LAMARR
INSTITUTE FOR MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Resource Aware Machine Learning@Lamarr

Sebastian Buschjäger

CS & Physics Meet-Up by Lamarr & B3D – November, 29th

technische universität dortmund
Fraunhofer IAIS
UNIVERSITÄT BONN
Fraunhofer IML

Resource consumption of computing hardware

Question How many resources are required by new computing hardware in general?

Resource consumption of computing hardware

Question How many resources are required by new computing hardware in general?

Idea Report carbon footprint as a (weak) proxy for general resource consumption

Resource consumption of computing hardware

Question How many resources are required by new computing hardware in general?

Idea Report carbon footprint as a (weak) proxy for general resource consumption

Apple's Product Environmental Report[https:/www.apple.com/environment/]

(excluding end-of-life processing here)

iPhone-14	1 Year [kg]	3 Years [kg]	10 Years [kg]
Production	48.19	48.19	48.19
Transport	1.22	1.22	1.22
Useage	3.66	10.98	36.6

Resource consumption of computing hardware

Question How many resources are required by new computing hardware in general?

Idea Report carbon footprint as a (weak) proxy for general resource consumption

Apple's Product Environmental Report[https:/www.apple.com/environment/]

(excluding end-of-life processing here)

iPhone-14	1 Year [%]	3 Years [%]	10 Years [%]
Production	90.8	79.0	56.0
Transport	2.3	1.9	1.4
Useage	6.9	18.0	42.5

(Percentages may not total 100 due to rounding.)

Resource consumption of computing hardware

Question How many resources are required by new computing hardware in general?

Idea Report carbon footprint as a (weak) proxy for general resource consumption

Apple's Product Environmental Report[https:/www.apple.com/environment/]

(excluding end-of-life processing here)

iPhone-14	1 Year [%]	3 Years [%]	10 Years [%]
Production	90.8	79.0	56.0
Transport	2.3	1.9	1.4
Useage	6.9	18.0	42.5

(Percentages may not total 100 due to rounding.)

Clear Use an iPhone-14 for around 13 years to break even with production costs!

But Average life-cycle for an iPhone-14 are 3 to 4 years

A Closer Look at Older / Smaller Hardware

Common Microcontroller Units[Branco et al. 2019]

MCU	CPU	Flash	(S)RAM
Arduino Uno (ATMega128P)	16MHz	32KB	2KB
Arduino Mega (ATMega2560)	16MHz	256KB	8KB
STM32L0 (Cortex-M0)	32MHz	192KB	20KB
Arduino MKR1000 (Cortex-M0)	48MHz	256KB	32KB
STM32F2 (Cortex-M3)	120MHz	1MB	128KB
STM32F4 (Cortex-M4)	180MHz	2MB	384KB
RPi A+	700MHz	SD Card	256MB
RPi Zero	1GHz	SD Card	512MB
RPi 3B	4@1.2GHz	SD Card	1GB
Apple A7 (iPhone 5)	2@1.4 Ghz	16-64 GB	1GB

Empirical Risk Minimization Revisited

$$f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{(x,y) \in \mathcal{D}} \ell(f(x), y) + \lambda R(f)$$

Empirical Risk Minimization Revisited

$$f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{(x,y) \in \mathcal{D}} \ell(f(x), y) + \lambda R(f)$$

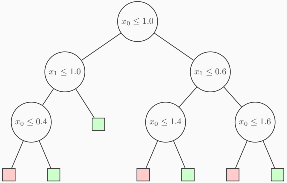
Use resource-friendly model class $\mathcal{F}$ directly
$\Rightarrow$ Guaranteed resource consumption, but maybe weak loss

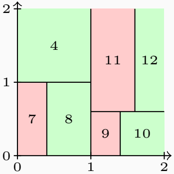
Guide selection via regularization
$\Rightarrow$ Direct trade-off between loss and model complexity via λ

Additive Tree Ensembles

In many applications Random Forests are outperforming Deep Learning methods

Axis-aligned Decision Trees Split data into groups of increasing label purity

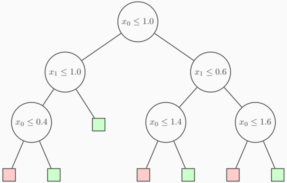


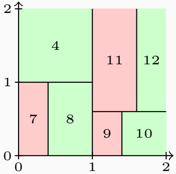


Additive Tree Ensembles

In many applications Random Forests are outperforming Deep Learning methods

Axis-aligned Decision Trees Split data into groups of increasing label purity



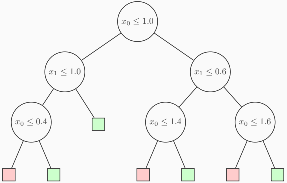


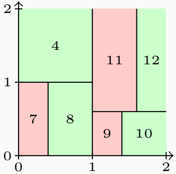
$$h(x) = \sum_{i=1}^{L} y_i \pi_i(x),\ \pi_i(x) = 1 \text{ if x in leaf i else 0}$$

Additive Tree Ensembles

In many applications Random Forests are outperforming Deep Learning methods

Axis-aligned Decision Trees Split data into groups of increasing label purity





$$h(x) = \sum_{i=1}^{L} y_i \pi_i(x),\ \pi_i(x) = 1 \text{ if x in leaf i else 0}$$

Random Forest Train multiple DTs on bootstrap samples and average predictions

$$f(x) = \frac{1}{M} \sum_{i=1}^{M} h_i(x)$$

Training Additive Ensembles for Small Devices

Wait DTs are simple! RFs is a set of trees. Hence, aren't RF already resource-aware?!

