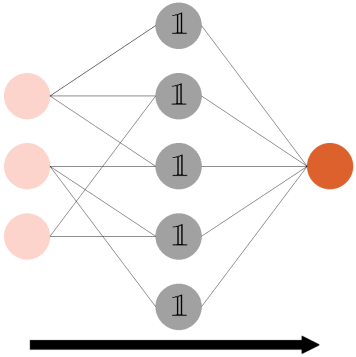
Seeking Interpretability and Explainability in Binary Activated Neural Networks

Benjamin Leblanc, **Pascal Germain**

July 17th, 2024

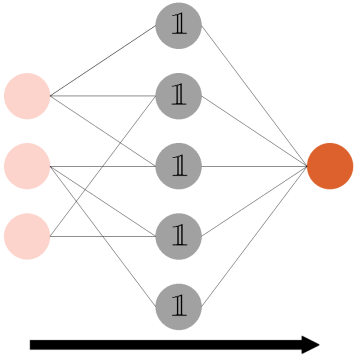
Simple neural networks

Simple neural networks



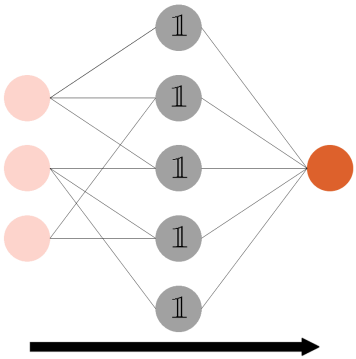
Simple neural networks

- Feedforward neural network



Simple neural networks

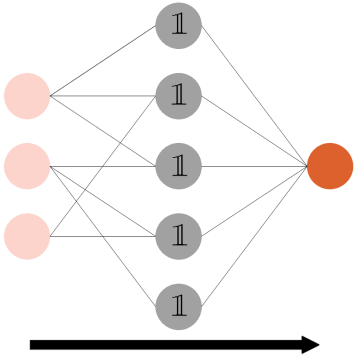
- Feedforward neural network
- Binary activations (threshold)



Simple neural networks

- Feedforward neural network
- Binary activations (threshold)

$$\mathbb{1}_{\{x\}} = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

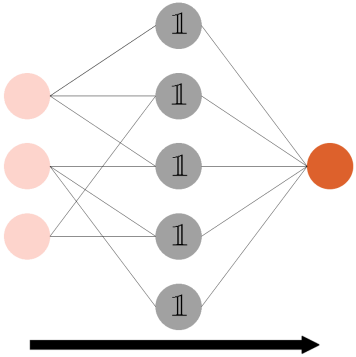


Simple neural networks

- Feedforward neural network
- Binary activations (threshold)

$$\mathbb{1}_{\{x\}} = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

- Shallow networks (1 hidden layer)

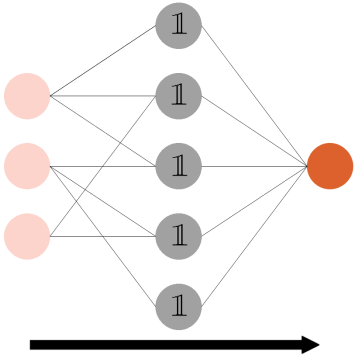


Simple neural networks

- Feedforward neural network
- Binary activations (threshold)

