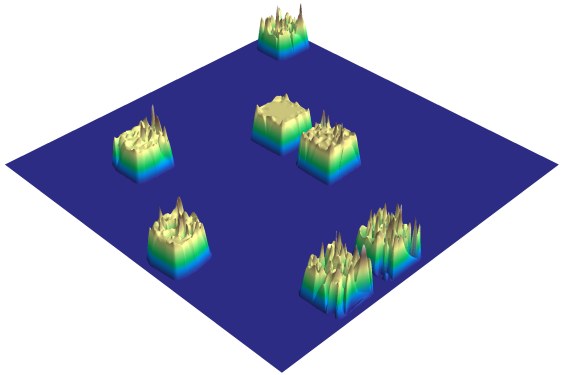


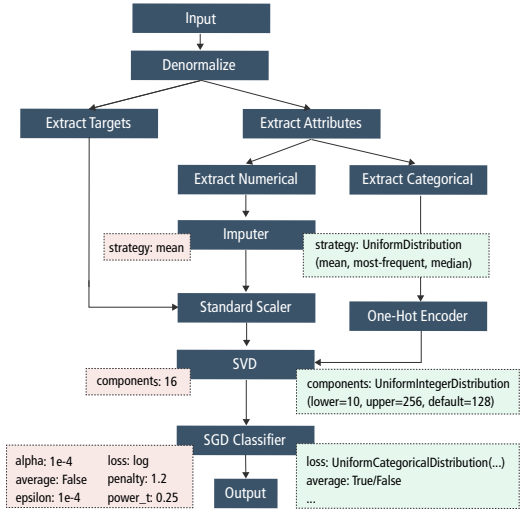
Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

Physical Pipeline Selection

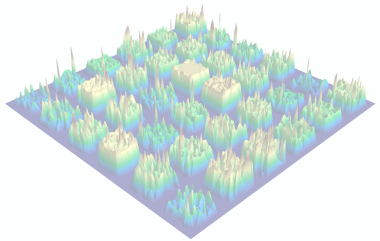
- Hyper-parameter tuning: Bayesian Optimization
- Efficient method for black-box function optimization
- Model the function behavior and select the next promising one





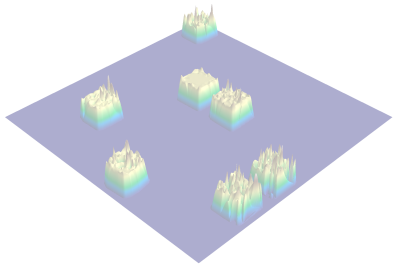
Overview of Alpine Meadow

"What modeling options do I have?"



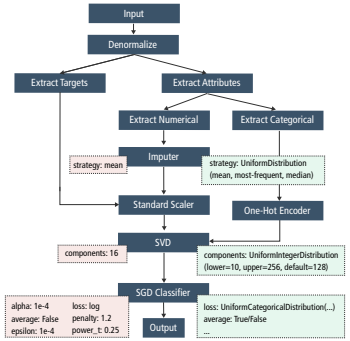
Rule-based Search Space Expansion

"What should I try first?"



Preselection Based On Past Experience (Learned Knowledge Base)

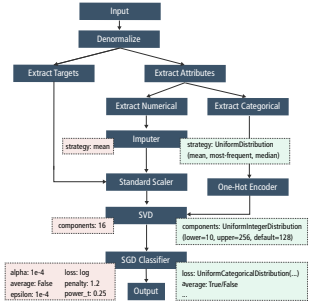
How can I get some quick results?

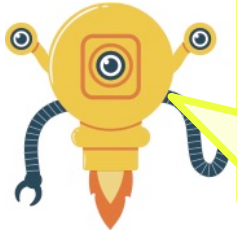


Adaptive sampling-based pruning

Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results





How can I get some quick results?