Training Additive Ensembles for Small Devices

Wait DTs are simple! RFs is a set of trees. Hence, aren't RF already resource-aware?!

Unfortunately RFs can easily grow in size, even for smaller datasets.

	adult	avila	bank	eeg	elec	mnist
accuracy [%]	86.78	98.58	90.39	93.42	88.98	96.53
model size [MB]	24.99	32.85	24.99	14.95	24.99	56.99

Training Additive Ensembles for Small Devices

Wait DTs are simple! RFs is a set of trees. Hence, aren't RF already resource-aware?!

Unfortunately RFs can easily grow in size, even for smaller datasets.

	adult	avila	bank	eeg	elec	mnist
accuracy [%]	86.78	98.58	90.39	93.42	88.98	96.53
model size [MB]	24.99	32.85	24.99	14.95	24.99	56.99

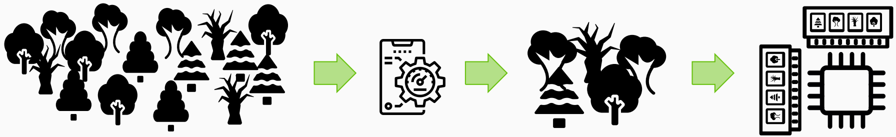
Can we compute a small *and* accurate tree ensemble?

Ensemble Pruning Revisited

Idea 1 Given a large forest with M trees select only a few trees

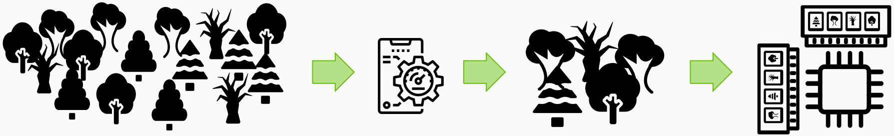
Ensemble Pruning Revisited

Idea 1 Given a large forest with M trees select only a few trees



Ensemble Pruning Revisited

Idea 1 Given a large forest with M trees select only a few trees



Formally

$$f_w(x) = \frac{1}{K}\sum_{i=1}^{M} w_i h_i(x)$$

solve

$$\arg\min_{w \in \{0,1\}^M} \sum_{(x,y)\in\mathcal{S}} \ell\left(f_w(x), y\right) \text{ s.t. } \|w\|_0 = K \ll M$$

Ensemble Pruning Revisited (2)

Ensemble Pruning Standard method to select fewer trees in a forest

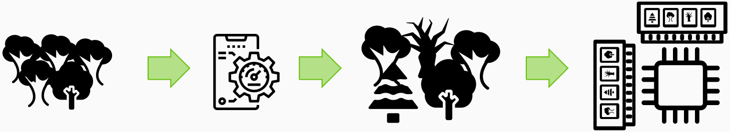
- **Ranking**[Martínez-Muñoz and Suárez 2004, Li et al. 2012, Margineantu and Diettereich 1997]
  Assign a score to each tree and select the top-k trees
- **Clustering**[Giacinto et al. 2000, Bakker and Heskes 2003, Lazarevic and Obradovic 2001, ...]
  Cluster trees and then select a representative from each cluster
- **MQIP**[Cavalcanti et al. 2016, Zhang et al. 2006]
  Construct Mixed Quadratic Integer Program to select trees
- **Ordering**[Jiang et al. 2017, Lu et al. 2010, Margineantu and Dietterich 1997, ...]
  Order the trees according to their overall contribution and select the first K trees

Leaf-Refinement

Idea 2 Use a small forest from the beginning and refine it[Ren et al. 2015, Buschjäger and Morik 2021]

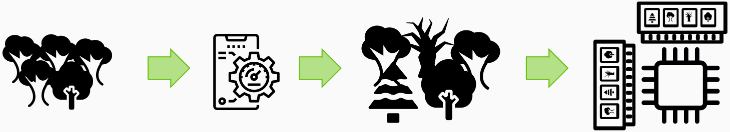
Leaf-Refinement

Idea 2 Use a small forest from the beginning and refine it[Ren et al. 2015, Buschjäger and Morik 2021]



Leaf-Refinement

Idea 2 Use a small forest from the beginning and refine it[Ren et al. 2015, Buschjäger and Morik 2021]



