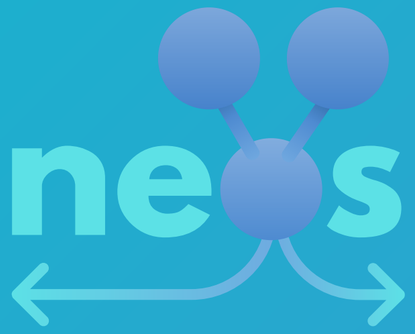


physics analysis as a differentiable program

Nathan Simpson, Lukas Heinrich

neos

Analysis with a *neural network observable*

The difference?
Uses the **expected CLs** as the loss function

Why haven't we done this before?

Loss functions need to be differentiable.

Backpropagation =>

Chain rule.

$$\frac{\partial \,\text{loss}}{\partial\, \varphi} = \frac{\partial \,\text{loss}}{\partial \,\text{likelihood}} \times \frac{\partial \,\text{likelihood}}{\partial \,\text{model parameters}} \times \frac{\partial \,\text{model parameters}}{\partial \,\text{cut values}} \times \ldots$$

Every step needs to be differentiable.

Not possible!

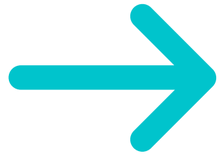
- Histograms ❌
- Model construction (HistFactory) ❌
- Profile likelihood fit ❌

Until now...

- Histograms

- Model construction (HistFactory)

- Profile likelihood fit



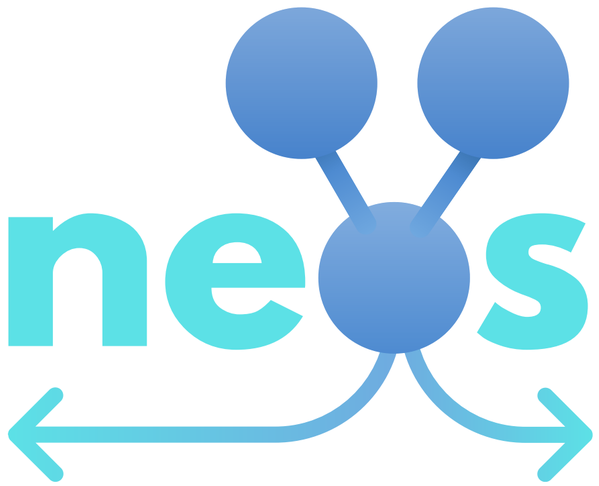
made differentiable using

Kernel density estimates!

Automatic differentiation software! (**jax + pyhf**)

Fixed point differentiation!

Mix it all together:

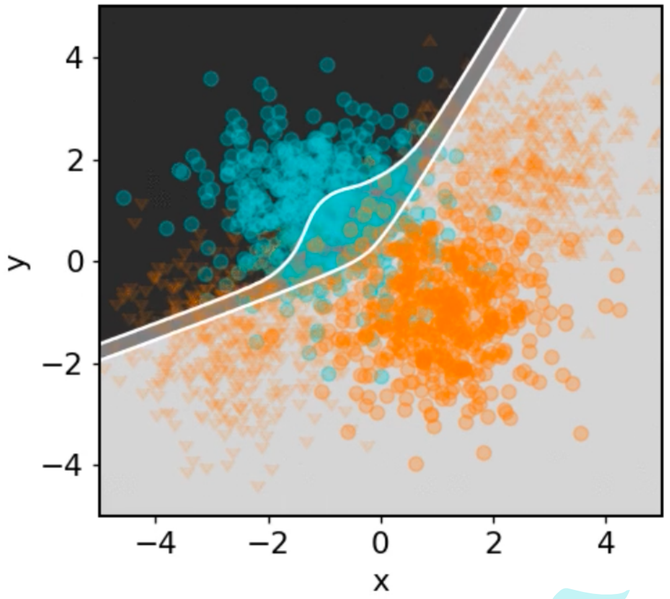


nice
~~neural~~ end-to-end
optimised statistics

github.com/gradhep/neos

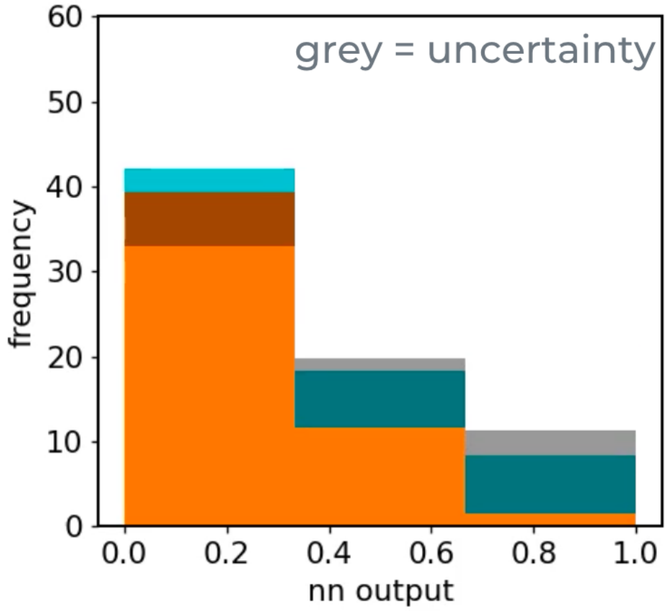
1)

