Today

- Resource-constrained ML
  Motivation
- Decision trees
- Bonsai
  - Design
  - Training
  - Results.

Resource Constrained ML

Q. Why do we want to run on-device training?

testing

privacy

lack comms

latency

comms also consume power.

Arduino Uno

2 kB SRAM

16 MHz

32 kB flash

1000

BBC Micro:bit

16 kB SRAM

16 MHz processor

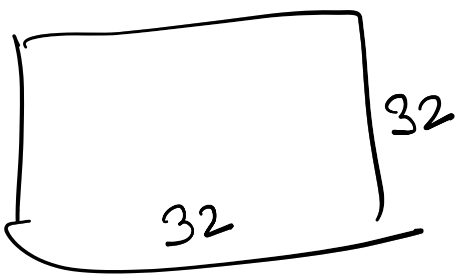
256 kB flash

AlexNet $10^7 - 10^8$ parameters

4 bits

400 Mb of data

→ 50 MB



$32 \times 32 \times 3 \times 4$

$2^5 \times 2^5 \times 2^2 \times 3$

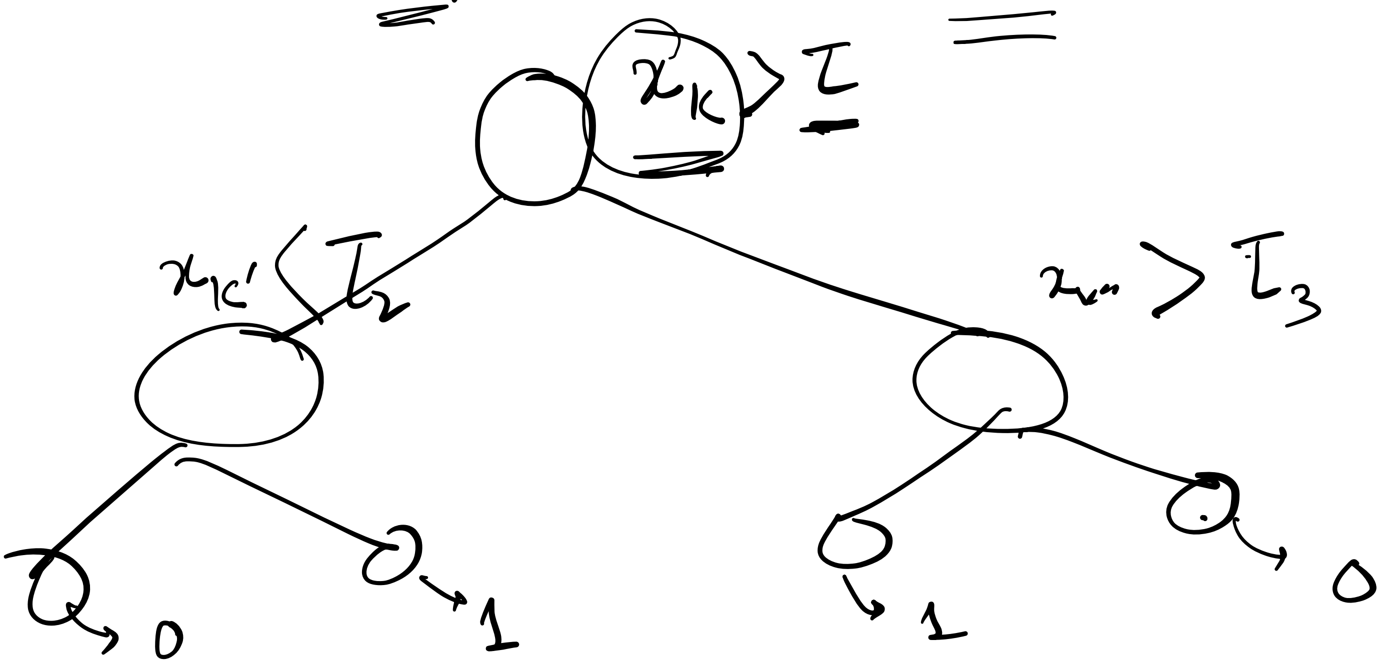
12×10^3

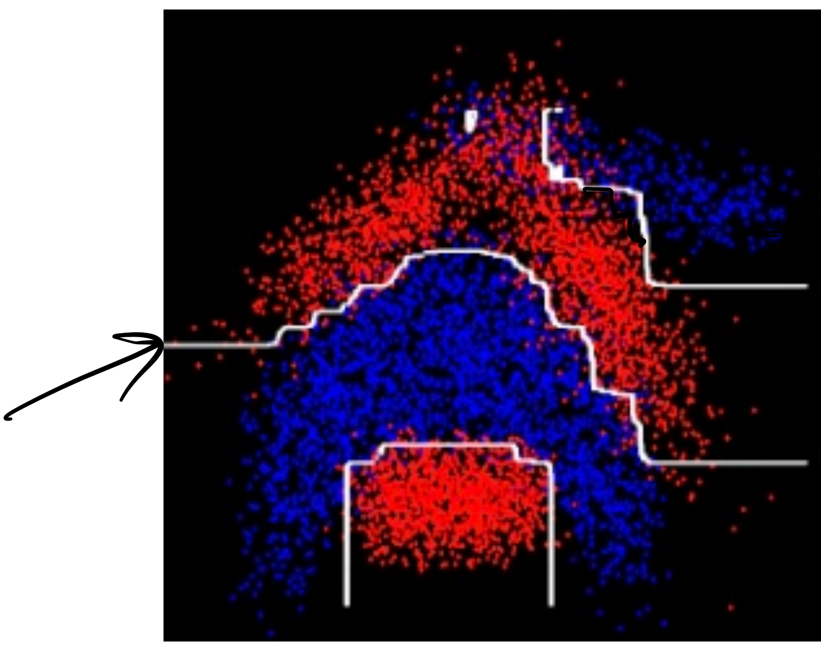
12 kb

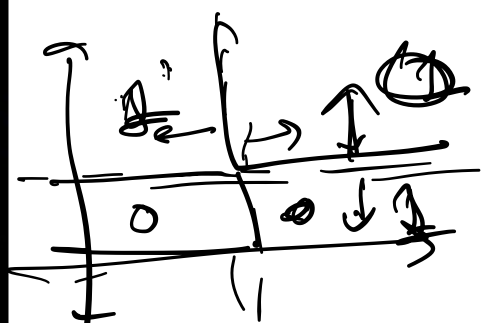
https://www.youtube.com/watch?v=J-VVh5ezqGA

Decision Trees

$[\; x_0, \; \ldots \; x \; \ldots \; x_N \;]$



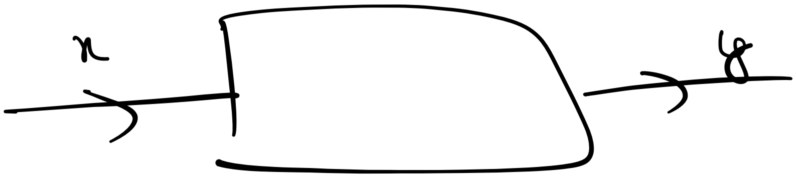




Bonsai: Key Ideas

$$\mathbf{y}(\mathbf{x}) = \sum_k I_k(\overset{z}{\mathbf{x}})\mathbf{W}_k^\top \mathbf{Z}\mathbf{x} \circ \tanh(\sigma \mathbf{V}_k^\top \mathbf{Z}\mathbf{x})$$

projection matrix.



