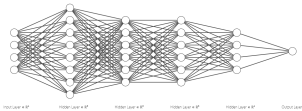
PAC-Bayesian Learning of Aggregated Binary Activated Neural Networks with Probabilities over Representations

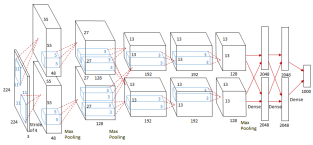
Benjamin Leblanc Louis Fortier-Dubois Gaël Letarte
Pascal Germain François Laviolette

June 6th, 2023

Introduction

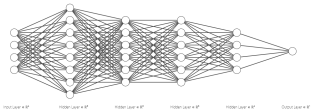
- Neural networks: why so difficult to analyse?

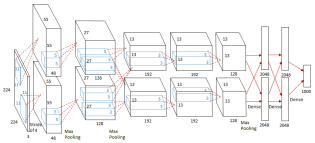




Introduction

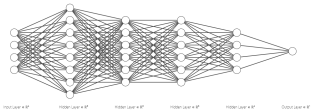
- Neural networks: why so difficult to analyse?
- Simplification as a solution

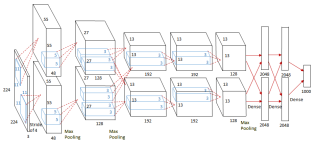




Introduction

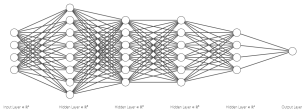
- Neural networks: why so difficult to analyse?
- Simplification as a solution
- Probability distribution over the weights of the model

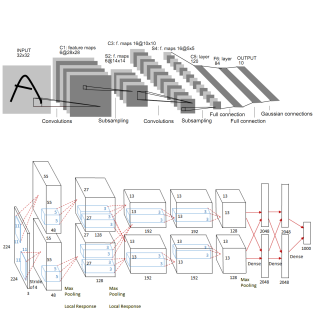




Introduction

- Neural networks: why so difficult to analyse?
- Simplification as a solution
- Probability distribution over the weights of the model
- Based on the work of *Letarte et al.*[1]

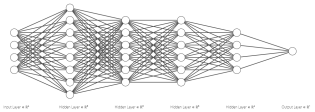


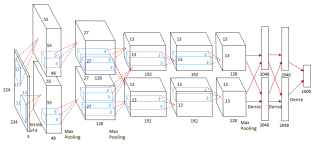


[1] Gaël Letarte, Pascal Germain, Benjamin Guedj, and François Laviolette. Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks. NeurIPS, 2019.

Introduction

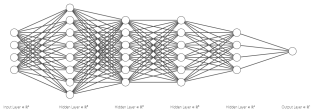
- PAC-Bayesian theory

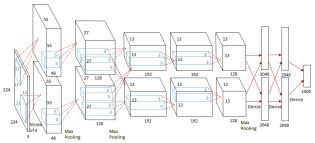




Introduction

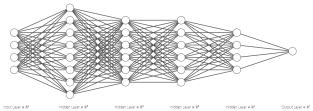
- PAC-Bayesian theory
  - Non-trivial generalization bound

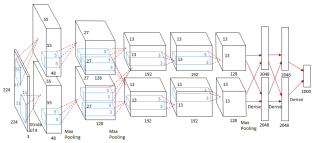




Introduction

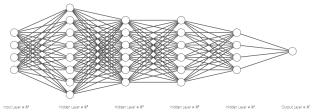
- PAC-Bayesian theory
  - Non-trivial generalization bound
- Contributions

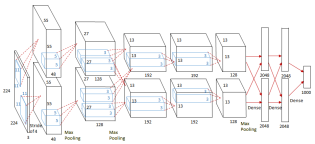




Introduction

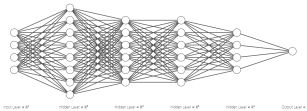
- PAC-Bayesian theory
  - Non-trivial generalization bound
- Contributions
  - Get rid of an approximation

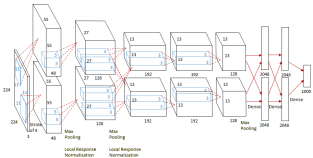




Introduction

- PAC-Bayesian theory
  - Non-trivial generalization bound
- Contributions
  - Get rid of an approximation
  - Constant-time prediction relative to network depth





