

# プラズマ・核融合分野における データ駆動研究の現状

2019.12.1

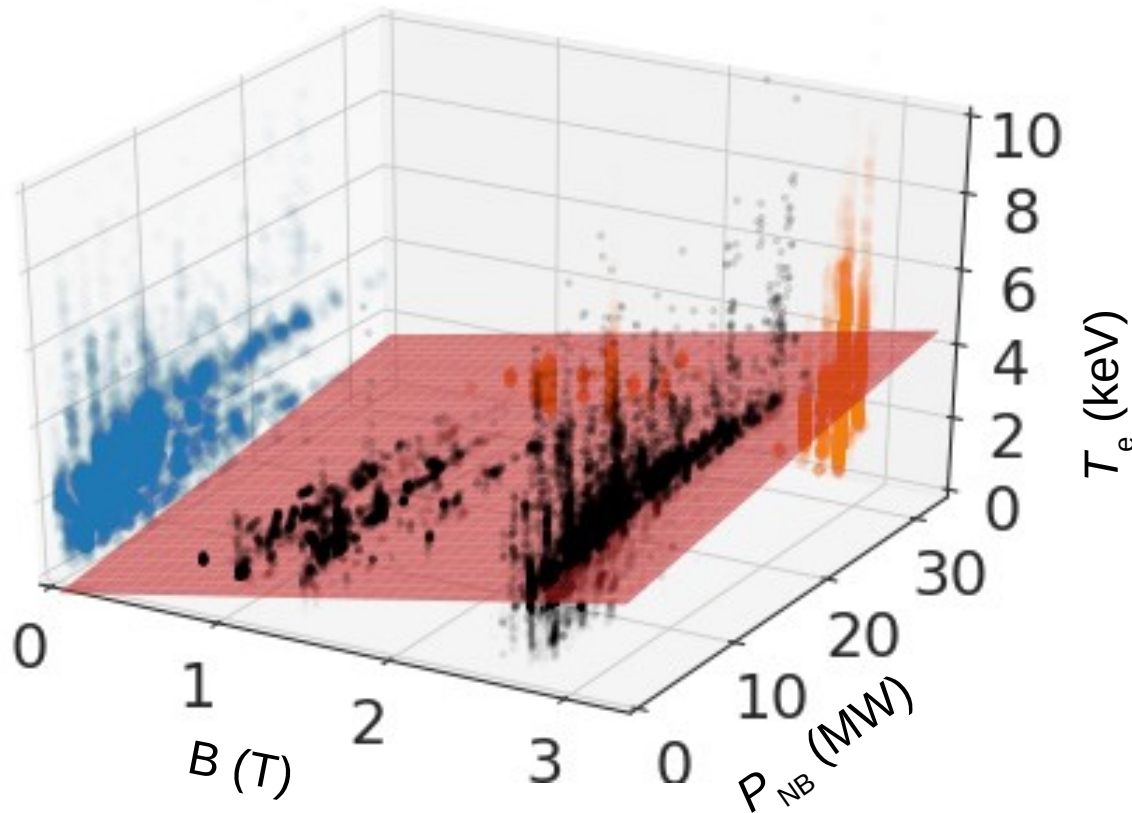


京都大学 工学研究科  
藤井恵介

プラ核 - 計算科学研究部会

# データ科学の基本：最小二乗近似

$$L = \sum_i^N \frac{1}{2} |y_i - (a_1 x_{1,i} + a_2 x_{2,i} + b)|^2$$



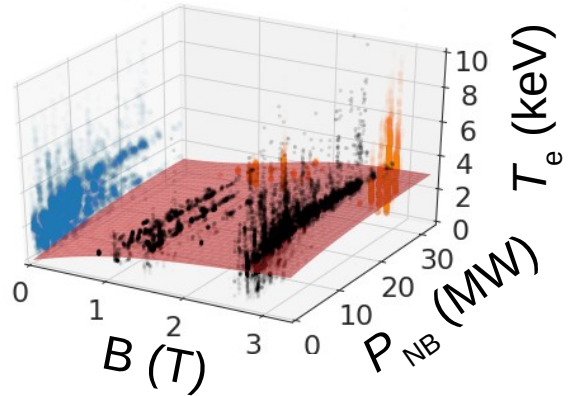
$$L = \frac{1}{2} |\mathbf{y} - \mathbf{X}\mathbf{a}|^2$$