don't have enough memory $\Rightarrow$ project x (d) $\Rightarrow$ Zx $d' << d$

streaming fashion

$$\sum_k I_k$$

$k \in \text{path} \Rightarrow I_k = 1$

$k \notin \text{path} \Rightarrow I_k = 0$

Imp 7 functions → 3 functions per input

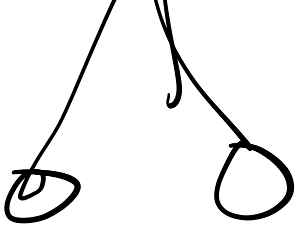
12 functions → 3 functions per input

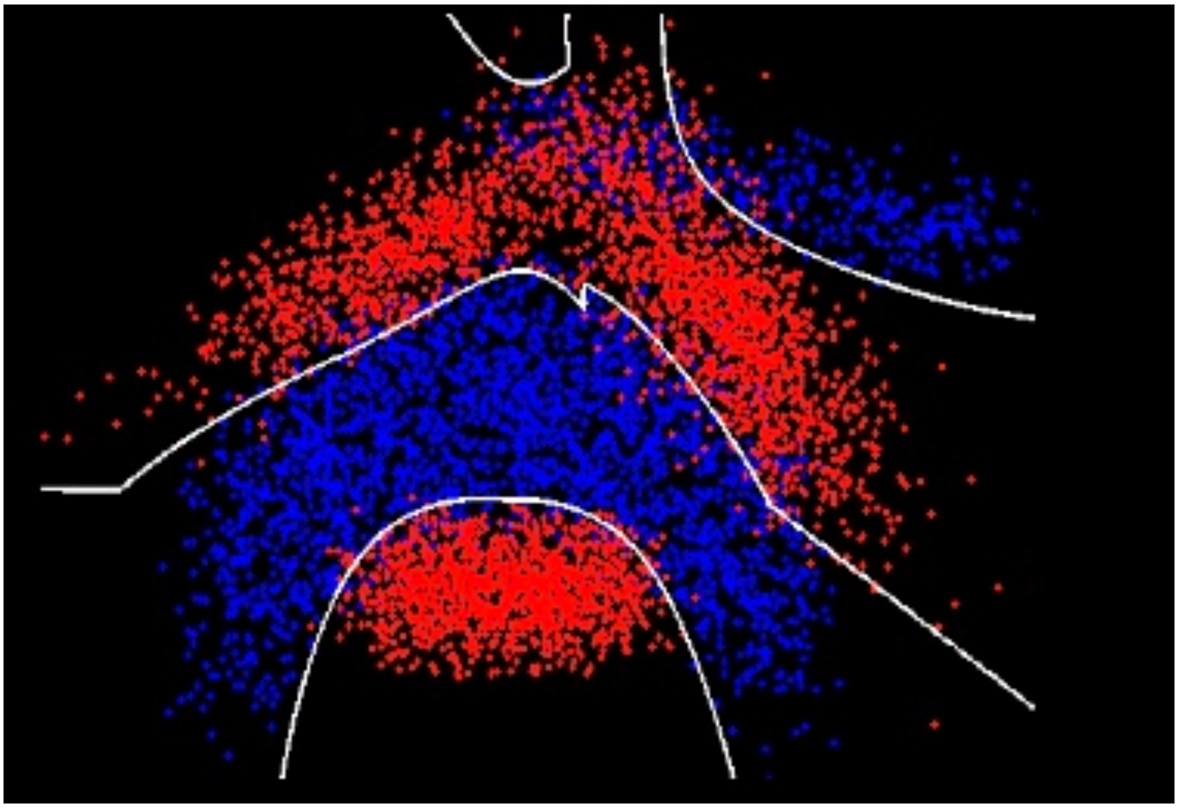
tanh → non-linearity in deep nets.

$$I_k[x] = \frac{1}{2} I_j(x) \left(1 + (-1)^{k-2j} \tanh\left(\sigma_I \, \theta_j^T z_x\right)\right)$$