Plan of the presentation

Plan of the presentation

1. Preliminary notions

Plan of the presentation

1. Preliminary notions
2. Neural networks aggregations

Plan of the presentation

1. Preliminary notions
2. Neural networks aggregations
3. Experimentation

Plan of the presentation

1. Preliminary notions
2. Neural networks aggregations
3. Experimentation
4. Question period

Preliminary notions

Neural networks

Neural networks

- Fully connected networks

Neural networks

- Fully connected networks
- Binary classification tasks

Neural networks

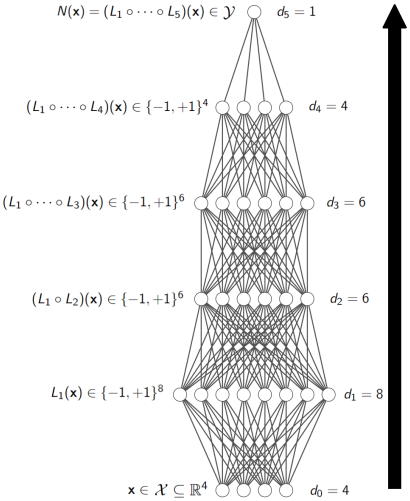
- Fully connected networks
- Binary classification tasks
- $F(\mathbf{x}) = (L_1 \circ \cdots \circ L_l)(\mathbf{x})$

Neural networks

- Fully connected networks
- Binary classification tasks
- $F(\mathbf{x}) = (L_1 \circ \cdots \circ L_l)(\mathbf{x})$
- $L_k(\mathbf{x}) = g_k(\mathbf{W}_k\mathbf{x} + \mathbf{b}_k)\ \forall k \in \{1, \ldots, l\}$

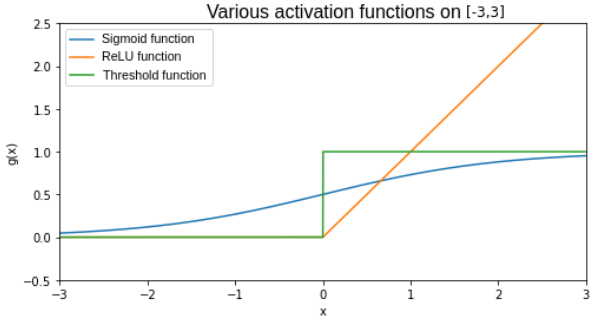
Neural networks

- Fully connected networks
- Binary classification tasks
- $F(\mathbf{x}) = (L_1 \circ \cdots \circ L_l)(\mathbf{x})$
- $L_k(\mathbf{x}) = g_k(\mathbf{W}_k\mathbf{x} + \mathbf{b}_k)\ \forall k \in \{1, \ldots, l\}$



Binary activation

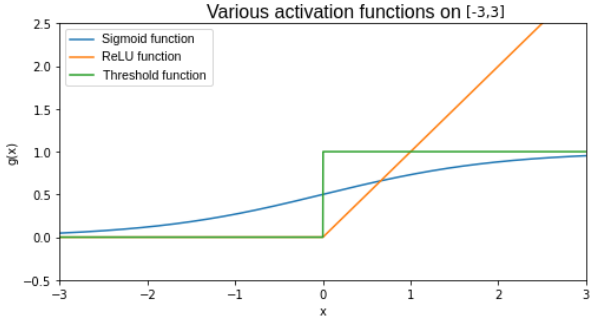
Why?



Binary activation

Why?

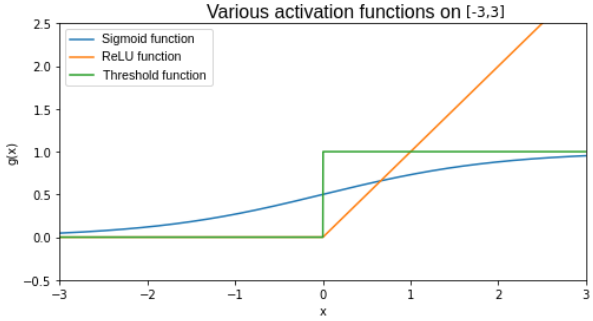
- Computation time savings



Binary activation

Why?

