ECAI 50th ANNIVERSARY
19th-24th Oct' 2024
Santiago, Spain

Vanilla Gradient Descent for Oblique Decision Trees

Subrat Prasad Panda
NTU Singapore

Blaise Genest
CNRS, IPAL, France
CNRS@CREATE, Singapore

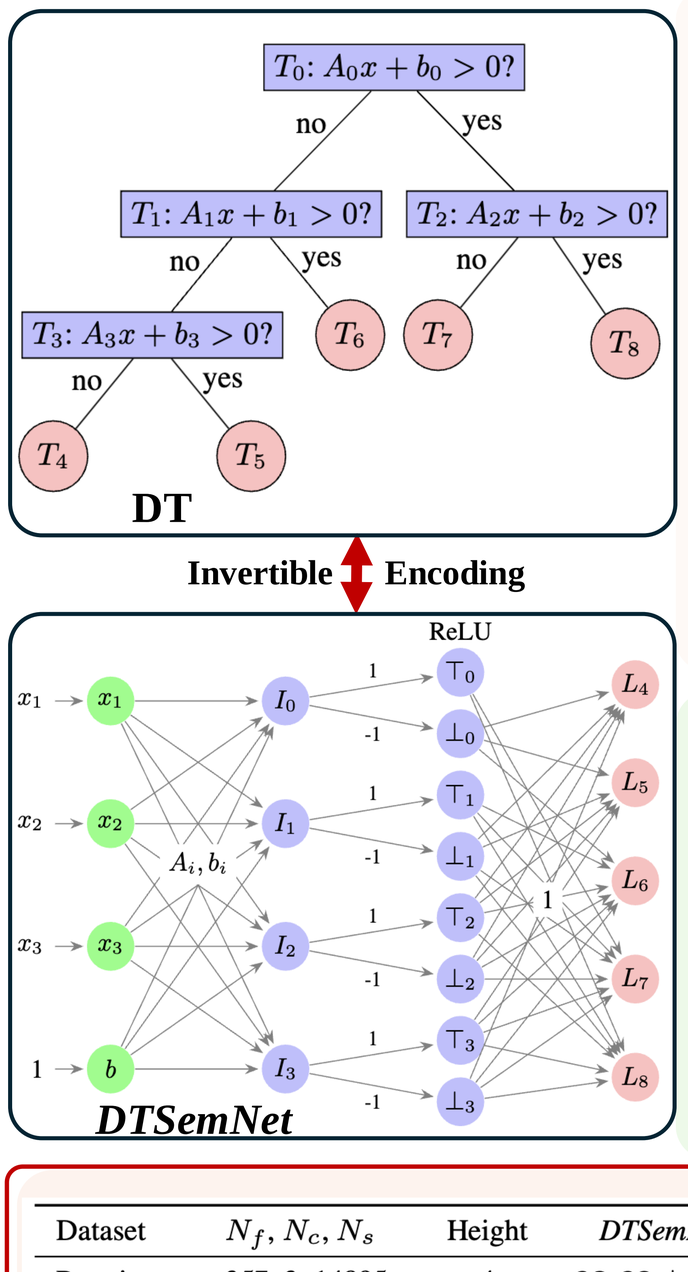
Arvind Easwaran
NTU Singapore

Ponnuthurai Nagaratnam Suganthan
KINDI Computing Research,
Qatar University, Qatar

email: subratpr001@e.ntu.edu.sg

TL;DR

We introduce ***DTSemNet***, a novel, semantically equivalent, and invertible encoding of oblique Decision Trees (DTs) as Neural Networks (NNs). Unlike traditional DT training methods, ***DTSemNet*** leverages standard vanilla gradient descent for training, which leads to more efficient and accurate DT learning.

