Machine Learning in HEP

(A gentle introduction)

Ritesh K. Singh

Indian Institute of Science Education & Research Kolkata
Mohanpur Campus, Nadia

May 18, 2026

Pre-school for ML4HEP V4

Online host: TIFR Mumbai

Statement of the Problem

Artifitial Neural Networks (ANN)

Statement of the Problem

Problems Types

Regression

Classification

Problems Types

Regression

- What is the mass (from histogram)?

Classification

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?

Classification

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?

Classification

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.

Classification

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?
- Is this jet b, q or g-type?

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?
- Is this jet b, q or g-type?
- Is this fat jet W, Z or h-type?

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?
- Is this jet b, q or g-type?
- Is this fat jet W, Z or h-type?
- Need to estimate the value of a variable from a set of possibilities.

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?
- Is this jet b, q or g-type?
- Is this fat jet W, Z or h-type?
- Need to estimate the value of a variable from a set of possibilities.
- Find the function describing decision boundary.

Problems Types

Regression

- What is the mass (from histogram)?
- What is the width (from histogram)?
- What is the couplings (σ and/or A)?
- Need to estimate the value of variable over a continuous range.
- Fit an obserable derived from the *data* to a function.

Classification

- Is this event signal type?
- Is this jet b, q or g-type?
- Is this fat jet W, Z or h-type?
- Need to estimate the value of a variable from a set of possibilities.
- Find the function describing decision boundary.

Need to approximate/estimate the said function.

Approximating/Estimating the function

Case 1:

Case 2:

Approximating/Estimating the function

Case 1:

Known function with unknown parameters

Case 2:

Approximating/Estimating the function

Case 1:

Known function with unknown parameters

Parameter extimation etc.

Case 2:

Approximating/Estimating the function

Case 1:

Known function with unknown parameters

Parameter extimation etc.

Case 2:

Unknown function with unknown parameters

Approximating/Estimating the function

Case 1:

Known function with unknown parameters

Parameter extimation etc.

Case 2:

Unknown function with unknown parameters

Need a parametrized approximant.

Universal Approximation Theorem

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \to \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \ldots$$

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \to \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \dots$$

This is an equation of a hyperplane
$P_0(\vec{x})$ which *approximates* $f(\vec{x})$ near $\vec{x}_0$.

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \to \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \ldots$$

This is an equation of a hyperplane
$P_0(\vec{x})$ which *approximates* $f(\vec{x})$ near $\vec{x}_0$.

Consider a set of fixed number of points
$\vec{x}_i \in D$ (D is a domain, subset of $\mathbb{R}^n$)

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \to \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \ldots$$

This is an equation of a hyperplane $P_0(\vec{x})$ which *approximates* $f(\vec{x})$ near $\vec{x}_0$.

Consider a set of fixed number of points $\vec{x}_i \in D$ (D is a domain, subset of $\mathbb{R}^n$)

and corresponding hyperplanes $P_i(\vec{x})$.

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \to \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \ldots$$

This is an equation of a hyperplane $P_0(\vec{x})$ which *approximates* $f(\vec{x})$ near $\vec{x}_0$.

Consider a set of fixed number of points $\vec{x}_i \in D$ (D is a domain, subset of $\mathbb{R}^n$)

and corresponding hyperplanes $P_i(\vec{x})$.

An envelope made out of $P_i(\vec{x})$'s approximates the original function $f(\vec{x})$

Universal Approximation Theorem

Cosider a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$
Near a point $\vec{x}_0$ it is given by

$$f(\vec{x}) = f(\vec{x}_0) + \vec{\partial} f(x_0) \cdot (\vec{x} - \vec{x}_0) + \ldots$$

This is an equation of a hyperplane $P_0(\vec{x})$ which *approximates* $f(\vec{x})$ near $\vec{x}_0$.