$$\mathbb{1}_{\{x\}} = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

- Shallow networks (1 hidden layer)
- Narrow networks

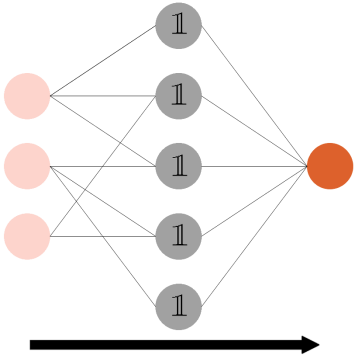


Simple neural networks

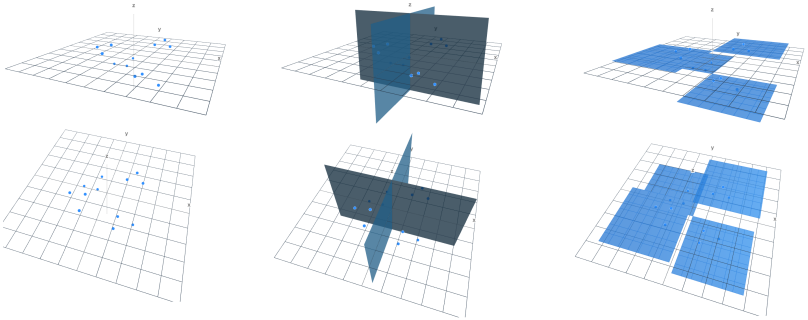
- Feedforward neural network
- Binary activations (threshold)

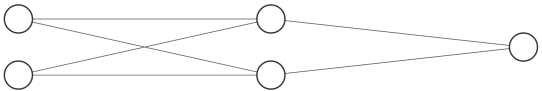
$$\mathbb{1}_{\{x\}} = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

- Shallow networks (1 hidden layer)
- Narrow networks
- Sparse (2)



Understanding binary activated neural networks





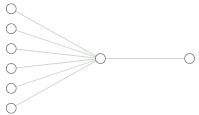
Characteristics

Characteristics

- For tackling regression tasks

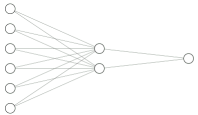
Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)



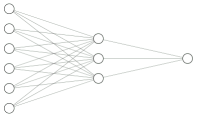
Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)



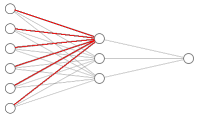
Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)



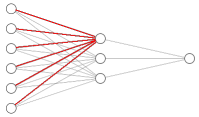
Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)



Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)

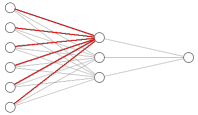


$$\underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)

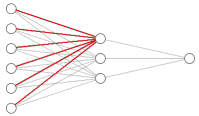


$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)
- Uses a L1 norm regularization

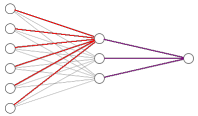


$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)
- Uses a L1 norm regularization

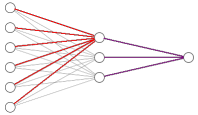


$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)
- Uses a L1 norm regularization
- Convergence guarantees on the train mean squared error

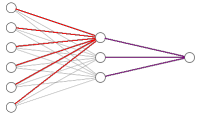


$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)
- Uses a L1 norm regularization
- Convergence guarantees on the train mean squared error
- No fixed architecture

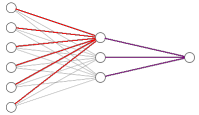


$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Characteristics

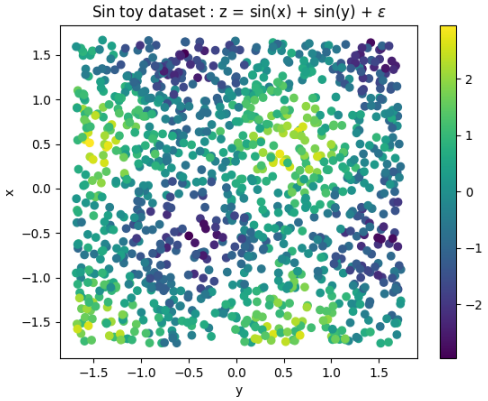
- For tackling regression tasks
- Greedy algorithm (inspired from Adaboost)
- Uses a L1 norm regularization
- Convergence guarantees on the train mean squared error
- No fixed architecture
- Fewer hyperparameter (no learning rate, batch size, ...)