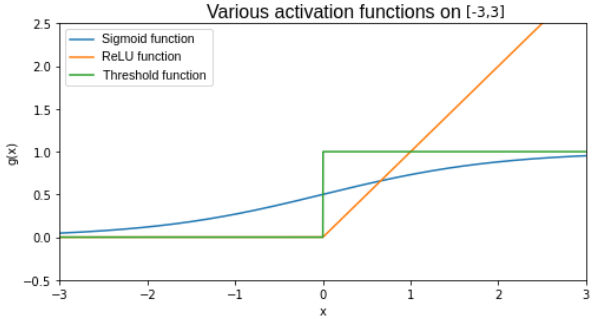
- Computation time savings
- Gains in memory required



Binary activation

Why?

- Computation time savings
- Gains in memory required
- Networks less prone to overfitting [1] [2]



[1] J. Lin, C. Gan, and S. Han, "Defensive quantization: When efficiency meets robustness". CoRR, 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". NIPS, 2016.

Dissecting binary activated neural networks

Definition : The parts of a binary activated networks

Dissecting binary activated neural networks

Definition : The parts of a binary activated networks

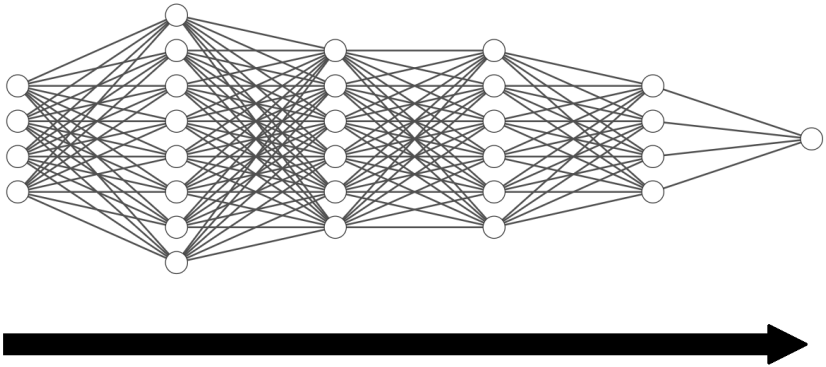
- The first part is L_1

Dissecting binary activated neural networks

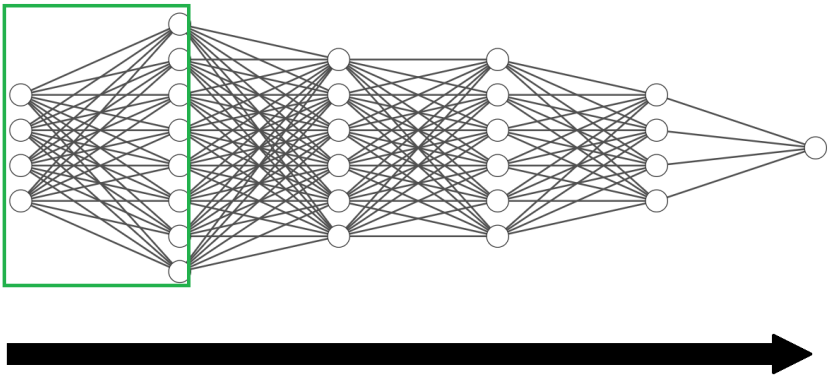
Definition : The parts of a binary activated networks

- The first part is L_1
- The second part is $(L_2 \circ \cdots \circ L_l)$

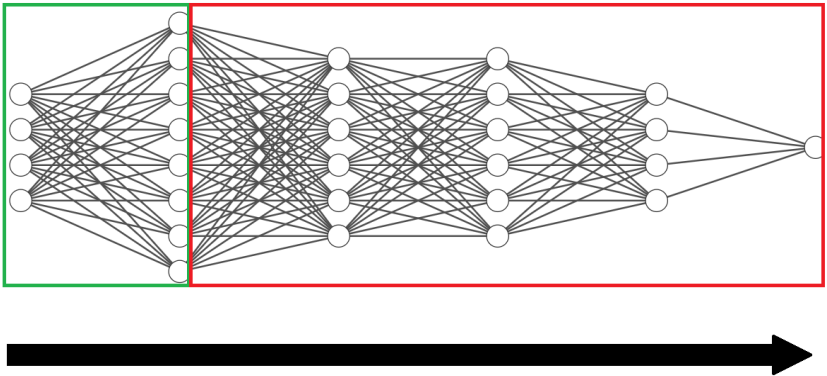
Binary activations - A visualization



Binary activations - A visualization



Binary activations - A visualization



Dissecting binary activated neural networks

Definition : The parts of a binary activated networks

- The first part (L_1)
- The second part $(L_2 \circ \cdots \circ L_l)$

We have $L_1 : \mathcal{X} \to \{-1, +1\}^{d_1}$ and $(L_2 \circ \cdots \circ L_l) : \{-1, +1\}^{d_1} \to \mathcal{Y}$