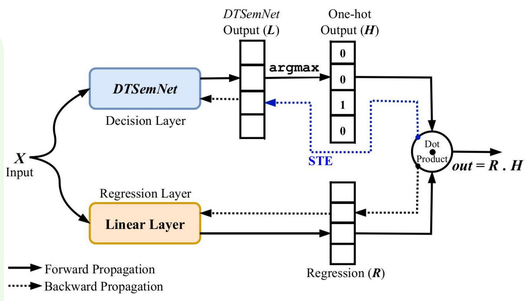


Introduction

- Decision Trees (DTs) excel on tabular data due to their inductive bias toward non-smooth functions [1].
- Gradient descent is the most efficient approach for training DTs [2].
- Existing gradient descent-based methods rely on approximations at decision nodes or during gradient computation using straight-through estimators (STE) [3].
- *DTSemNet* overcomes approximations by encoding oblique DTs as NN in a semantically equivalent way.
- *DTSemNet* is extended to regression tasks by using regressors at that leaf.
- *DTSemNet* can be integrated with the existing Reinforcement Learning (RL) framework.

DTSemNet: *Decision Tree Semantic Network*

- The learnable weights of DT have a one-to-one mapping to the first layer of *DTSemNet*, while some of the weights in *DTSemNet* are fixed.
- For any given input to DT and *DTSemNet*, the classification decisions made by DT and *DTSemNet* are the same.
- *DTSemNet* is adapted for regression by simultaneously learning the parameters of linear regression at each leaf and the decision nodes to the most appropriate leaf.

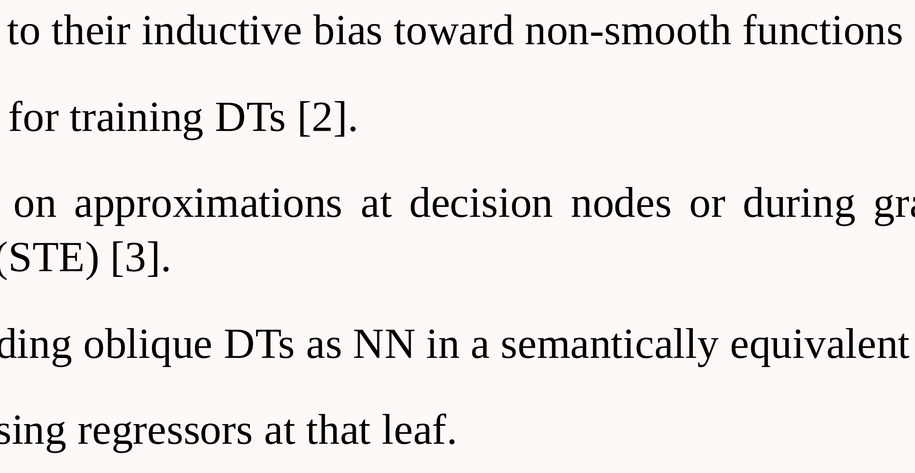


DTSemNet for Regression

Results

Dataset	N_f, N_c, N_s	Height	*DTSemNet*	DGT	TAO	CART
Protein	357, 3, 14895	4	**68.60 ± 0.22**	67.80 ± 0.40	68.41 ± 0.27	57.53 ± 0.00
SatImages	36, 6, 3104	6	**87.55 ± 0.59**	86.64 ± 0.95	87.41 ± 0.33	84.18 ± 0.30
Segment	19, 7, 1478	8	**96.10 ± 0.53**	95.86 ± 1.16	95.01 ± 0.86	94.23 ± 0.86
Pendigits	16, 10, 5995	8	**97.02 ± 0.32**	96.36 ± 0.25	96.08 ± 0.34	89.94 ± 0.34
Connect4	126, 3, 43236	8	**82.03 ± 0.39**	79.52 ± 0.24	81.21 ± 0.25	74.03 ± 0.60
MNIST	780, 10, 48000	8	**96.16 ± 0.14**	94.00 ± 0.36	95.05 ± 0.16	85.59 ± 0.06
SensIT	100, 3, 63058	10	**84.29 ± 0.11**	83.67 ± 0.23	82.52 ± 0.15	78.31 ± 0.00
Letter	16, 26, 10500	10	**89.19 ± 0.29**	86.13 ± 0.72	87.41 ± 0.41	70.13 ± 0.08

DTSemNet performs statistically significantly better than other approaches across all classification tasks.



