Today: AutoML.
→ Logistics
(Tuesday: Whiteboard via IPad)
Thursday: Discussion Sessions (Presenters will lead discussion + Theo)

Today: AutoML

Last class: Automated Feature Selection and how to frame this problem as search over the set of possible features

How to search over the space of possible models

Deep Learning:
→ Vision (CNNs)
→ Audio analytics
→ Text (RNNs, Transformers)
→ Images (CNNs)
→ Tabular data.

Problem

The performance of DL: depends on hyperparameters
→ Units in each layer / # layers
→ Learning rate, crossvalidation parameters

model search

Two types of HP (hyperparameters)
→ architectural HPs
→ Algorithmic HP (optimization) algo

hyperparameter optimization

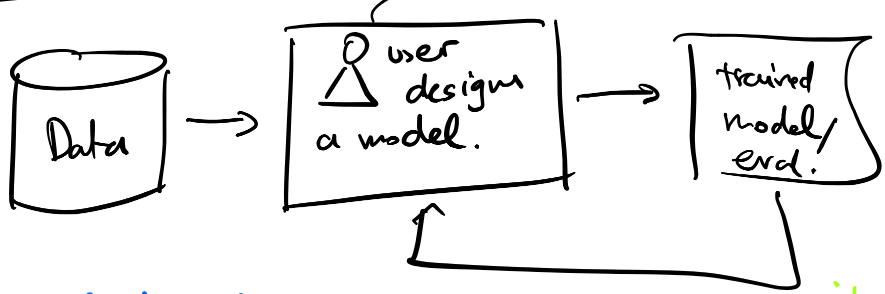
evaluation

↑ costly

Explosion of parameters
50 design decisions

Current solution

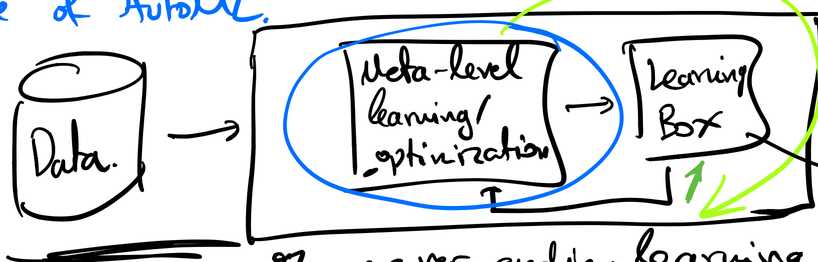
decision on hyper-parameters.



Expensive.

Promise of AutoML.

iterate to maxize performance.



→ not just a trained model (learning pipeline)

?? never ending learning box

define this box

Learning box

- → data cleaning ←
- → " pre-proc ←
- → feature selection ←
- → training algo selection ←
- → model selection ←

etc.

λ

Meta-level optimization

↳ Need to solve a search problem to find the optimal configuration for your learning box

Formal Problem statement

AutoML : it is a hyper-parameter opt. problem.

Grid search → eliminate parts of the grid

λ: hyperparameters. of a ML algorithm A has a domain Λ (valid values that HB λ can take)

Find λ^* s.t. some utility is maximized

Utility function for ML minimize our generalization loss

$\mathcal{L}(A_\lambda, D_{train}, D_{valid}) \xrightarrow{\text{some } \lambda\text{'s}} (D_{train}, D_{validation})$

↳ loss of A, using HPs λ trained on D_{train} and evaluated on D_{valid} (simulating generalization error)

HPO: $\lambda^* = \arg\min_{\lambda \in \Lambda} \mathcal{L}(A_\lambda, D_{train}, D_{valid})$

→ Q: what is this domain Λ? What kind of variable types do we have?

LR: continuous variable
\# of h.units: discrete variable
ReLU or sigmoid: categorical (binary) variable (finite domain)

Λ has mixed data-types

What optimizer to use: ADAM or SGD?

Some of the λs are "unlocked" depending on specific configurations for other parameter λ's

W. Adam → ~~momentum~~

SGD →

Choose the ML model (conditional HPs)

→ Random Forest (# of trees, depth)

→ SVM (kernel)

Instead of a single Algorithm $\underline{\underline{A}} \to \Lambda$

I have access to a $\underline{\underline{\text{set}}}$ of As

$\mathcal{A} = \{A^{(1)}, \ldots, A^{(n)}\}$

$\Lambda^{(i)}$ the HP space of $A^{(i)}$ $\forall\, i = 1, \ldots n$

$\mathcal{L}(A_\lambda^{(i)}, D_{train}, D_{valid})$

$$A^*_{\lambda^*} \in \underset{A^{(i)} \in \mathcal{A},\, \lambda \in \Lambda^{(i)}}{\operatorname{argmin}} \mathcal{L}(A_\lambda^{(i)}, \underline{D_{train}}, \underline{D_{val}})$$

Analyze function $\underline{\underline{\mathcal{L}}}$ (costs)

$\underline{\text{enumerate } A^{(i)}} \rightsquigarrow \underline{\lambda \in \Lambda^{(i)}}$

$\underline{\text{train}}$ over all $\left(|\mathcal{A}| \times \max_i |\Lambda^{(i)}| \right)$

size of search space I have to consider

A is a NN $\rightsquigarrow |\Lambda^{(i)}| = 50$ (2)

→ Search is expensive due to exponential explosion.