Dissecting binary activated neural networks

Definition : The parts of a binary activated networks

- The first part (L_1) assigns a *group* to a given example
- The second part ($L_2 \circ \cdots \circ L_l$)

We have $L_1 : \mathcal{X} \to \{-1, +1\}^{d_1}$ and $(L_2 \circ \cdots \circ L_l) : \{-1, +1\}^{d_1} \to \mathcal{Y}$

Dissecting binary activated neural networks

Definition : The parts of a binary activated networks

- The first part (L_1) assigns a *group* to a given example
- The second part ($L_2 \circ \cdots \circ L_l$) assigns a *prediction* to each group

We have $L_1 : \mathcal{X} \to \{-1, +1\}^{d_1}$ and $(L_2 \circ \cdots \circ L_l) : \{-1, +1\}^{d_1} \to \mathcal{Y}$

Neural networks aggregation

Neural networks aggregation

Bayesian aspect

We suppose the weights $B = \langle \mathbf{W}_k \rangle_{k=1}^{l}$ to follow a Gaussian probability distribution, centred on B and with isotropic covariance matrix.

Neural networks aggregation

Bayesian aspect

We suppose the weights $B = \langle \mathbf{W}_k \rangle_{k=1}^{l}$ to follow a Gaussian probability distribution, centred on B and with isotropic covariance matrix.

The output of a neuron

The expected output of a sign function applied Gaussian distributed variable:

$$\mathop{\mathbf{E}}_{\mathbf{v}\sim\mathcal{N}(\mathbf{w},\mathbf{I})} \operatorname{sgn}(\mathbf{v}\cdot\mathbf{a}) = \operatorname{erf}\left(\frac{\mathbf{w}\cdot\mathbf{a}}{\sqrt{2}||\mathbf{a}||}\right) ,$$

where $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ is the gaussian error function.

Neural networks aggregation

Bayesian aspect

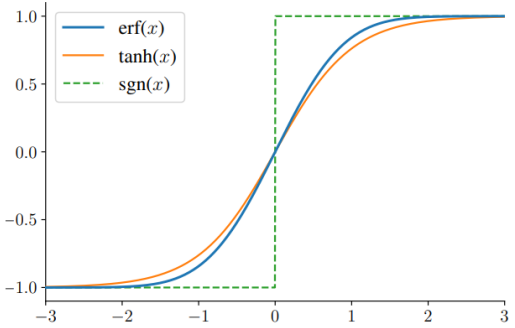
We suppose the weights $B = \langle \mathbf{W}_k \rangle_{k=1}^{l}$ to follow a Gaussian probability distribution, centred on B and with isotropic covariance matrix.

The output of a neuron

The expected output of a sign function applied Gaussian distributed variable:

$$\mathop{\mathbf{E}}_{\mathbf{v}\sim\mathcal{N}(\mathbf{w},\mathbf{I})} \text{sgn}(\mathbf{v}\cdot\mathbf{a}) = \text{erf}\left(\frac{\mathbf{w}\cdot\mathbf{a}}{\sqrt{2}||\mathbf{a}||}\right) ,$$

where $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ is the gaussian error function.



