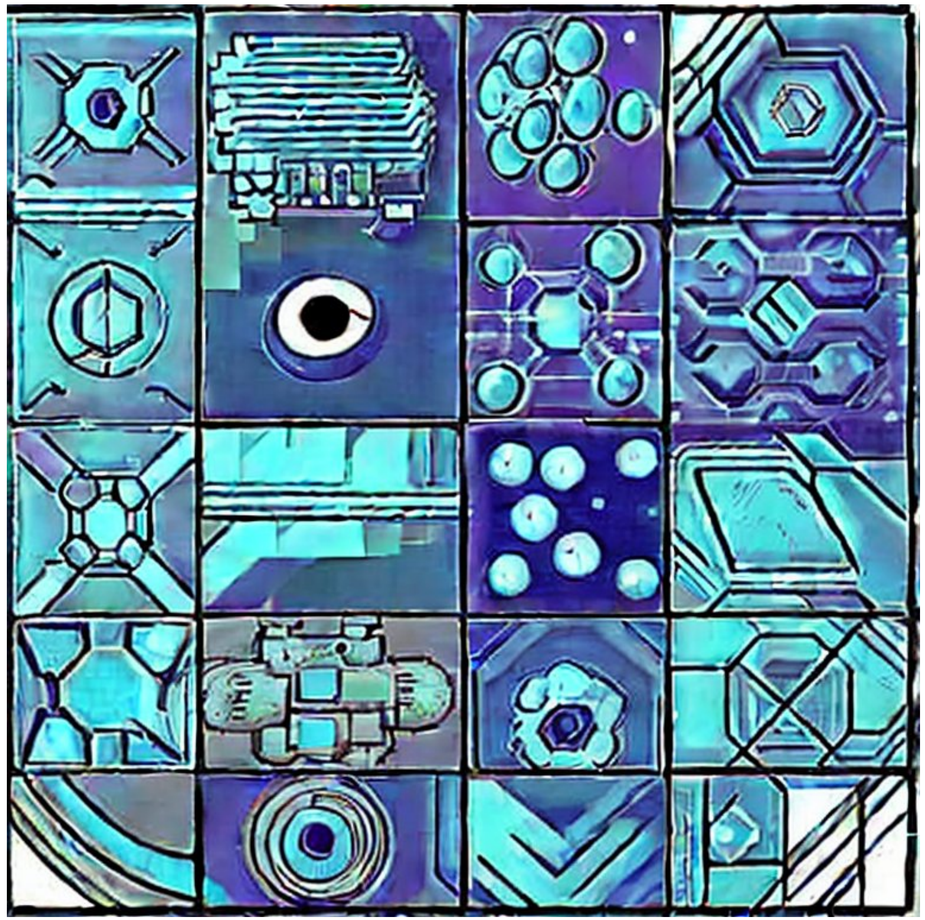


Machine learning for physicists

https://github.com/wangleiphy/ml4p

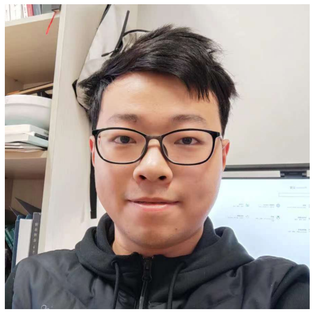
每周四上午10点

2.23	Overview
3.2	Machine learning practices
3.9	A hitchhiker’s guide to deep learning
3.16	Research projects hands-on
3.23	Symmetries in machine learning
3.30	Differentiable programming
4.6	Generative models-I
4.13	Generative models-II
4.20	Research projects presentation
4.27	AI for science: why now ?

课程微信群

助教

李子航

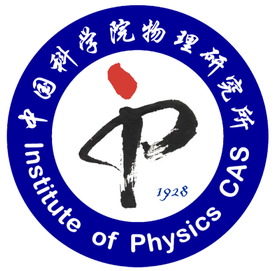


李扬帆

考核方式： project + presentation （1学分）

Machine learning practices

Lei Wang (王磊)
Institute of Physics, CAS
https://wangleiphy.github.io



Deep learning tools

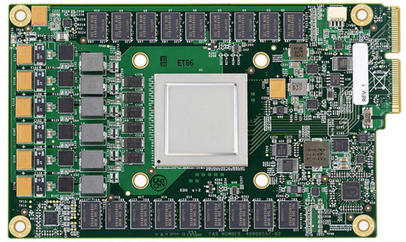
HIPS/autograd

Differentiation tensor operations on hardware accelerators

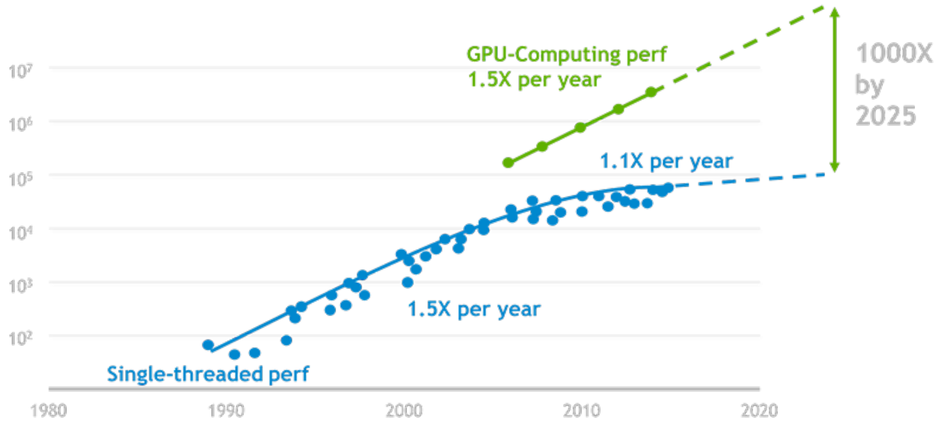
Hardware accelerators

for massive homogeneous computations (e.g. matrix multiplication)