→ Evaluation at a single point ?

Operations during Eval: → train $A^{(i)}$ using $\lambda^{(i)}$ on Dtrain

→ evaluate on Dvalid

Depending on the

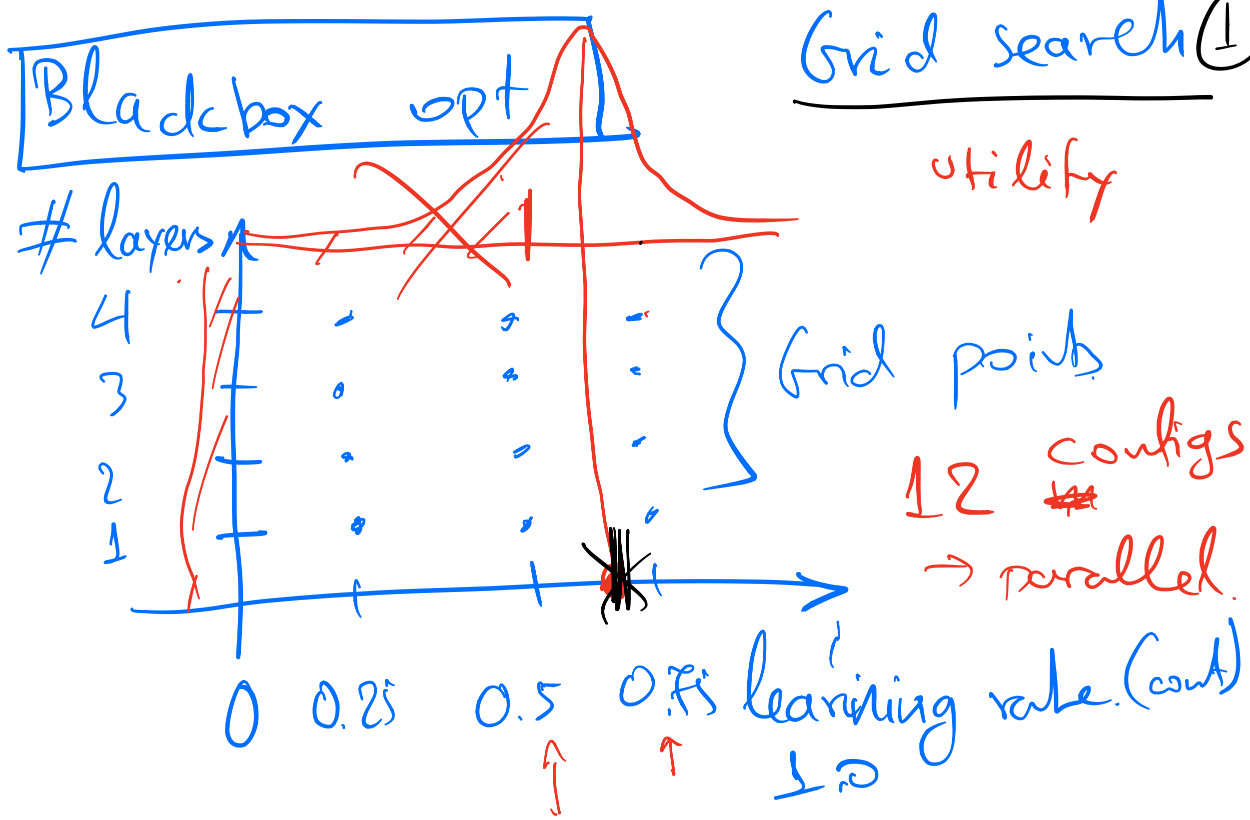
→ model

→ |Dtrain|, |Dvalid| ⇒ expensive op

How to solve this opt problem

$\boxed{\text{Blackbox opt}}$

Grid search (1)