$$\frac{dL}{d\mathbf{a}} = 0$$

$$\mathbf{a} \leftarrow \mathbf{a} - \eta \frac{dL}{d\mathbf{a}}$$

# 多次元最小二乗法 (非線形)

# ニューラルネットワーク



線形

$$L = \frac{1}{2} |\mathbf{y} - \mathbf{X}\mathbf{a}|^2$$

非線形

$$L = \frac{1}{2} |\mathbf{y} - f(\mathbf{X}; \theta)|^2$$

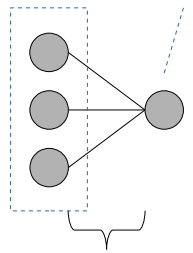
$$f(\mathbf{X}; \theta) = h(\mathbf{X}\mathbf{a})$$

$$f(\mathbf{X}; \theta) = h(\mathbf{X}\mathbf{A}_1)\mathbf{a}$$

$$f(\mathbf{X}; \theta) = h(h(\mathbf{X}\mathbf{A}_1)\mathbf{A}_2)\mathbf{a}$$

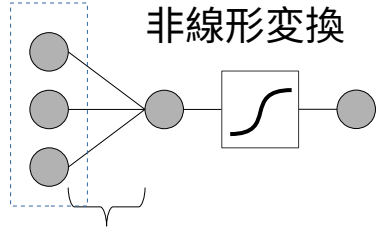
表現力: 大  
パラメータ: 多

入力 出力

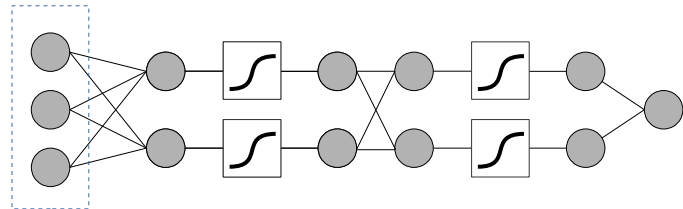
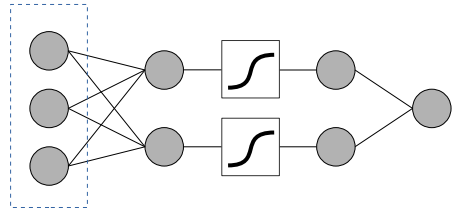


線形変換

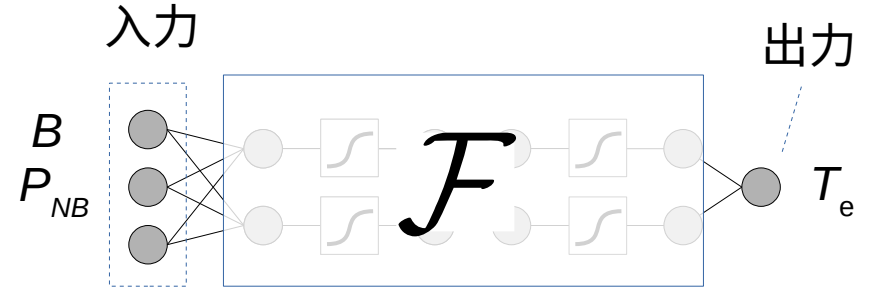
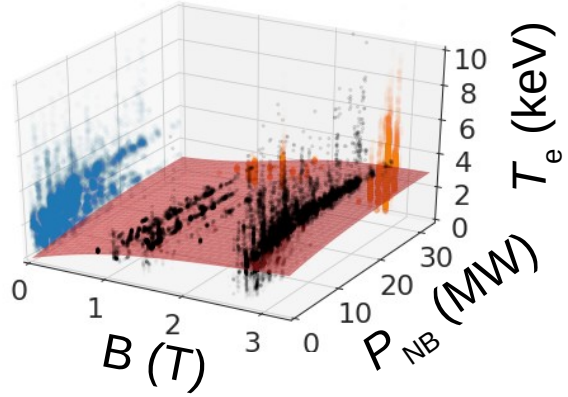
要素ごとの  
非線形変換



線形変換

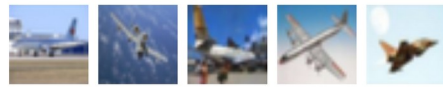


# ニューラルネットワーク



予測タスク：入力と出力のペアから、その関係を近似する

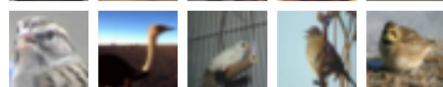
airplane



automobile



bird

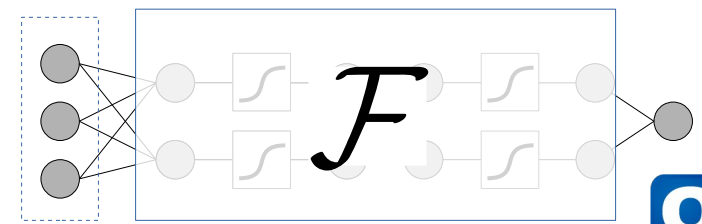


cat



ネコ!