$\theta^T z_x > \alpha$

$x_i > \alpha) I$





Sparse Streaming

Loss Functions

$$
\begin{aligned}
\min_{\mathbf{Z},\boldsymbol{\Theta}} \quad \mathcal{J}(\mathbf{Z},\boldsymbol{\Theta}) &= \frac{\lambda_{\boldsymbol{\theta}}}{2}\mathrm{Tr}(\boldsymbol{\theta}^\top\boldsymbol{\theta}) + \frac{\lambda_{\mathbf{W}}}{2}\mathrm{Tr}(\mathbf{W}^\top\mathbf{W}) \\
&+ \frac{\lambda_{\mathbf{V}}}{2}\mathrm{Tr}(\mathbf{V}^\top\mathbf{V}) + \frac{\lambda_{\mathbf{Z}}}{2}\mathrm{Tr}(\mathbf{Z}\mathbf{Z}^\top) \\
&+ \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathbf{x}_i, \mathbf{y}_i, \mathbf{y}(\mathbf{x}_i); \mathbf{Z}, \boldsymbol{\Theta}) \\
\text{s. t.} \quad \|\mathbf{Z}\|_0 \le B_{\mathbf{Z}}, \ & \|\boldsymbol{\Theta}\|_0 \le B_{\boldsymbol{\Theta}}
\end{aligned}
$$

$\Theta \in \{\theta, W, V\}$

pruning → zero out the smallest values.

optimization ⟷ pruning

Training Process

Gradient-based update step

$$
\left.\begin{aligned}
\mathbf{Z}^{t+1} &= \mathbf{Z}^{t} - \eta_{\mathbf{Z}}^{t} \nabla_{\mathbf{Z}} \mathcal{J}(\mathbf{Z}^{t}, \boldsymbol{\Theta}^{t})|_{\mathrm{supp}(\mathbf{Z}^{t})} \\
\boldsymbol{\Theta}^{t+1} &= \boldsymbol{\Theta}^{t} - \eta_{\boldsymbol{\Theta}}^{t} \nabla_{\boldsymbol{\Theta}} \mathcal{J}(\mathbf{Z}^{t}, \boldsymbol{\Theta}^{t})|_{\mathrm{supp}(\boldsymbol{\Theta}^{t})}
\end{aligned}\right\} \text{M steps}
$$

$\downarrow$

Update gradient everywhere once

$\downarrow$ pruning

$$
\mathbf{Z}^{t+M+1} = \mathbf{T}_{B_{\mathbf{Z}}}(\mathbf{Z}^{t+M} - \eta_{\mathbf{Z}}^{t+M} \nabla_{\mathbf{Z}} \mathcal{J}(\mathbf{Z}^{t+M}, \boldsymbol{\Theta}^{t+M}))
$$

$$
\boldsymbol{\Theta}^{t+M+1} = \mathbf{T}_{B_{\boldsymbol{\Theta}}}(\boldsymbol{\Theta}^{t+M} - \eta_{\boldsymbol{\Theta}}^{t+M} \nabla_{\boldsymbol{\Theta}} \mathcal{J}(\mathbf{Z}^{t+M}, \boldsymbol{\Theta}^{t+M}))
$$

