Exploration of Parameter Spaces Assisted by Machine Learning

Raymundo Ramos
Seoultech
(based on: A. Hammad, M. Park, R.R., P. Saha, arXiv:2207.09959)

The 2nd AEI Workshop for BSM and the 10th KIAS Workshop on Particle Physics and Cosmology, The Grand Sumorum, Jeju Island
November 17, 2022

The problem we want to solve

- Some high energy physics calculations (HEPC) take **a very long time/too much computational power**
- More parameters $\rightarrow$ **exponential** increase in the number of required points
- Multiple disconnected regions

How we want to solve it

- Neural networks (NN) as generic function approximations
- Training a NN could be more efficient than passing every single point through the HEPC
- Eventually, the accuracy of the NN is proportional to how much we care about the sampled regions
- Spend, relatively, **more time sampling regions of interest**
- *Just enough time* for low importance regions

Regression

- likelihood from predicting the results for observables

$$\hat{Y} = \mathcal{L}(\hat{O}_j(\Theta); c, \sigma)$$

- or predict the likelihood itself

$$\hat{Y} = \hat{\mathcal{L}}(\Theta)$$

- We need a **diverse set of points**
- with a **diverse set of results**

Points that need attention

- Few points $\rightarrow$ **bad predictions**
- Poorly sampled observable/likelihood $\rightarrow$ **bad predictions**
- Poorly distributed initial samples $\rightarrow$ **bad predictions**

Classification – Allowed region...

- Or any other condition
- Classify from likelihood, χ^2 or observables

$$Y = 1 \quad \textbf{if} \quad \mathcal{L} < \mathcal{L}_0 \quad \text{or} \quad \chi^2 < \chi_0^2 \quad \text{or} \quad O_{\text{min}} < O < O_{\text{max}} \quad \text{or} \quad \ldots$$

- We need to start with points in all classes

Points that need attention

- Accurate predictions require comparable amount of points inside and outside allowed region

Details of the process

Before the iterative process, we need a set of random points and their results to train a NN.

1. L: large set of points prior:

$$L \rightarrow \text{NN} \rightarrow \hat{Y}(L)$$

2. Select an smaller set, K, through a selection criteria

$$(L, \hat{Y}(L)) \rightarrow \text{selection criteria} \rightarrow (K, \hat{Y}(K))$$

3. Get the correct results, $Y(K)$, from the HEPC

$$K \rightarrow \text{HEPC} \rightarrow Y(K)$$

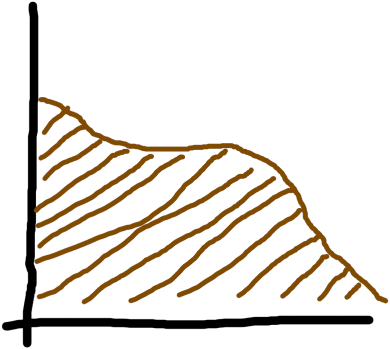
4. Train with set K and its results $Y(K)$

$$(K, Y(K)) \rightarrow \text{train the NN again}$$

Selection of points for HEPC

We want to pass a set of meaningful points to the HEPC.

- **Highest** predicted likelihood/**lowest** predicted χ^2
  - But keep **diversity** of observables/likelihood
- Points predicted with low likelihood/high χ^2 may be included as part of some **rectifying strategy**.
- Fraction of random points to find new regions



Selection of points for training

Training is also time consuming

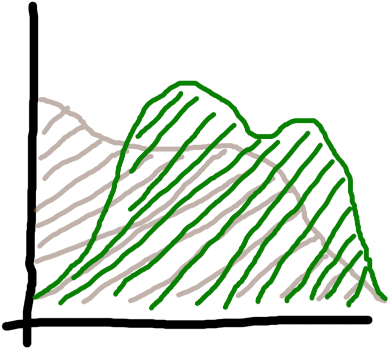
Required time depends non trivially on:

- **epochs**
- **number of hidden layers**
- **number of nodes**

And the number of points used for training also adds time

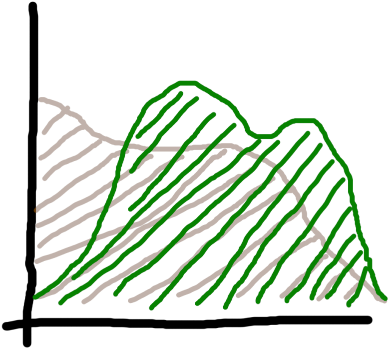
Selection of points for training – Regression

- wrongly predicted in group above: **rectify inaccurate predictions**
- What about points wrongly predicted with low likelihood/high χ^2
  - This needs a well thought strategy



Selection of points for training – Classification

- **True allowed**: Good certainty. These we are interested in
- **False allowed**: Confusing. These we want to correct
- **False excluded**: Confusing. These we want to correct
- *True excluded*: Good certainty. The region we care the least