Try pipeline first on a small sample

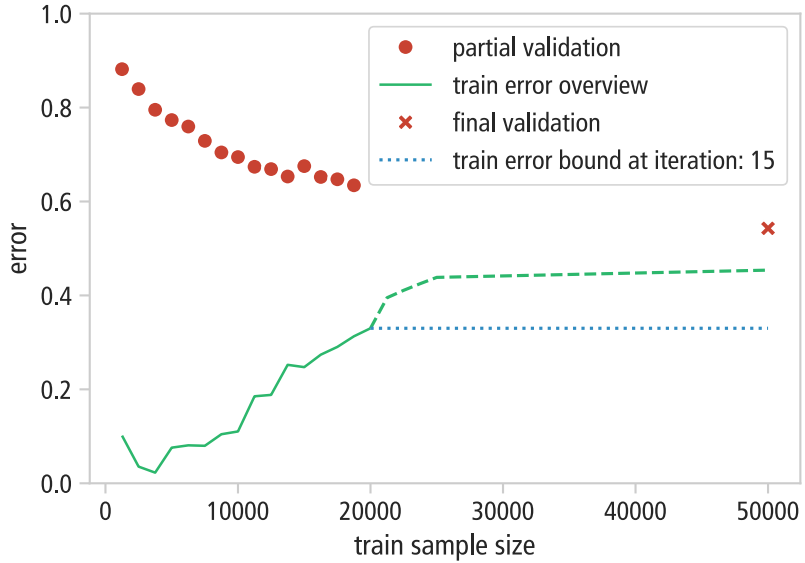
- Observe training and test error
- If pipeline performs well, increase sample size

Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results

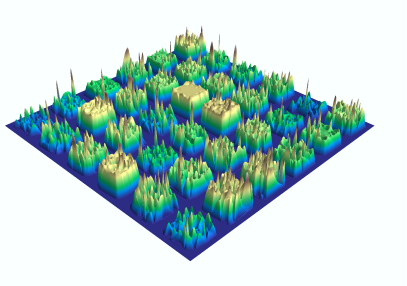
Adaptive Pipeline Selection

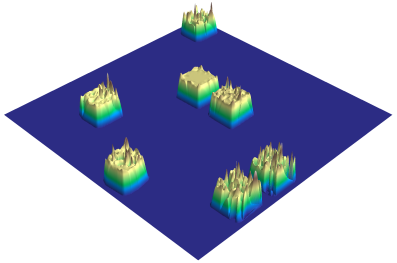
- Train error as the lower bound the test error
- Prune if the train error is beyond the current best validation error