B_Z → 10KB

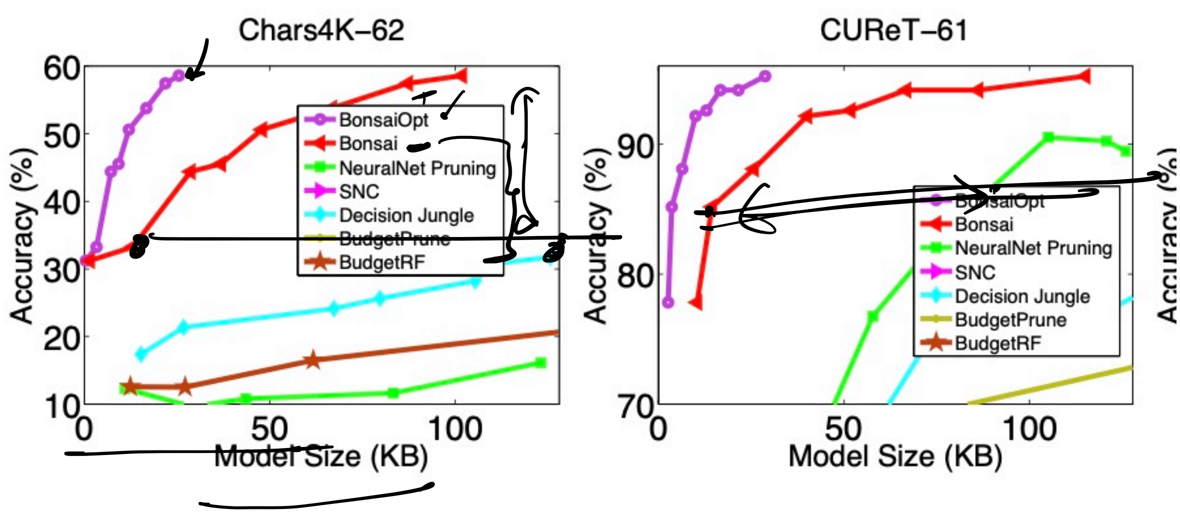
B (H)

10000 , 1B

2000 , 4bits each.

Experiments

TinyML



Dataset		BonsaiOpt	Bonsai	Linear	LDKL	NeuralNet Pruning	Cloud GBDT
Eye-2	Model Size (KB)	0.30	1.20	2.00	1.88	1.96	586.00
	Accuracy (%)	88.78	88.26	80.10	66.33	80.45	84.18
	Prediction Time (ms)	10.75	12.26	15.13	15.80	15.48	2186.59
	Prediction Energy (mJ)	2.64	3.01	3.72	3.89	3.81	1311.95
RTWhale-2	Model Size (KB)	0.33	1.32	0.86	1.00	1.17	156.00
	Accuracy (%)	60.94	61.74	50.76	50.24	52.44	59.40
	Prediction Time (ms)	5.24	7.11	4.68	6.16	8.86	521.27
	Prediction Energy (mJ)	1.29	1.75	1.15	1.52	2.18	312.76
Chars4K-2	Model Size (KB)	0.50	2.00	1.56	1.95	1.96	125.00
	Accuracy (%)	74.71	74.28	51.06	67.29	63.90	73.49
	Prediction Time (ms)	4.21	8.55	7.39	8.61	14.09	160.40
	Prediction Energy (mJ)	1.03	2.10	1.81	2.13	3.48	63.52
WARD-2	Model Size (KB)	0.47	1.86	1.99	1.99	1.91	93.75
	Accuracy (%)	95.70	95.86	87.57	89.64	91.76	98.05
	Prediction Time (ms)	4.85	8.13	7.48	9.99	14.22	293.13
	Prediction Energy (mJ)	1.18	1.99	1.84	2.47	3.49	116.08
CIFAR10-2	Model Size (KB)	0.50	1.98	1.56	1.88	1.96	625.00
	Accuracy (%)	73.05	73.02	69.11	67.54	67.01	76.68
	Prediction Time (ms)	4.55	8.16	7.73	8.12	13.87	160.40
	Prediction Energy (mJ)	1.11	2.01	1.90	2.00	3.43	63.52
USPS-2	Model Size (KB)	0.50	2.00	1.02	1.87	2.00	468.75
	Accuracy (%)	94.42	94.42	83.11	91.96	88.68	96.11
	Prediction Time (ms)	2.93	5.57	4.15	5.59	9.51	83.45
	Prediction Energy (mJ)	0.71	1.37	1.02	1.37	2.33	33.05
MNIST-2	Model Size (KB)	0.49	1.96	1.93	1.87	1.90	93.75
	Accuracy (%)	94.28	94.38	86.16	87.01	88.65	98.24
	Prediction Time (ms)	5.17	8.90	6.72	8.72	14.67	264.96
	Prediction Energy (mJ)	1.27	2.18	1.65	2.16	3.59	104.92

400

1000 x 20

400

256

784 x 20

9.520

MIT