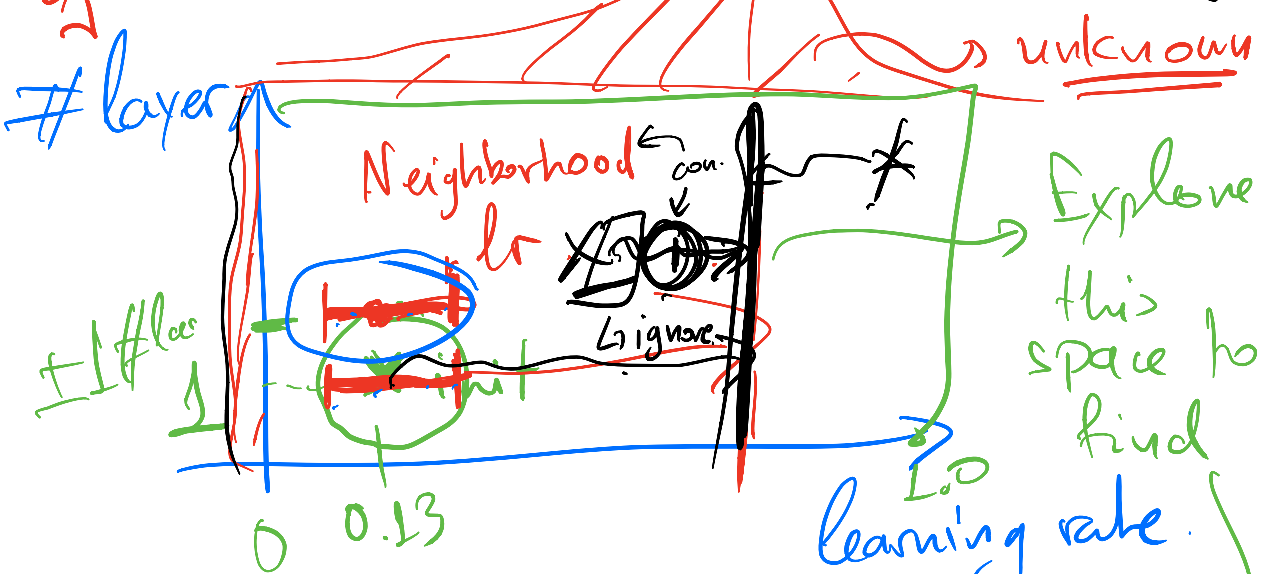


Blackbox opt Case 2 : (previous configs)

Fix $A \rightsquigarrow \Lambda$

$\arg\min_{\lambda} \mathcal{L}(A_\lambda, D_{train}, D_{valid})$

$\boxed{\text{Random Search}}$



↳ a (#layers, lr) config the maximizes my utility.

Random search considers only <u>very recent</u> information

Type 3 of BlackBox opt

Bayesian optimization

High-level

Fit a probabilistic model to your function evaluations

$\underline{f(\lambda)}$ $\langle \lambda, f(\lambda) \rangle$

f: prob. distribution

$\underline{\underline{f:}}$ parameter values $\lambda_1, \lambda_2, \ldots \lambda_D$

$f(\lambda_1), f(\lambda_2), \ldots f(\lambda_D)$

I have a point λ $\underline{f(\lambda) = ?}$

$f(\lambda) \to$ hidden random variable.

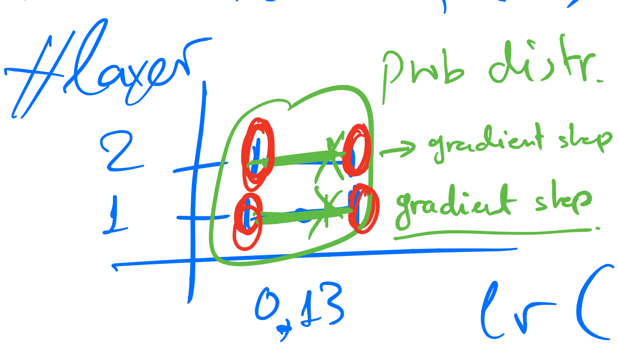
$$\boxed{P(\underset{\sim}{f(\lambda)} \mid f(\lambda_1), f(\lambda_2), \ldots, f(\lambda_D))}$$

find the

use this to $\underline{\text{estimate}}$ utility f on unknown points

given a config param. $\underline{\lambda}$

instead of running my model to find $f(\lambda)$ $\quad \underline{\hat{f}(\lambda)} \approx \underline{P(f(\lambda) \mid f(\lambda_1), f(\lambda_2), \ldots)}$



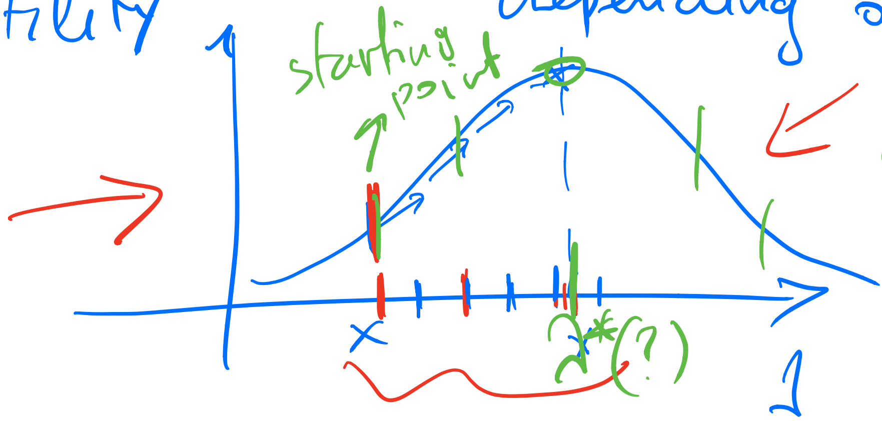
lr (continuous)

(2 optimization problems to find the best performing model.)

find $\hat{\lambda} = \underset{\lambda}{\arg\max}\, P(f(\lambda) \mid f(\lambda_1), f(\lambda_2), f(\lambda_n))$

= cheap evaluation

utility depending on λ



Gaussian distribution

very expensive evaluations

= Bayesian opt: exchange exp. evaluations with cheap inference

→ give me one good point to evaluate on