predict one number per event



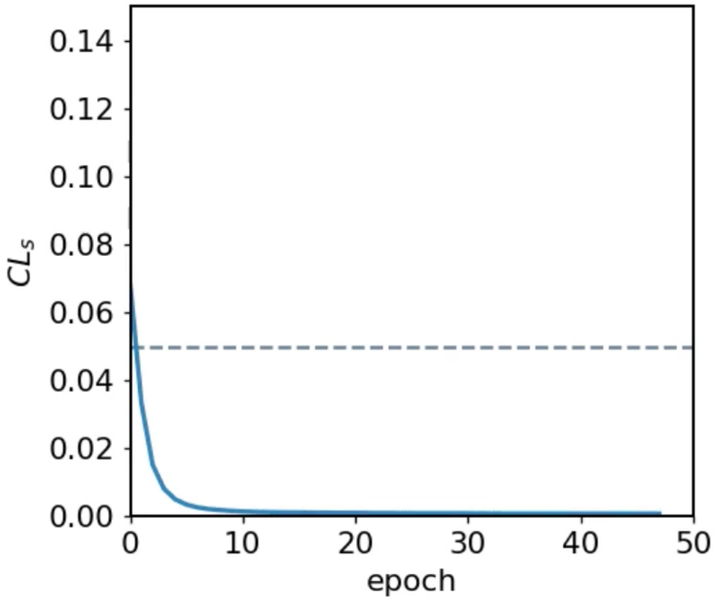
2)

histogram that number over a batch of events



3)