Neural networks aggregation

Probability of observing a given output for a given neuron

The binary activation acts as a Bernoulli distribution. $P(s_i \in \{-1, +1\})$ is given by

$$\Pr(L_{k;i}(\mathbf{a}) = s) = \frac{1}{2} + \frac{s}{2}\mathrm{erf}\left(\frac{\mathbf{w}_{k,i} \cdot \mathbf{a}}{\sqrt{2}\|\mathbf{a}\|}\right), i \in \{1, \ldots, d_k\}, k \in \{1, \ldots, l\}.$$

Neural networks aggregation

Probability of observing a given output for a given neuron

The binary activation acts as a Bernoulli distribution. $P(s_i \in \{-1, +1\})$ is given by

$$\Pr(L_{k;i}(\mathbf{a}) = s) = \frac{1}{2} + \frac{s}{2}\mathrm{erf}\left(\frac{\mathbf{w}_{k,i} \cdot \mathbf{a}}{\sqrt{2}\|\mathbf{a}\|}\right), i \in \{1, \ldots, d_k\}, k \in \{1, \ldots, l\}.$$

Probability of observing a given output for a given layer

Let $\Pr((L_1 \circ \cdots \circ L_k)(\mathbf{a}) = \mathbf{s}) = L^{\mathbf{s}}_{1,k}(\mathbf{a})$. An iterative computation can be used:

$$\Pr(L^{\mathbf{s}}_{1,k}(\mathbf{a})) = \begin{cases} \prod_{i=1}^{d_1} \Pr(L^{s_i}_{1,i}(\mathbf{a})) & \text{si } k = 1, \\ \sum_{\bar{\mathbf{s}}} \Pr(L^{\mathbf{s}}_{1,k}(\mathbf{a}) \mid L^{\bar{\mathbf{s}}}_{1,k-1}(\mathbf{a})) \Pr(L^{\bar{\mathbf{s}}}_{1,k-1}(\mathbf{a})) & \text{sinon.} \end{cases}$$

The exact computation of ABNet

Dynamic programming approach

The exact computation of ABNet

Dynamic programming approach

First layer: $\mathbf{P}_1(\mathbf{x}) = \left[\Pr(L_1^{\mathbf{s}}(\mathbf{a})) \right]_{\mathbf{s} \in R_1}$

The exact computation of ABNet

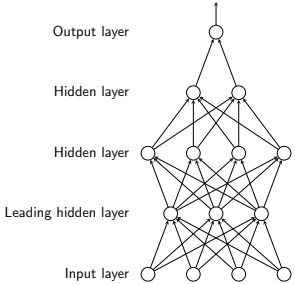
Dynamic programming approach

$$\text{First layer: } \mathbf{P}_1(\mathbf{x}) = \left[\Pr(L_1^{\mathbf{s}}(\mathbf{a})) \right]_{\mathbf{s} \in R_1}$$

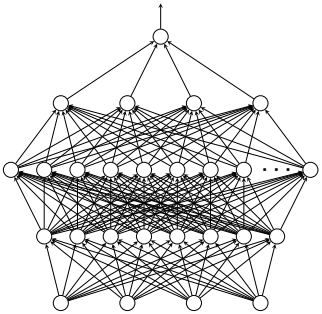
$$\begin{aligned} \text{Subsequent layers: } \mathbf{P}_k &= \left[\Pr(L_{1,k-1}^{\mathbf{s}}(\mathbf{a})) \right]_{\mathbf{s} \in R_k} \\ &= \mathbf{\Psi}_k \cdot \mathbf{P}_{k-1} \ , \end{aligned}$$

$$\text{with } \mathbf{\Psi}_k = \left[\Pr(L_{1,k}^{\mathbf{s}}(\mathbf{a}) \mid L_{1,k-1}^{\bar{\mathbf{s}}}(\mathbf{a})) \right]_{\mathbf{s} \in R_k, \bar{\mathbf{s}} \in R_{k-1}}$$

Neural networks aggregation



(a) A typical neural network



(b) The architecture of **ABNet**, with underlying network from (a).

Neural networks aggregation

Compact ABNet

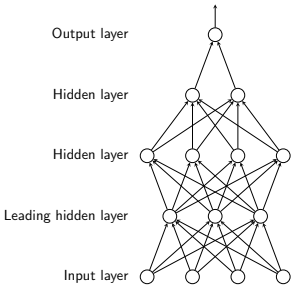
$$\Pr(L^s_{1,l}(\mathbf{a})) = s \cdot (\mathbf{\Psi}_l(\mathbf{\Psi}_{l-1}(\dots \mathbf{\Psi}_3(\mathbf{\Psi}_2\mathbf{P}_1(\mathbf{x}))\dots)))$$