Boosting initial convergence

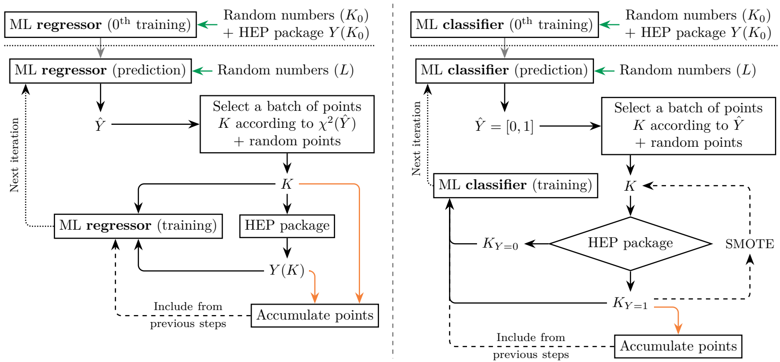
For the very few steps, predictions should be expected to be **mostly wrong**

Many options to improve initial convergence:

- **Naive/Brute force**: run more points to collect usable points
- Sample more points **around** known points in the target region
- Sample points **between** known points (**S**ynthetic **M**inority **O**versampling **Te**chnique, SMOTE) [Chawla et al, arXiv:1106.1813]

If these techniques work they should be needed **only in the first few iterations**.

Summary of the process



Applied to toy model

We tested these two processes against a simple toy model:

$$O_{3\mathrm{d}} = \left[2 + \cos\left(\frac{x_1}{7}\right)\cos\left(\frac{x_2}{7}\right)\cos\left(\frac{x_3}{7}\right)\right]^5$$

assuming a measured central value $c_{3\mathrm{d}} = 100$ with a standard deviation of $\sigma_{3\mathrm{d}} = 20$.

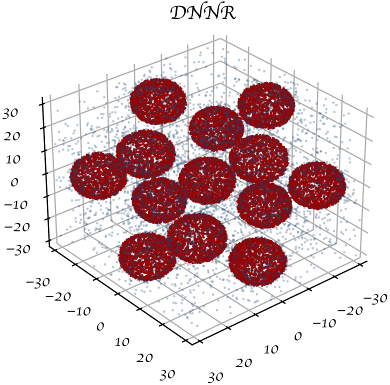
We define the likelihood for this toy model as

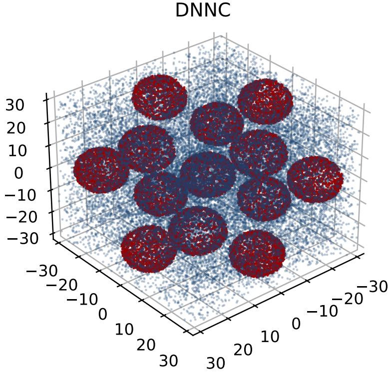
$$\mathcal{L} = exp\left[\frac{(O_{3\mathrm{d}} - c_{3\mathrm{d}})^2}{\sigma_{3\mathrm{d}}}\right]$$

and assume a region of interest where $\mathcal{L} > 0.9$, and all x_j in the range $[-10\pi, 10\pi]$

Applied to toy model, region coverage

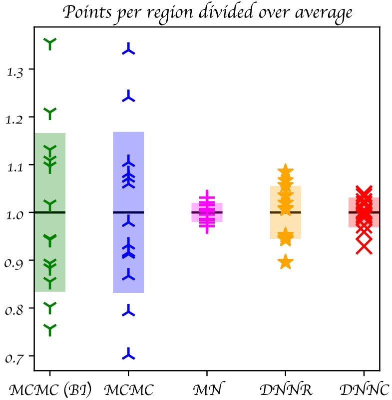
With this setup, there is a total of 13 disconnected regions, shaped like hollow shells.





4 hidden layers (ReLU), 1000 epochs, Adam, loss: (MAE, Binary cross-entropy), output layer activation: (linear, sigmoid)

Applied to toy model



Learning the Higgs signal strength in the 2HDM

The two Higgs doublet models (2HDM) [Lee, PRD **8**, 1226] are extensions of the standard model scalar sector

$$\phi_1 = \begin{pmatrix} \eta_1^+ \\ (v_1 + h_1 + i h_3)/\sqrt{2} \end{pmatrix}, \qquad \phi_2 = \begin{pmatrix} \eta_2^+ \\ (v_2 + h_2 + i h_4)/\sqrt{2} \end{pmatrix}.$$