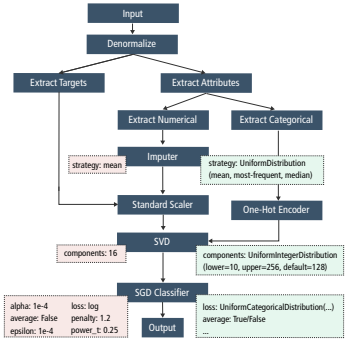


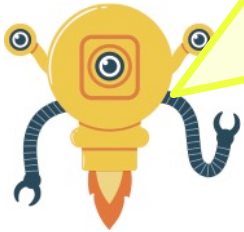
Alpine Meadow

Not Your Normal AutoML–Tool: Built For Interactive Results









Based on the observed performance , let me try again

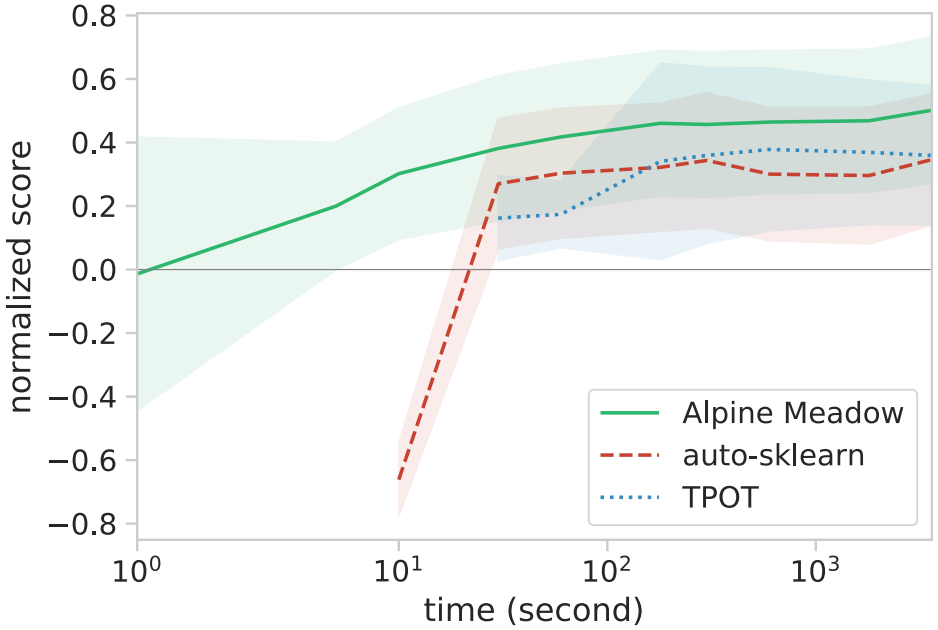
Experiment Setup

- 300 datasets from DARPA (including manual-made baselines from MIT-LL)
  - Classification and regression datasets collected from Kaggle, OpenML, UCI ML Repository
  - 150 for training (learning the knowledge-base) and another 150 for evaluation
- A 40-core machine with various time limits (10s, 60s, .., 10mins, 30mins, 1hr)
- Evaluation Metric: normalized score = $\frac{Score - Baseline\ Score}{Baseline\ Score}$
  - E.g., the baseline score can be accuracy (for classification) or negative mean squared error (for regression)
- Comparison against state-of-the-art AutoML methods and hand-made baselines

Alpine Meadow: Not Only Tabular

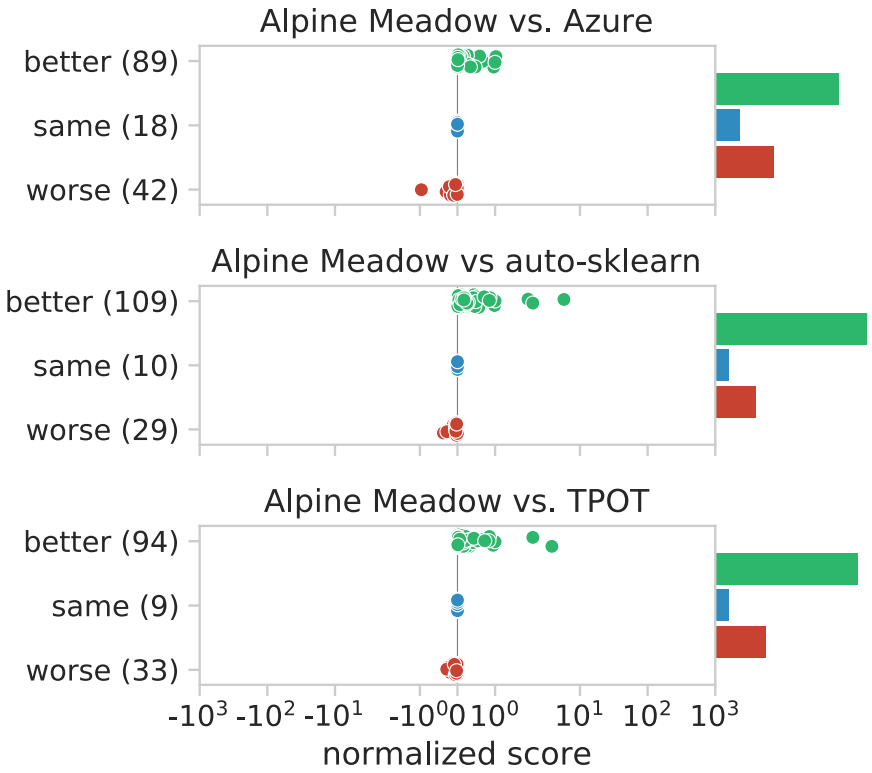
	Azure	auto-sklearn	TPOT	Alpine Meadow
Tabular Classification	99%	99%	87%	100%
Tabular Regression	100%	98%	100%	100%
Graph Matching	Not Supported	Not Supported	Not Supported	100%
Community Detection	Not Supported	Not Supported	Not Supported	100%
Image Classification	Not Supported	Not Supported	Not Supported	100%
Audio Classification	Not Supported	Not Supported	Not Supported	100%
Collaborative Filtering	Not Supported	Not Supported	Not Supported	100%