Formally Perform SGD on the leaf nodes $\theta_i = (y_{i,1}, \dots, y_{i,L_i}), \quad \theta = [\theta_1, \dots, \theta_M]$

$$\arg\min_{\theta \in \mathbb{R}^{M \cdot L_1 \dots L_M}} \sum_{(x,y) \in \mathcal{S}} \ell\left(f_\theta(x), y\right)$$

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

$$\underset{\substack{w \in [0,1]^M \\ \theta \in \mathbb{R}^{M \cdot L_1 \dots L_M}}}{\arg\min} \sum_{(x,y) \in \mathcal{S}} \ell\left(f_{w,\theta}(x), y\right) + \lambda \|w\|_1$$

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

$$\underset{\boxed{\substack{w\in[0,1]^M \\ \theta\in\mathbb{R}^{M\cdot L_1 \ldots L_M}}}}{\arg\min} \sum_{(x,y)\in\mathcal{S}} \ell\left(f_{w,\theta}(x), y\right) + \lambda\|w\|_1$$

Relaxed Constraints

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

$$\underset{\boxed{\substack{w\in[0,1]^M \\ \theta\in\mathbb{R}^{M\cdot L_1 \ldots L_M}}}}{\arg\min} \sum_{(x,y)\in\mathcal{S}} \ell\left(f_{\boxed{w,\theta}}(x), y\right) + \lambda\|w\|_1$$

Relaxed Constraints

Optimization over both parameters

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

$$\underset{\boxed{\substack{w\in[0,1]^M \\ \theta\in\mathbb{R}^{M\cdot L_1 \dots L_M}}}}{\arg\min} \sum_{(x,y)\in\mathcal{S}} \ell\left(f_{\boxed{w,\theta}}(x), y\right) + \boxed{\lambda\|w\|_1}$$

Relaxed Constraints

Regularization to enforce pruning

Optimization over both parameters

Leaf-Refinement and Pruning combined

Why not combine both approaches?[Buschjäger and Morik 2023]

$$\underset{\boxed{\substack{w\in[0,1]^M \\ \theta\in\mathbb{R}^{M\cdot L_1\ldots L_M}}}}{\arg\min} \sum_{(x,y)\in\mathcal{S}} \ell\left(f_{\boxed{w,\theta}}(x), y\right) + \boxed{\lambda\|w\|_1}$$

Relaxed Constraints

Optimization over both parameters

Regularization to enforce pruning

Challenge Constraint optimization $\rightarrow$ use Proximal Gradient Descent

Experiment 1: Compare with Vanilla Random Forest

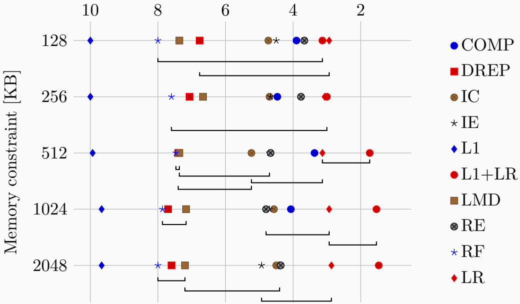
		adult	avila	bank	eeg	elec	mnist
RF	accuracy [%]	86.78	98.58	90.39	93.42	88.98	96.53
	model size [MB]	24.99	32.85	24.99	14.95	24.99	56.99
LR+L1	accuracy [%]	87.25	99.78	90.5	95.55	92.49	98.05
	model size [MB]	0.06	3.52	0.07	5.88	14.37	28.49

Experiment 2: Perform systematic experiments on more datasets

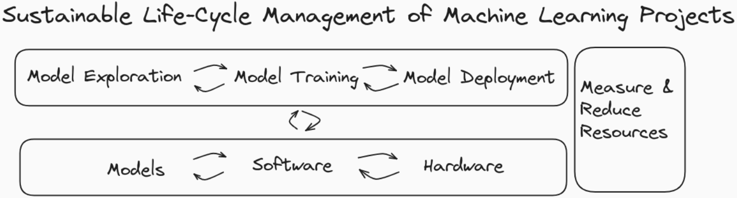
Comparison with more algorithms on more datasets

15 datasets, 10 methods, 920 hyperparameter configs per datasets

13 800 models cross-validated



Towards Sustainable Life Cycle Management of ML Projects



Possible Research Questions

- How can we measure the {energy, performance, embodied carbon} of ML systems?
- What {abstraction,language} is required to {reason about, optimize} ML systems?
- How can we reduce {bandwidth, voltage, model size, runtime}?
- Can we re-use old/existing hardware for new models?
- Is {anytime, online, preiodical} training more efficient than batch processing?
- How do you manage a fleet of ML systems?

Summary

Use the iPhone-14 for $\approx$ 10 years to make it worth building it

Ensmeble Pruning

- Ensemble Pruning removes unncessary trees; Leaf-Refinement improves trees
- Leaf-Refinement + Pruning leads to smaller *and* better models

Sustainable Life Cycle Management of ML Projects

- Explore ML Projects while looking at resource constraints
- Explore the entire pipeline of ML projects from training to deployment