128x128 pixels  $\rightarrow 10^4$

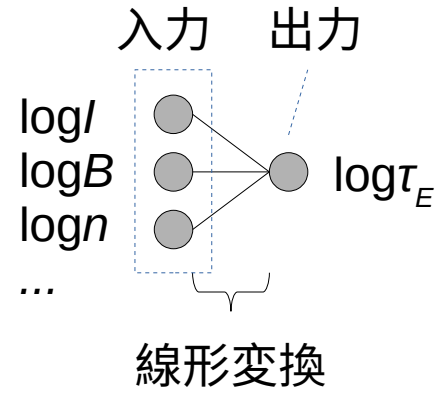
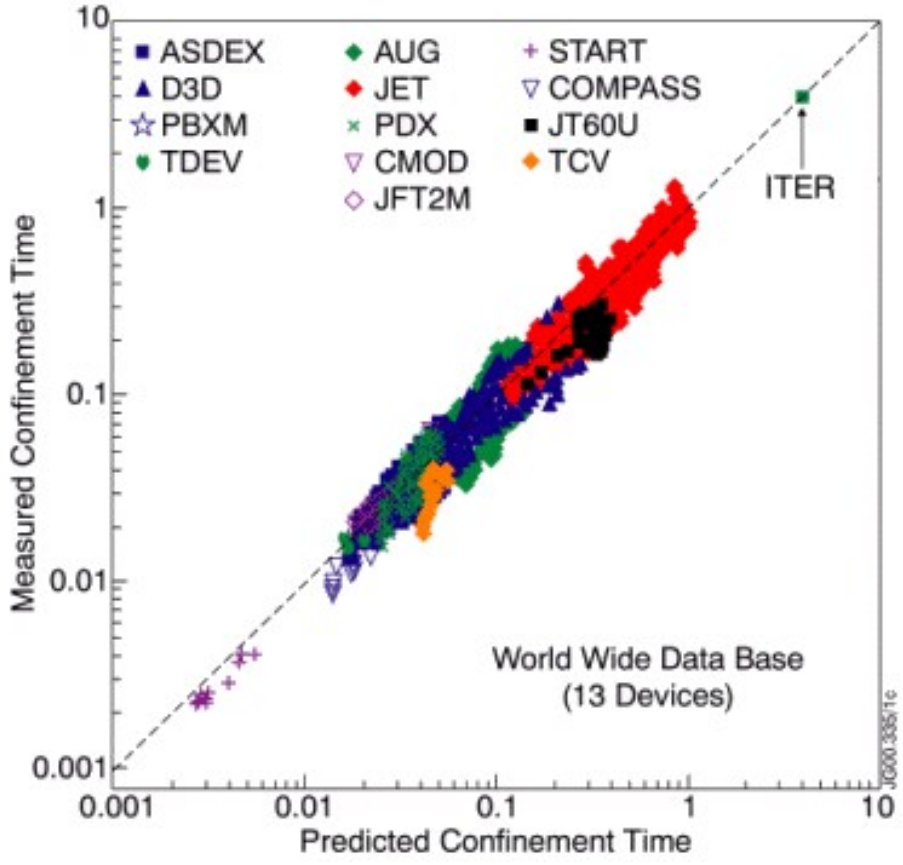


<https://alexisbcook.github.io/2017/using-transfer-learning-to-classify-images-with-keras/>

# プラズマ・核融合分野での応用（1）

- 目標達成のために必要なパラメータを決定する（補外）

$$\tau_E = C I^{\alpha_I} B^{\alpha_B} \bar{n}^{\alpha_n} P^{\alpha_P} R^{\alpha_R} \kappa^{\alpha_\kappa} \epsilon^{\alpha_\epsilon} S_{cr}^{\alpha_S} M^{\alpha_M}$$



# プラズマ・核融合分野での応用 (2)

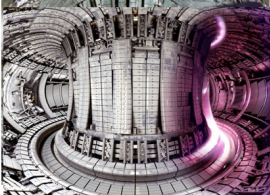
複雑なシミュレーションを高速化する

Citrin, J., et al (2015). Nuclear Fusion, 55(9), 092001.