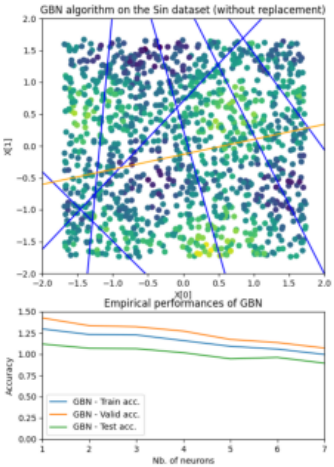
$$\approx \underset{\mathbf{w},b}{\operatorname{argmin}} \left(\frac{|\mathbf{y}_{-1}|}{m} \operatorname{Var}(\mathbf{y}_{-1}) + \frac{|\mathbf{y}_{+1}|}{m} \operatorname{Var}(\mathbf{y}_{+1}) \right),$$

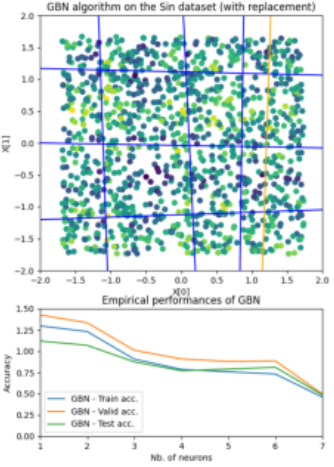
$$\text{with} \quad \mathbf{y}_{\pm 1} = \{ y_i \mid (\mathbf{x}_i, y_i) \in S,\ \operatorname{sgn}(\mathbf{w} \cdot \mathbf{x}_i + b) = \pm 1 \}$$

Tackling the drawbacks of greediness



Tackling the drawbacks of greediness





SHAP values

SHAP values

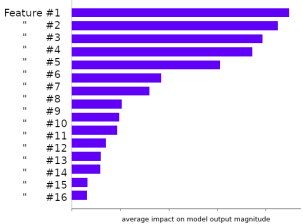
- Goal: quantifying the contribution (magnitude, impact) of each feature to a prediction

SHAP values

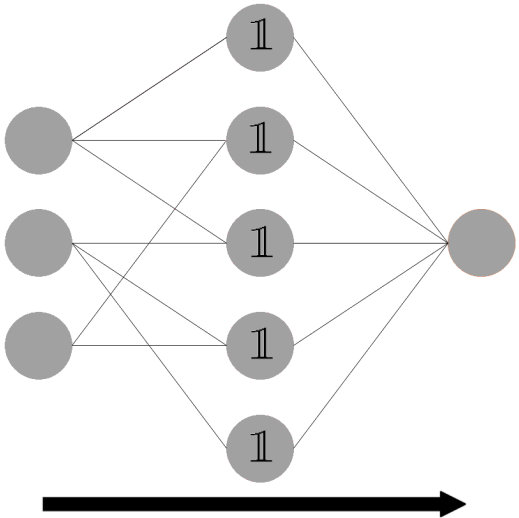
- Goal: quantifying the contribution (magnitude, impact) of each feature to a prediction
- At a prediction or a dataset level of aggregation

SHAP values

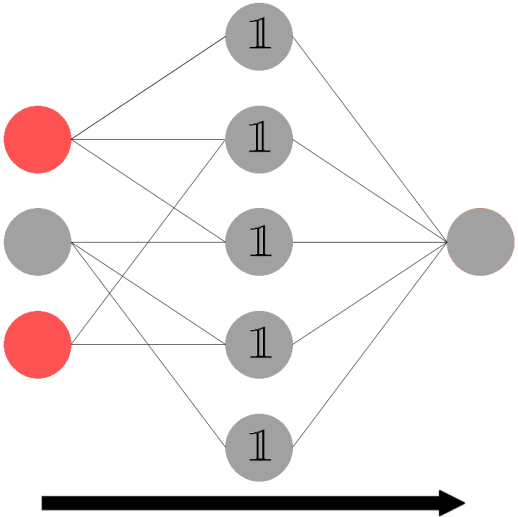
- Goal: quantifying the contribution (magnitude, impact) of each feature to a prediction
- At a prediction or a dataset level of aggregation