Neural networks aggregation

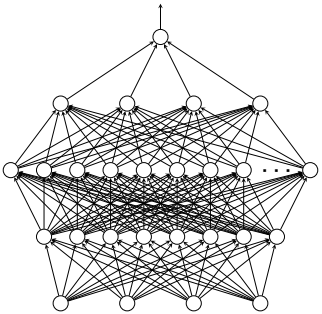
Compact ABNet

$$\begin{aligned}\Pr(L^s_{1,l}(\mathbf{a})) &= s \cdot (\mathbf{\Psi}_l(\mathbf{\Psi}_{l-1}(\dots \mathbf{\Psi}_3(\mathbf{\Psi}_2\mathbf{P}_1(\mathbf{x}))\dots))) \\ &= \underbrace{s \cdot \mathbf{\Psi}_l \cdot \mathbf{\Psi}_{l-1} \cdot \dots \cdot \mathbf{\Psi}_3 \cdot \mathbf{\Psi}_2}_{\mathbf{H}_s} \cdot \mathbf{P}_1(\mathbf{x}))\end{aligned}$$

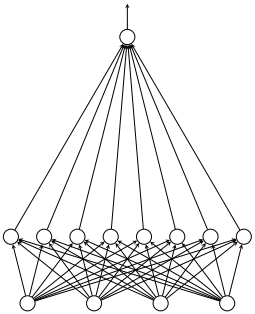
Neural networks aggregation



(a) A typical neural network.



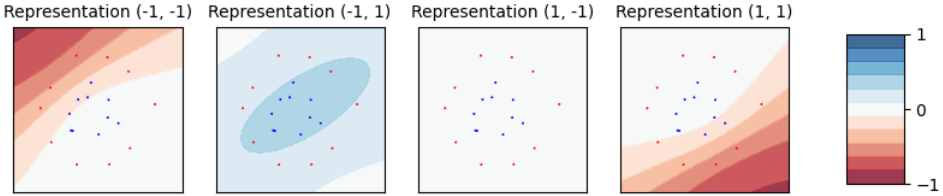
(b) The architecture of **ABNet**, with underlying architecture from (a).



(c) The structure of **compact ABNet**, obtain from the network presented in (b).

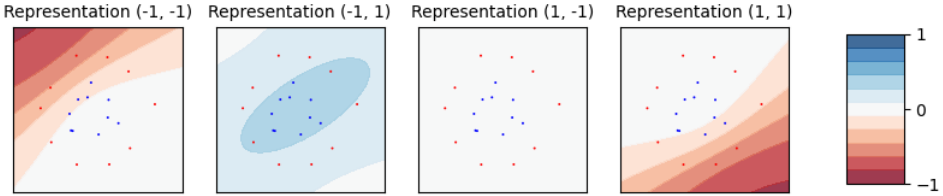
Experimentations

ABNet - a pratical case

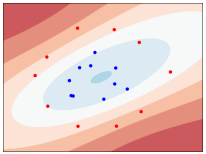


(a) For different values of $\mathbf{x}$, the probability of obtaining a certain representation as an output of the first hidden layer, obtained with $\mathbf{P}_1(\mathbf{x})$, multiplied by $\mathbf{H}$.

ABNet - a pratical case

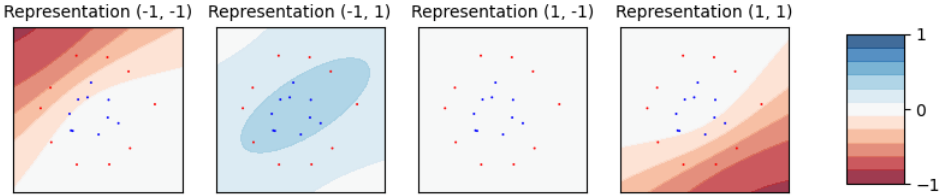


(a) For different values of $\mathbf{x}$, the probability of obtaining a certain representation as an output of the first hidden layer, obtained with $\mathbf{P}_1(\mathbf{x})$, multiplied by $\mathbf{H}$.

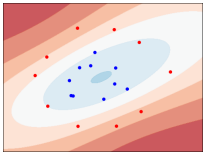


(b) The output of the aggregation $\Pr(L_{1,l})$.

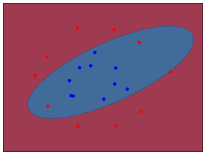
ABNet - a pratical case



(a) For different values of $\mathbf{x}$, the probability of obtaining a certain representation as an output of the first hidden layer, obtained with $\mathbf{P}_1(\mathbf{x})$, multiplied by $\mathbf{H}$.