GPU



TPU



The Hardware Lottery

Sara Hooker

Google Research, Brain Team
shooker@google.com

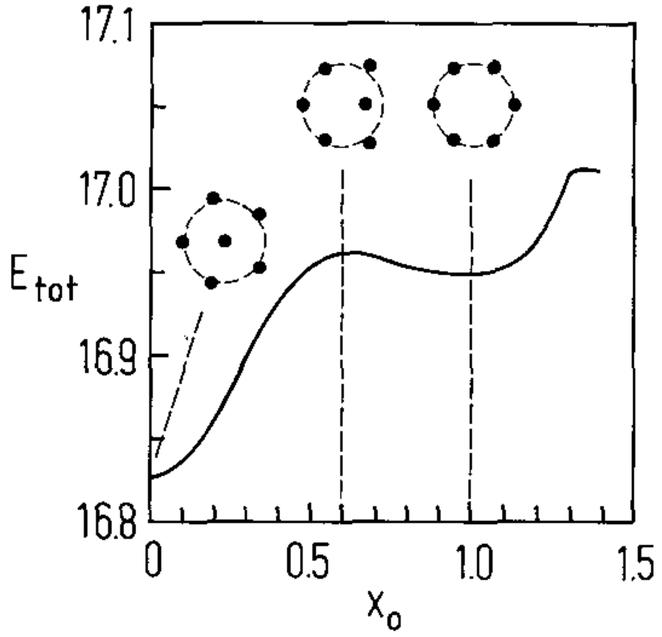
Yann LeCun @ylecun · Feb 22

Even hotter take: the fact that ML and Computer Vision researchers were largely using Matlab *held back progress for years*, mostly because implementing something like a ConvNet in 2005 Matlab would have been a total nightmare.

> 
>
> **Yann LeCun** @ylecun · Feb 22
>
> Hotter take: ML would have advanced faster if another front-end language had been available and widely adopted instead of Python. One that is interactive yet fast & compilable, multithreaded (no GIL), isn't bloated, doesn't care about white spaces,...
> E.g. Julia or some Lisp. twitter.com/tunguz/status/...

Coulomb gas

$$H = \sum_{i<j} \frac{1}{|\boldsymbol{x}_i - \boldsymbol{x}_j|} + \sum_{i}^{N} \boldsymbol{x}_i^2$$



Ground state

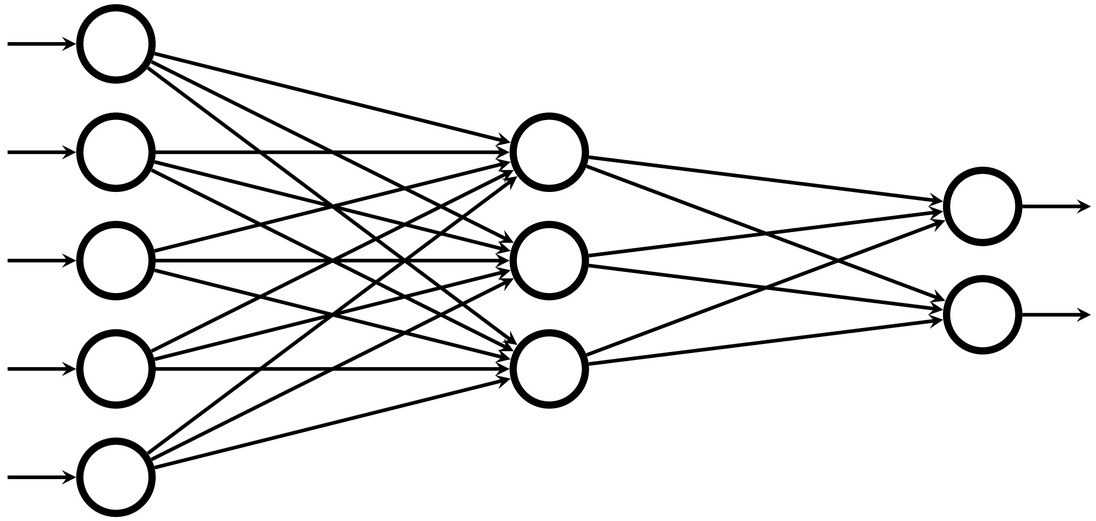
Optimization $\quad x^* = \underset{x}{\operatorname{argmin}}\, H(x)$

Finite temperature

Sampling $\quad x \sim \dfrac{e^{-\beta H(x)}}{Z}$

https://github.com/wangleiphy/ml4p/blob/main/materials/2_handson/jax.ipynb

MNIST



digits

https://github.com/wangleiphy/ml4p/blob/main/materials/2_handson/mnist.ipynb

Machine learning practices

Testing there is no excuse for not testing your code

- test invariances
- gradient test
- be aware of sum/mean/multiplication over various dimensions
- over-fit a small dataset

Tuning https://github.com/google-research/tuning_playbook

- parameter initialization
- gradient clipping

Logging

- checkpoint
- metadata