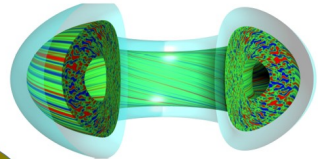
Narita, E., Honda, M., et al. (2019). Nuclear Fusion, 59(10)

Parameter	Min value	Max value	No. of points
$R/L_{Ti}$	2	12	30
$T_i/T_e$	0.3	3	20
$q$	1	5	20
$\hat{s}$	0.1	3	20
$k_{\theta\rho_s}$	0.05	0.8	16
Total no. of points			3 840 000

1 second of plasma in...



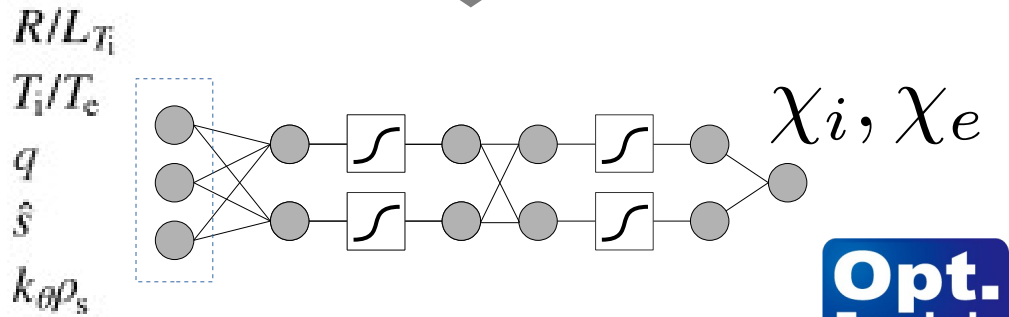
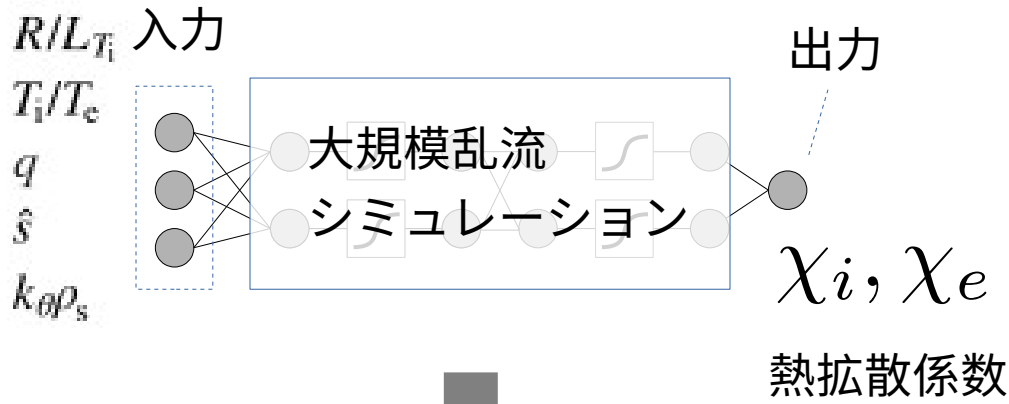
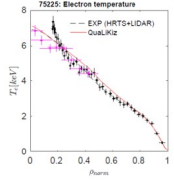
1 month on 10,000 cores



1 second on 1 core!