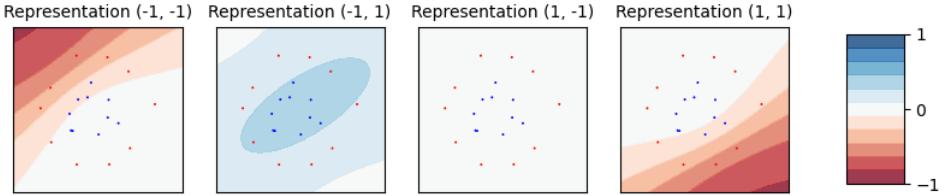


(b) The output of the aggregation $\Pr(L_{1,l})$.

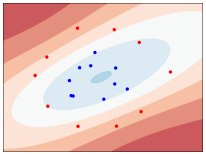


(c) ABNet as a classifier.

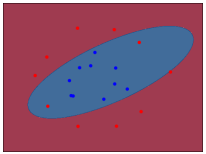
ABNet - a pratical case



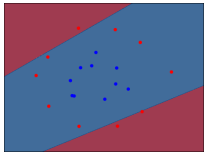
(a) For different values of $\mathbf{x}$, the probability of obtaining a certain representation as an output of the first hidden layer, obtained with $\mathbf{P}_1(\mathbf{x})$, multiplied by $\mathbf{H}$.



(b) The output of the aggregation $\Pr(L_{1,l})$.



(c) ABNet as a classifier.



(d) Unique network, centered on the *a posteriori* means of the Gaussian probability distribution.

Experimentations

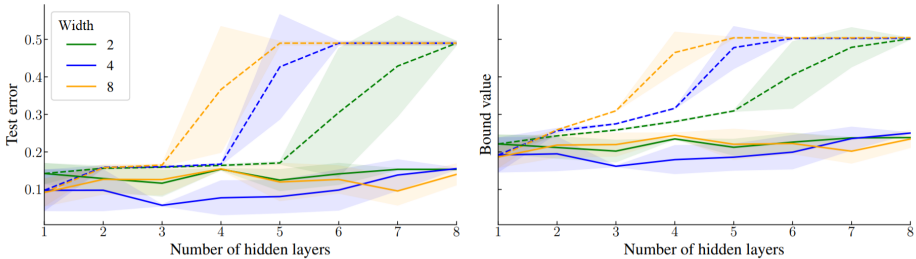


Figure: Impact of the depth for **PBGNet (dotted)** and **ABNet (continuous)** on the test error and the bound value depending on the width of the network on mnistLH. 5 random seeds.

Experimentations

Dataset	Model	$L-1$	d	Bound	Error_S	Error_T
ads	PBGNet	3	2	0.192 ± 0.004	0.140 ± 0.004	0.141 ± 0.012
	ABNet	3	2	0.192 ± 0.004	0.140 ± 0.004	0.141 ± 0.012
	PBGNet_ℓ	3	4	1.000 ± 0.001	0.018 ± 0.005	0.026 ± 0.004
	ABNet_ℓ	3	4	0.887 ± 0.064	0.015 ± 0.003	0.026 ± 0.003
	EBP	2	2	–	0.003 ± 0.002	0.040 ± 0.008
	BC	1	4	–	0.025 ± 0.005	0.031 ± 0.004
	BNN	1	8	–	0.037 ± 0.002	0.038 ± 0.004
mnistLH	PBGNet	1	8	0.186 ± 0.028	0.091 ± 0.037	0.092 ± 0.036
	ABNet	3	4	0.162 ± 0.001	0.056 ± 0.001	0.058 ± 0.002
	PBGNet_ℓ	3	8	1.000 ± 0.000	0.018 ± 0.003	0.038 ± 0.002
	ABNet_ℓ	2	8	0.998 ± 0.003	0.025 ± 0.008	0.042 ± 0.006
	EBP	3	8	–	0.016 ± 0.002	0.043 ± 0.002
	BC	2	8	–	0.023 ± 0.002	0.035 ± 0.001
	BNN	1	2	–	0.123 ± 0.005	0.133 ± 0.004

Questions / comments

Thank you for listening :) !