Consider a set of fixed number of points $\vec{x}_i \in D$ (D is a domain, subset of $\mathbb{R}^n$)

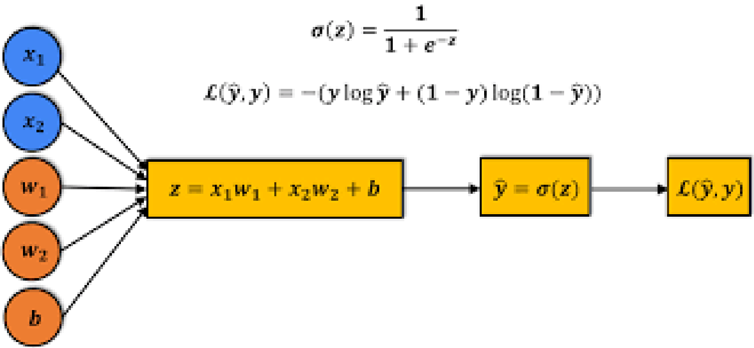
and corresponding hyperplanes $P_i(\vec{x})$.

An envelope made out of $P_i(\vec{x})$'s approximates the original function $f(\vec{x})$

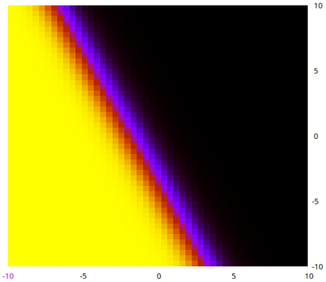
The approximant is alternating sequance of *linear maps* (hyperplane) and *non-linear maps* (envelope).

Artifitial Neural Networks (ANN)