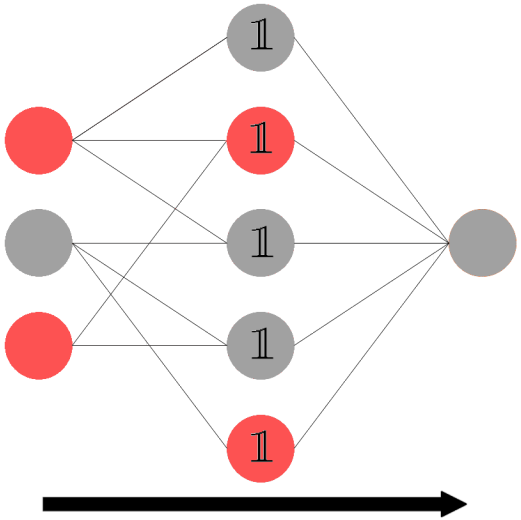
Explaining the networks: SHAP values



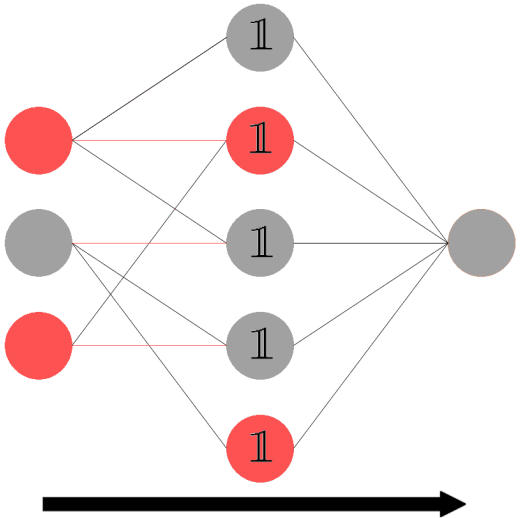
Explaining the networks: SHAP values



Explaining the networks: SHAP values



Explaining the networks: SHAP values



Testing our *interpretable* approach

The task

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)
- Retained features (out of eight):

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)
- Retained features (out of eight):
  - The median age of a house within a block (MedAge)

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)
- Retained features (out of eight):
  - The median age of a house within a block (MedAge)
  - The total number of bedrooms within a block (TotalBed)

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)
- Retained features (out of eight):
  - The median age of a house within a block (MedAge)
  - The total number of bedrooms within a block (TotalBed)
  - The median income for households within a block (measured in 1k USD) (MedInc)

Testing our *interpretable* approach

The task

- Task: predicting the cost of a house (measured in 100Ks of USD)
- Retained features (out of eight):
  - The median age of a house within a block (MedAge)
  - The total number of bedrooms within a block (TotalBed)
  - The median income for households within a block (measured in 1k USD) (MedInc)

$$
\begin{aligned}
&B(\text{MedInc, MedAge, TotalBed}) = \\
&1.00 + \\
&1.27 \cdot \mathbb{1}_{\{0.08 \cdot \text{TotalBed} + \text{MedInc} > 65.46\}} + \\
&1.01 \cdot \mathbb{1}_{\{0.59 \cdot \text{TotalBed} + \text{MedInc} > 63.56\}} + \\
&0.60 \cdot \mathbb{1}_{\{\text{MedInc} > 28.20\}} + \\
&0.36 \cdot \mathbb{1}_{\{\text{TotalBed} > 622.0\}} + \\
&0.27 \cdot \mathbb{1}_{\{\text{MedAge} > 20.0\}}
\end{aligned}
$$