Avoid FCNC by assuming a softly broken global Z_2 symmetry [Glashow, Weinberg, PRD **15**, 1958 (1977)] where $(\phi_1, \phi_2) \to (\phi_1, -\phi_2)$

$$\begin{aligned} V_\phi =& m_{11}^2(\phi_1^\dagger\phi_1) + m_{22}^2(\phi_2^\dagger\phi_2) - \left[m_{12}^2(\phi_1^\dagger\phi_2) + \text{h.c.}\right] + \lambda_1(\phi_1^\dagger\phi_1)^2 \\ &+ \lambda_2(\phi_2^\dagger\phi_2)^2 + \lambda_3(\phi_1^\dagger\phi_1)(\phi_2^\dagger\phi_2) + \lambda_4(\phi_1^\dagger\phi_2)(\phi_2^\dagger\phi_1) \\ &+ \frac{1}{2}\left[\lambda_5(\phi_1^\dagger\phi_2)^2 + \text{H.c.}\right], \end{aligned}$$

where m_{12}^2 softly breaks the Z_2 symmetry

Numerical scan details

Scanned parameters and ranges

$$0 \leq \lambda_1 \leq 10, \quad 0 \leq \lambda_2 \leq 0.2, \quad -10 \leq \lambda_3 \leq 10, \quad -10 \leq \lambda_4 \leq 10,$$
$$-10 \leq \lambda_5 \leq 10, \quad 5 \leq \tan\beta \leq 45, \quad -3000 \leq \frac{m_{12}^2}{\mathrm{GeV}^2} \leq 0\,,$$

Tools used

- `SPheno` to obtain the mass spectrum
- `HiggsBounds` to obtain limits on the Higgses [Bechtle et al, arXiv:1507.06706]
- `HiggsSignals` to obtain a χ^2 for the signals and mass of the Higgs [Bechtle et al, arXiv:1403.1582]

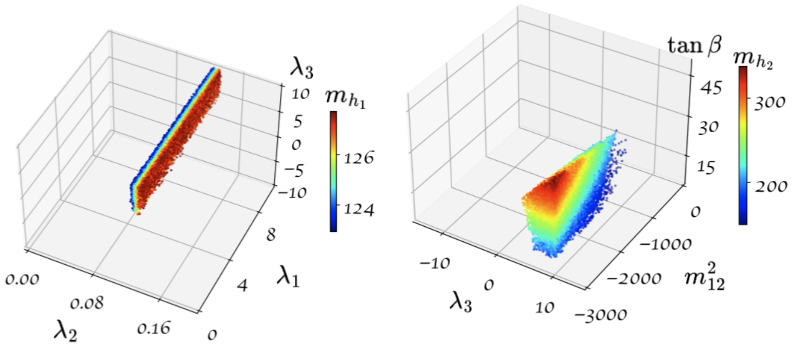
Numerical scan details

We assume our target region as all the points with $\chi^2 < 95$.

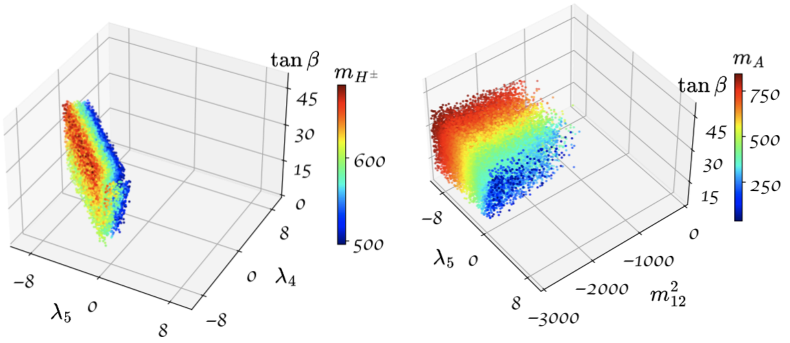
- Classification: $Y = 1$ for $\chi^2 < 95$
- Accumulated points: 20 000
- **4 hidden layers, 100 nodes**
  - **ReLU** activation function
- Output layer, 1 node
  - **Sigmoid** activation function
- Train **1000 epochs** per iteration
- Loss: **Binary cross-entropy**
- Optimizer: **Adam**
  - learning rate: 0.001

In every step, the classifier suggests $K = 300$ points to the HEPC from a larger set $L = 100\,000$.

Numerical scan results



Numerical scan results



The code

Implementation using `tensorflow`

- https://github.com/AHamamd150/MLscanner

What to do with this tool

This tool could be good for

- Adjusting complicated allowed regions
- Reduce the amount of calls to a time consuming calculation
- Compare against an ever increasing amount of experimental tests

What to do with this tool

This tool could be *great* for

- An study where we already have a sense of the parameter space
  - Update limits to new data
  - Test future expectations of a model
- Anything where a precise and fast estimation of observables/likelihood could be employed

What NOT to do with this tool

This tool CANNOT

- Precisely estimate parameter distributions (yet)
- Replace other tools or packages

Thanks for listening!