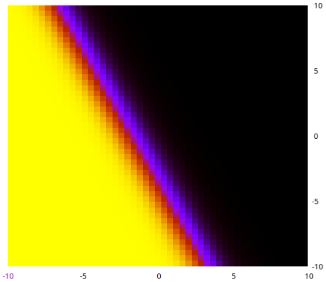
Logistic Regression

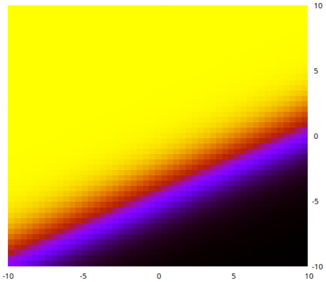


Logistic Regression

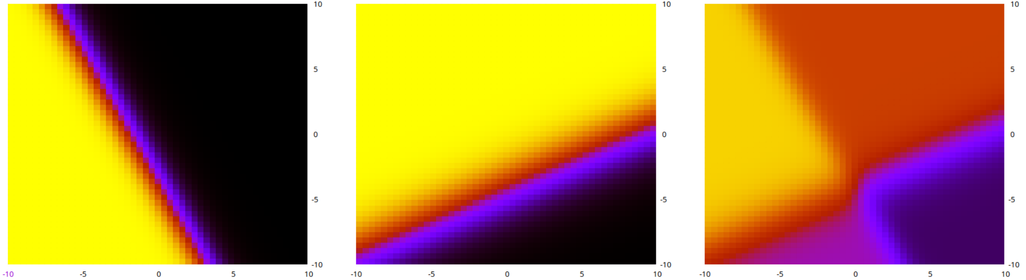


Logistic Regression

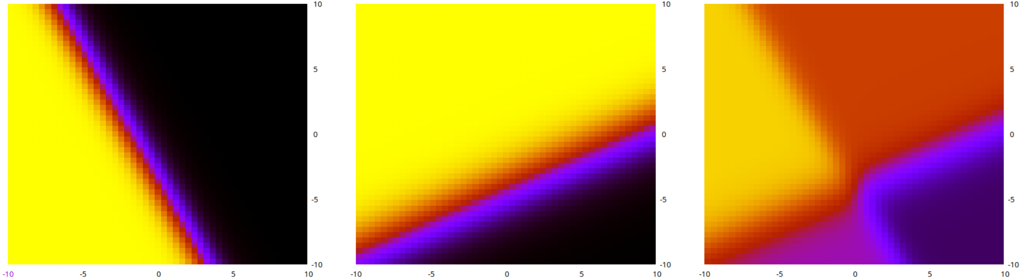




Logistic Regression

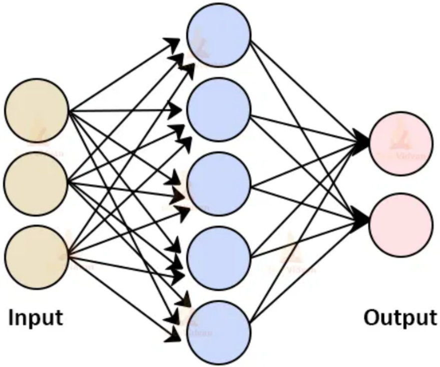


Logistic Regression

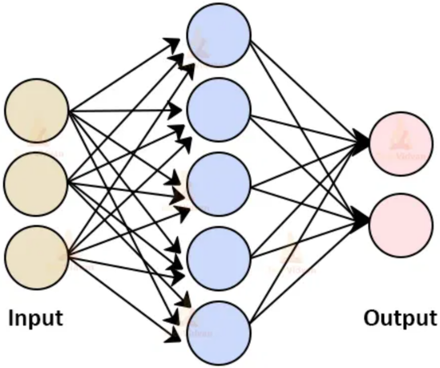


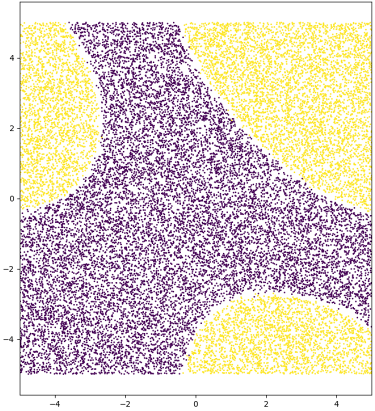
Combining several Logistic maps, we make an ANN.