make statistical model + calculate expected CLs

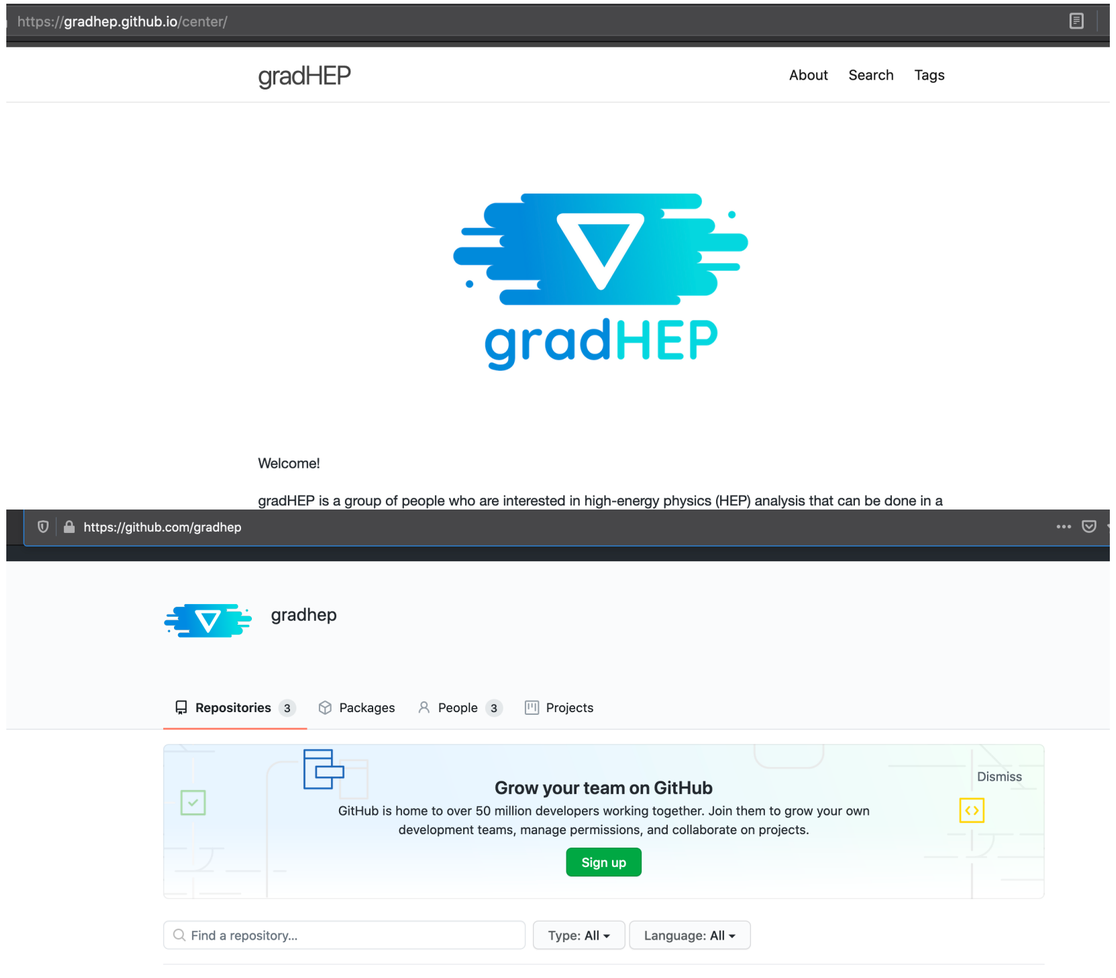


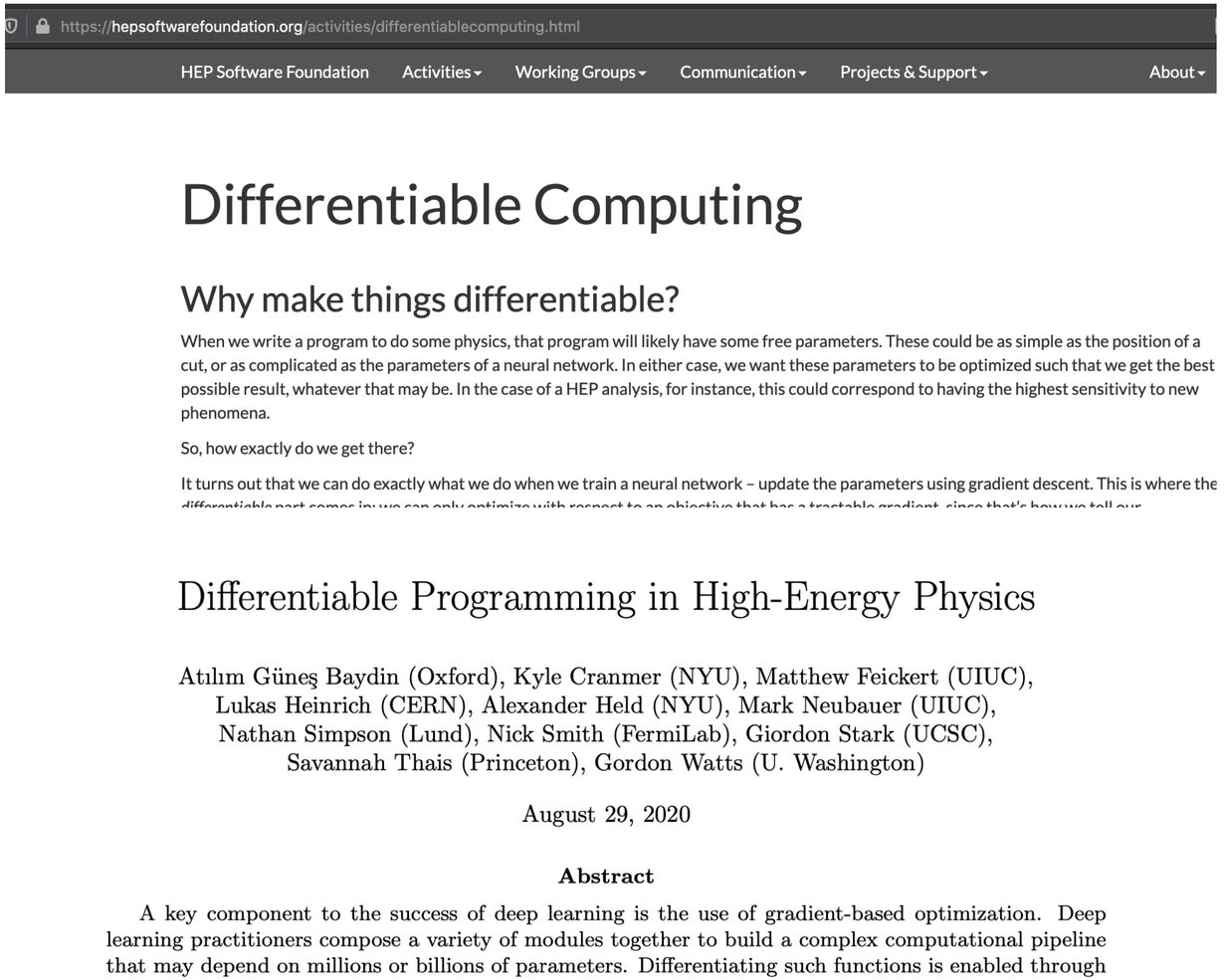
4) update weights

orange = background
either side of bkg = up/down variations
cyan = signal

The future?

A common group has formed around differentiable computing: gradHEP -- this is just the beginning :)





Differentiable Programming in High-Energy Physics

Atılım Güneş Baydin (Oxford), Kyle Cranmer (NYU), Matthew Feickert (UIUC),
Lukas Heinrich (CERN), Alexander Held (NYU), Mark Neubauer (UIUC),
Nathan Simpson (Lund), Nick Smith (FermiLab), Giordon Stark (UCSC),
Savannah Thais (Princeton), Gordon Watts (U. Washington)

August 29, 2020

Abstract

A key component to the success of deep learning is the use of gradient-based optimization. Deep learning practitioners compose a variety of modules together to build a complex computational pipeline that may depend on millions or billions of parameters. Differentiating such functions is enabled through a computational technique known as automatic differentiation. The success of deep learning has led to

thanks for listening :)