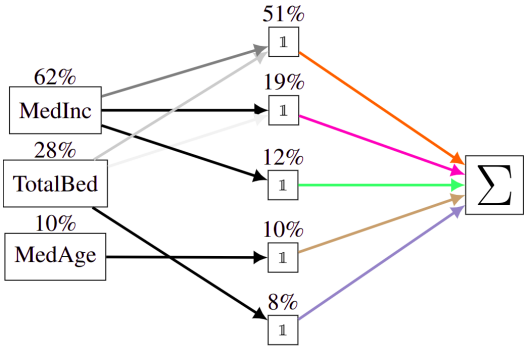
Testing our *interpretable* approach

The criterion

- Additive model
- Simple interactions between features
- Simple feature manipulations
- Small amount of features

$$
\begin{aligned}
&B(\text{MedInc, MedAge, TotalBed}) = \\
&1.00 + \\
&1.27 \cdot \mathbb{1}_{\{0.08 \cdot \text{TotalBed} + \text{MedInc} > 65.46\}} + \\
&1.01 \cdot \mathbb{1}_{\{0.59 \cdot \text{TotalBed} + \text{MedInc} > 63.56\}} + \\
&0.60 \cdot \mathbb{1}_{\{\text{MedInc} > 28.20\}} + \\
&0.36 \cdot \mathbb{1}_{\{\text{TotalBed} > 622.0\}} + \\
&0.27 \cdot \mathbb{1}_{\{\text{MedAge} > 20.0\}}
\end{aligned}
$$

Testing our *interpretable* approach



$$B(\text{MedInc, MedAge, TotalBed}) =$$
$$1.00 +$$
$$1.01 \cdot \mathbb{1}_{\{0.59 \cdot \text{TotalBed} + \text{MedInc} > 63.56\}} +$$
$$1.27 \cdot \mathbb{1}_{\{0.08 \cdot \text{TotalBed} + \text{MedInc} > 65.46\}} +$$
$$0.60 \cdot \mathbb{1}_{\{\text{MedInc} > 28.20\}} +$$
$$0.27 \cdot \mathbb{1}_{\{\text{MedAge} > 20.0\}} +$$
$$0.36 \cdot \mathbb{1}_{\{\text{TotalBed} > 622.0\}}$$

Experiments

Experiments

- Performances experiments

Dataset	BGN			BC			BNN*			BNN+			Bi-real net*			QN		
	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l
bike hour [13]	3573.67	259	2	21881.79	1000	3	1948.73	1000	3	**1712.45**	1000	2	1818.36	1000	3	2008.47	1000	2
diabete [12]	**3755.93**	6	1	9134.43	500	1	8919.54	1000	1	8999.32	500	2	8816.83	100	1	8811.23	500	2
housing [25]	**0.32**	158	1	1.65	100	2	2.19	500	1	2.09	1000	1	2.36	500	1	2.16	500	1
hung pox [44]	223.76	3	1	235.75	100	2	250.07	100	2	229.83	100	3	242.55	100	3	**214.27**	500	3
ist. stock [1]	2.27	5	1	22787.86	100	1	2.98	1000	3	1.95	1000	1	2.42	1000	2	**1.94**	1000	1
parking [49]	**6954.87**	1000	1	176286.81	1000	1	32705.94	500	3	11684.12	1000	3	16543.25	1000	2	107347.98	1000	2
power p.[52][31]	17.18	187	2	13527.9	500	2	18.34	1000	1	**16.66**	1000	2	16.77	1000	2	17.14	1000	2

Experiments

- Performances experiments
  - Is state-of-the-art when it comes to performances

Dataset	BGN			BC			BNN*			BNN+			Bi-real net*			QN		
	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l
bike hour [13]	3573.67	259	2	21881.79	1000	3	1948.73	1000	3	**1712.45**	1000	2	1818.36	1000	3	2008.47	1000	2
diabete [12]	**3755.93**	6	1	9134.43	500	1	8919.54	1000	1	8999.32	500	2	8816.83	100	1	8811.23	500	2
housing [25]	**0.32**	158	1	1.65	100	2	2.19	500	1	2.09	1000	1	2.36	500	1	2.16	500	1
hung pox [44]	223.76	3	1	235.75	100	2	250.07	100	2	229.83	100	3	242.55	100	3	**214.27**	500	3
ist. stock [1]	2.27	5	1	22787.86	100	1	2.98	1000	3	1.95	1000	1	2.42	1000	2	**1.94**	1000	1
parking [49]	**6954.87**	1000	1	176286.81	1000	1	32705.94	500	3	11684.12	1000	3	16543.25	1000	2	107347.98	1000	2
power p.[52][31]	17.18	187	2	13527.9	500	2	18.34	1000	1	**16.66**	1000	2	16.77	1000	2	17.14	1000	2