Artifitial Neural Networks

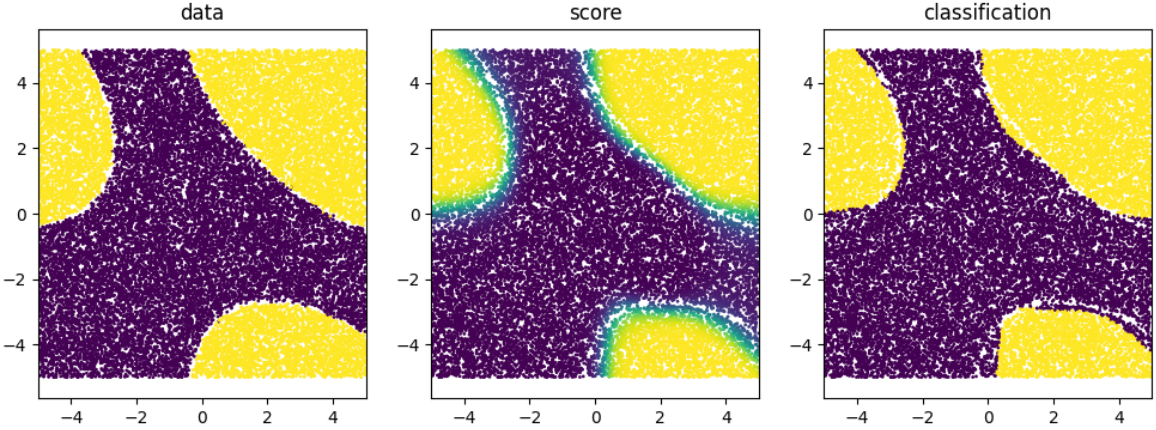


Artifitial Neural Networks

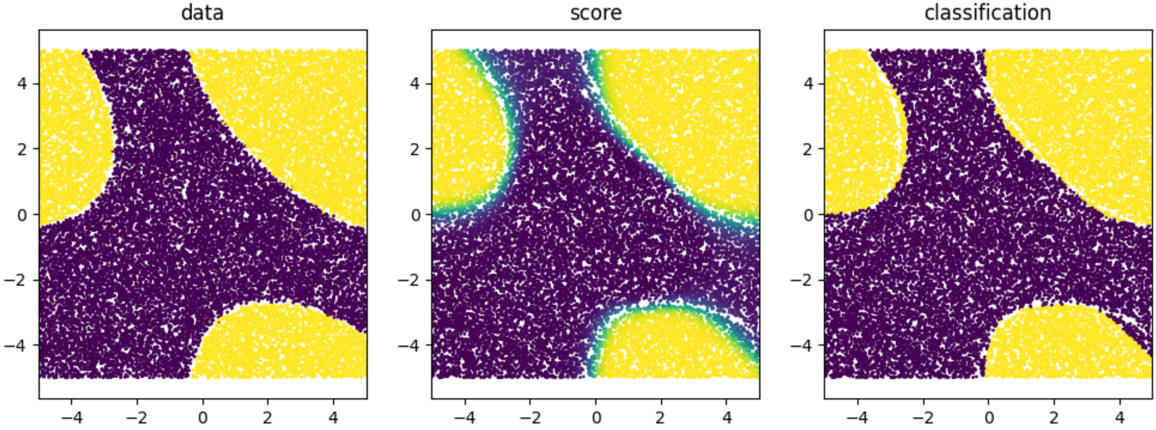




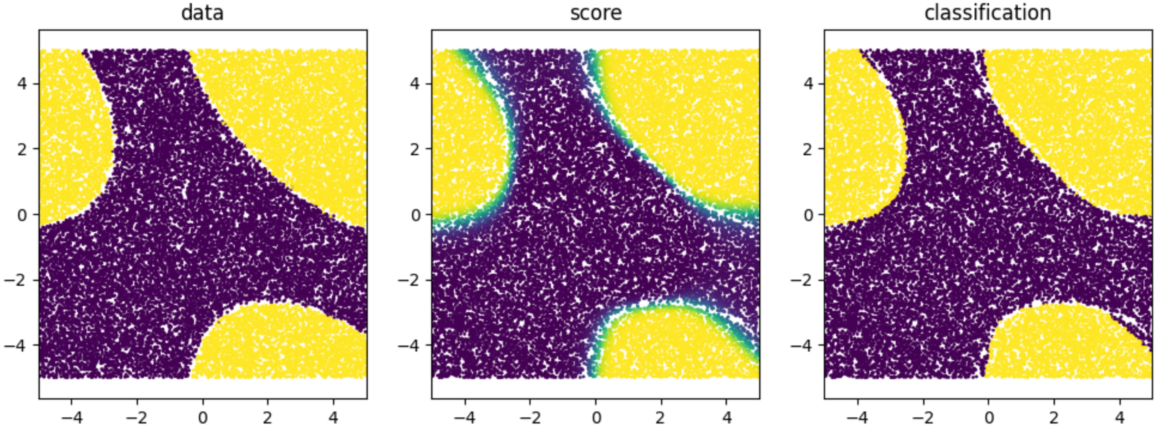
ANN: $N_{\text{hidden}} = 4$, $N_{\text{epoch}} = 10$, Accuracy $= \mathbf{97.4}\%$



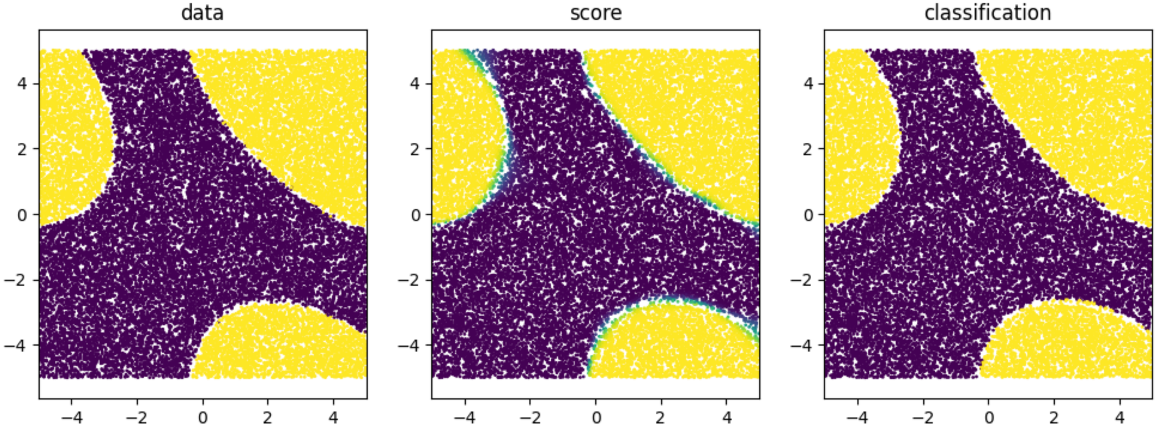
ANN: $N_{\text{hidden}} = 4$, $N_{\text{epoch}} = 20$, Accuracy $= \mathbf{97.9}\%$