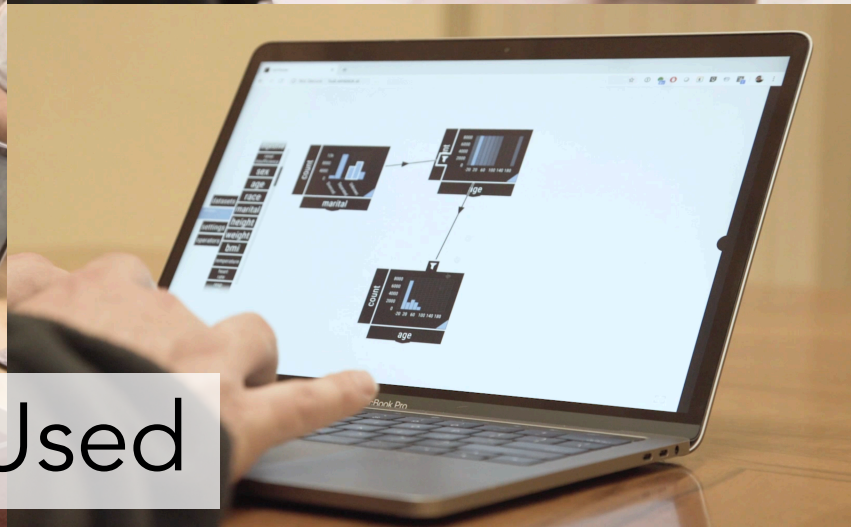
Alpine Meadow: Better Results Faster



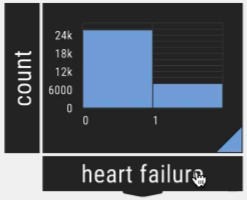
- 150 dateset used for learning the knowledge-base and another 150 for evaluation
- Results averaged over 150 datasets

Alpine Meadow: Better Results Faster





How Alpine Meadow is Used



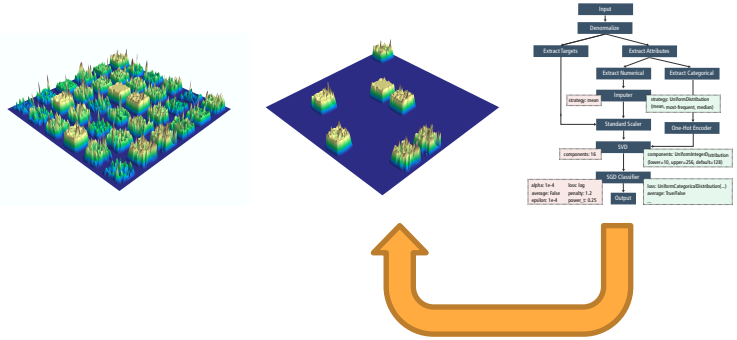
datasets
attributes
settings
operators
feature processing
feature selection
algorithms
raw data
correlation matrix
predictor
frequent itemset
classfication
regression

all
target
mimic
predictor
undefined

Zeyuan Shang
zeyuans@mit.edu

DSAIL
Data Systems and AI Lab

- Democratizing Data Science requires rethinking of the entir analytics stack
- Alpine Meadow: Interactive Virtual Data Scientist
  - Rule-based Search Space
  - Logical and Physical Pipeline Selection
  - Adaptive Pipeline Evaluation
  - Evaluation shows good performance with short latency compared against start-of-the-art



Special Thanks to:

DARPA NSF Google intel Microsoft