The loss landscape of *DTSemNet* and DGT shows *DTSemNet* has better generalization (flatter loss landscape in *DTSemNet*).

Dataset	*DTSemNet*	DGT	TAO	CRO-DT
MNIST	306 (96.1)	288 (94.0)	1200 (95.0)	4659 (58.2)
DryBean	4.4 (91.4)	3.8 (89.0)	NA (83.2)	1300 (77.9)

DTSemNet has a faster training time compared to non-gradient-based learning approaches.

Dataset	N_f, N_s	Height	*DTSemNet*	DGT-Linear	DGT	TAO-Linear	CART
Abalone	10, 2004	5	2.135 ± 0.03	2.144 ± 0.03	2.15 ± 0.026 (6)	**2.07 ± 0.01**	2.29 ± 0.034
Comp-Active	21, 3932	5	2.645 ± 0.18	2.645 ± 0.15	2.91 ± 0.149 (6)	**2.58 ± 0.02**	3.35 ± 0.221
Ailerons	40, 5723	5	**1.66 ± 0.01**	1.67 ± 0.017	1.72 ± 0.016 (6)	1.74 ± 0.01	2.01 ± 0.00
CTSlice	384, 34240	5	1.45 ± 0.12	1.78 ± 0.25	2.30 ± 0.166 (10)	**1.16 ± 0.02**	5.78 ± 0.224
YearPred	90, 370972	6	**8.99 ± 0.01**	9.02 ± 0.025	9.05 ± 0.012 (8)	9.08 ± 0.03	9.69 ± 0.00
PDBBind	2052, 9013	2	**1.33 ± 0.017**	1.34 ± 0.013	1.39 ± 0.017 (6)	NA	1.55 ± 0.00
Microsoft	136, 578729	5	**0.766 ± 0.00**	**0.766 ± 0.00**	0.772 ± 0.00 (8)	NA	0.771 ± 0.00

For regression tasks, *DTSemNet* is either the best-performing or the second-best approach.

Environments	N_f, N_a	*Height*	*DTSemNet*	Deep RL	DGT	ICCT	VIPER
CartPole	4, 2	4	**500 ± 0**	**500 ± 0**	**500 ± 0**	496 ± 0.3	499.95 ± 0.05
Acrobot	6, 3	4	**−82.5 ± 1.05**	−84 ± 0.84	−83.1 ± 1.88	−88.6 ± 1.77	−83.92 ± 1.59
LunarLander	8, 4	5	**252.5 ± 3.9**	245 ± 14.5	183.6 ± 14.6	−85 ± 16.3	86.73 ± 7.93
Zerglings	32, 30	6	**15.54 ± 2.07**	10.47 ± 0.23	8.21 ± 1.03	9.40 ± 1.10	10.61 ± 0.46
Cont. LunarLander	8, 2 dim.	4	**277.24 ± 2.09**	276.12 ± 1.45	*scalar*: 131.92 ± 51.49 *linear*: 267.9 ± 9.37	255.57 ± 4.19	NA
Bipedal Walker	24, 4 dim.	7	314.98 ± 3.35	**315.3 ± 6.91**	*scalar*: 78.33 ± 57.19 (8) *linear*: 244.5 ± 61.84 (8)	301.34 ± 3.09 (6)	NA

The performance of *DTSemNet* in RL tasks is comparable to or better than that of NNs.

Conclusion

- *DTSemNet* outperforms other gradient-based methods by avoiding approximations and trains faster than non-gradient-based DT methods.
- *DTSemNet*-classification reduces errors by over 10% on difficult tasks, while *DTSemNet*-regression is competitively accurate.
- *DTSemNet* policies in RL environments demonstrate high efficiency and often outperform NN policies.

References

[1] Grinsztajn, L., Oyallon, E., & Varoquaux, G., "Why do tree-based models still outperform deep learning on typical tabular data?," NeurIPS, 2022.
[2] G. A. K. (ajaykrishna karthikeyan), N. Jain, N. Natarajan, and P. Jain., "Learning accurate decision trees with bandit feedback via quantized gradient descent," TMLR, 2022.
[3] Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y., "Binarized neural networks," NeurIPS, 2016.