ANN: $N_{\text{hidden}} = 4$, $N_{\text{epoch}} = 30$, Accuracy = $\mathbf{98.1}\%$



ANN: $N_{\text{hidden}} = 10$, $N_{\text{epoch}} = 10$, Accuracy = $\mathbf{99.7}\%$



Feature Engineering

Features: $(\mathbf{x}, \mathbf{y})$

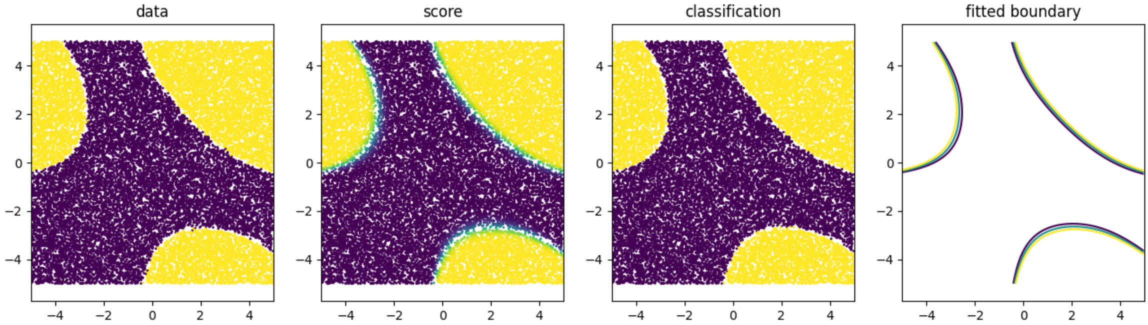
Decision boundary approximation with 4 neurons $\Rightarrow \sim 98.0\%$ accuracy
About 15 parameters to fit. Improves slowly with N_{epoch}

Decision boundary approximation with 10 neurons $\Rightarrow \sim 99.7\%$ accuracy
About 40 parameters to fit. May get stuck in local minima. Less **interpretable**.

Features: $(\mathbf{x}, \mathbf{y}, \mathbf{x}^2, \mathbf{y}^2, \mathbf{xy}, \mathbf{x}^2\mathbf{y}, \mathbf{xy}^2)$

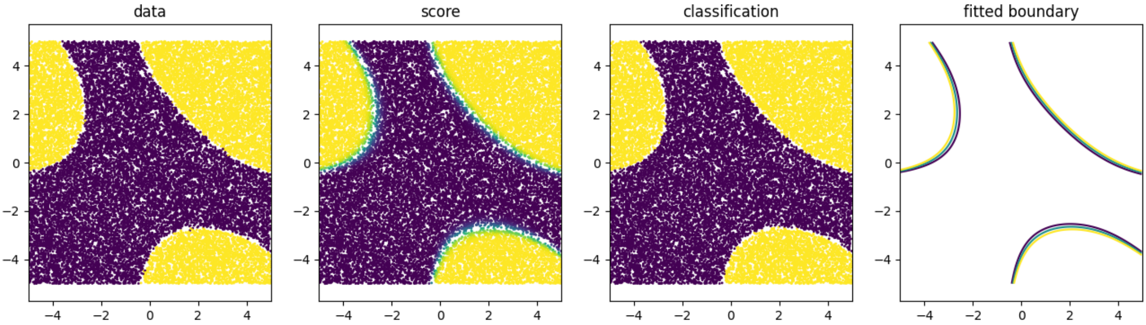
Decision boundary approximation with one neuron $\Rightarrow \sim 99.8\%$ accuracy
Less number of parameters to fit. Symmetries can help further. More **interpretable**.

ANN: $N_{\text{input}} = 7$, $N_{\text{epoch}} = 10$, Accuracy = $\mathbf{99.86}\%$, (non-linear)



x	y	x^2	y^2	xy	x^2y	xy^2
-0.031	0.055	**0.705**	**0.710**	0.051	**0.751**	**0.748**

ANN: $N_{\text{input}} = 4$, $N_{\text{epoch}} = 10$, Accuracy = **99.87**%, (non-linear + sym)



$x + y$	$x^2 + y^2$	xy	$x^2y + xy^2$
0.0069	**0.711**	-0.025	**0.776**