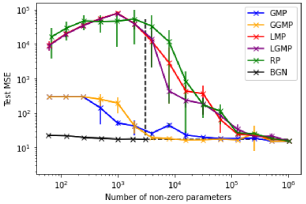
Experiments

- Performances experiments
  - Is state-of-the-art when it comes to performances
  - Truly sparse and small predictors

Dataset	BGN			BC			BNN*			BNN+			Bi-real net*			QN		
	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l	Error_T	d^*	l
bike hour [13]	3573.67	259	2	21881.79	1000	3	1948.73	1000	3	**1712.45**	1000	2	1818.36	1000	3	2008.47	1000	2
diabete [12]	**3755.93**	6	1	9134.43	500	1	8919.54	1000	1	8999.32	500	2	8816.83	100	1	8811.23	500	2
housing [25]	**0.32**	158	1	1.65	100	2	2.19	500	1	2.09	1000	1	2.36	500	1	2.16	500	1
hung pox [44]	223.76	3	1	235.75	100	2	250.07	100	2	229.83	100	3	242.55	100	3	**214.27**	500	3
ist. stock [1]	2.27	5	1	22787.86	100	1	2.98	1000	3	1.95	1000	1	2.42	1000	2	**1.94**	1000	1
parking [49]	**6954.87**	1000	1	176286.81	1000	1	32705.94	500	3	11684.12	1000	3	16543.25	1000	2	107347.98	1000	2
power p.[52][31]	17.18	187	2	13527.9	500	2	18.34	1000	1	**16.66**	1000	2	16.77	1000	2	17.14	1000	2

Experiments

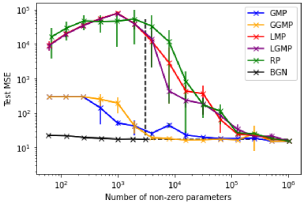
- Performances experiments
  - Is state-of-the-art when it comes to performances
  - Truly sparse and small predictors
- Pruning experiments



Experimentations found in the article

Experiments

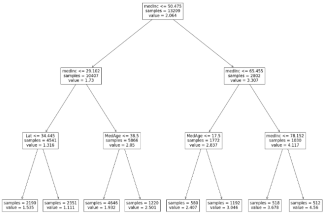
- Performances experiments
  - Is state-of-the-art when it comes to performances
  - Truly sparse and small predictors
- Pruning experiments
  - Better than pruned large networks



Experimentations found in the article

Experiments

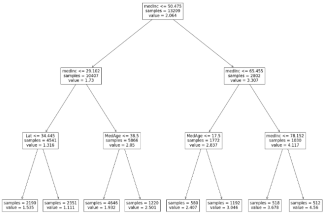
- Performances experiments
  - Is state-of-the-art when it comes to performances
  - Truly sparse and small predictors
- Pruning experiments
  - Better than pruned large networks
- Interpretability experiments



Experimentations found in the article

Experiments

- Performances experiments
  - Is state-of-the-art when it comes to performances
  - Truly sparse and small predictors
- Pruning experiments
  - Better than pruned large networks
- Interpretability experiments
  - Arguably more interpretable than regression tree



Conclusion

Artificial neural networks can be **interpretable** predictors...

Conclusion

Artificial neural networks can be **interpretable** predictors...
... When trained with such a goal in mind!

References

[1] J. Lin, C. Gan, and S. Han, "Defensive quantization: When efficiency meets robustness," CoRR, vol. abs/1904.08444, 2019.

[2] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or -1," 2016.

Thank you for your attention :)