1 day on 24 cores



# プラズマ・核融合分野での応用（3）

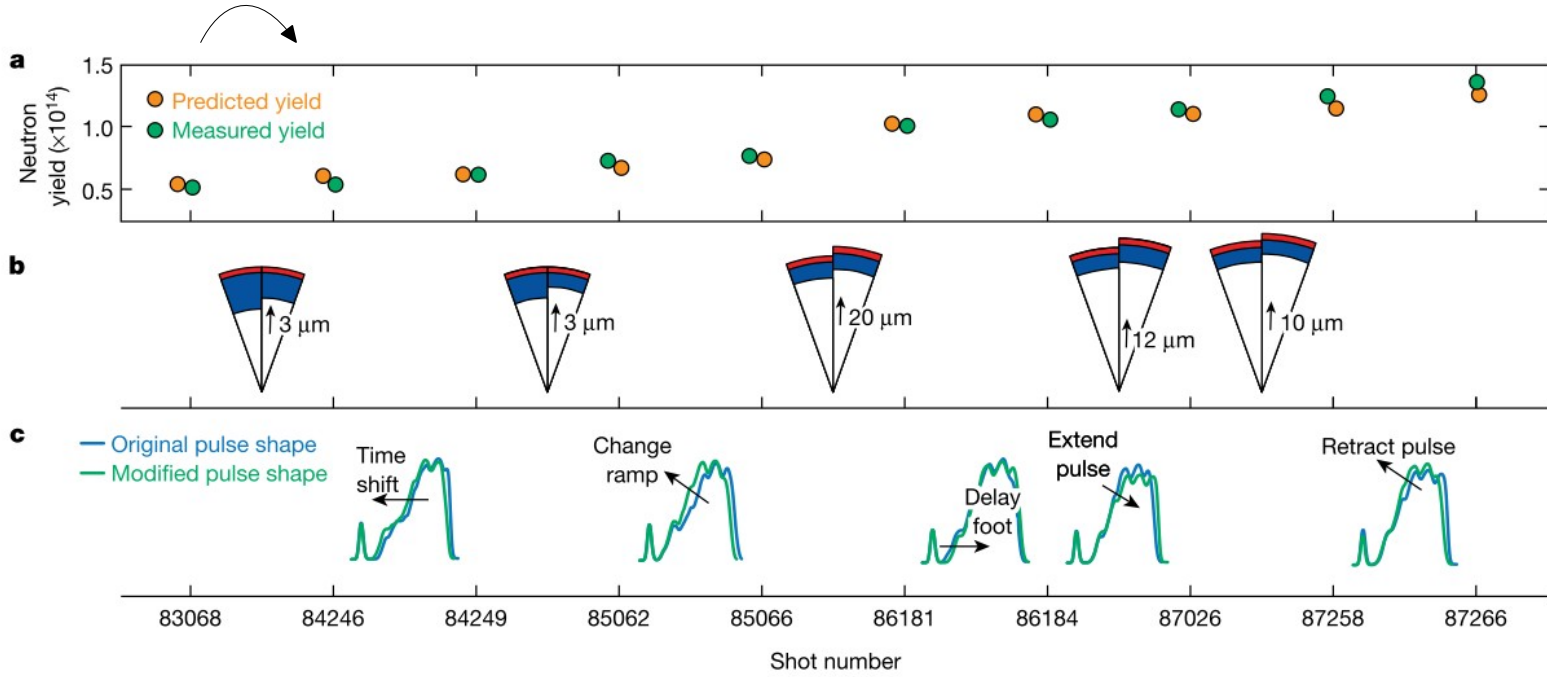
高性能プラズマを達成する

Gopaldaswamy, V., et al. (2019). Nature, 565(7741), 581–586.  
<https://doi.org/10.1038/s41586-019-0877-0>

Baltz, E. A., et. al. (2017). Scientific Reports, 7(1), 6425.  
<https://doi.org/10.1038/s41598-017-06645-7>

モデルの学習・最適パラメータの予測・実験データ取得を繰り返して  
モデルの精度をあげながら最高性能を達成していく

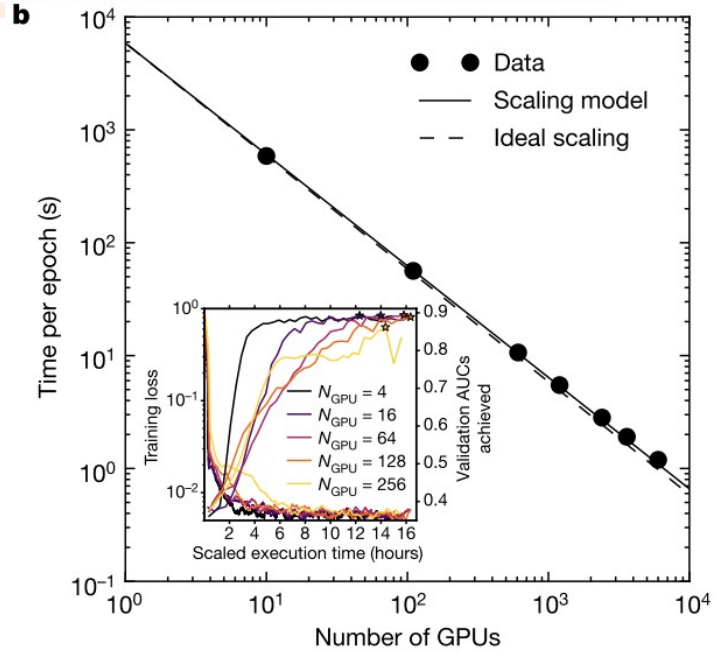
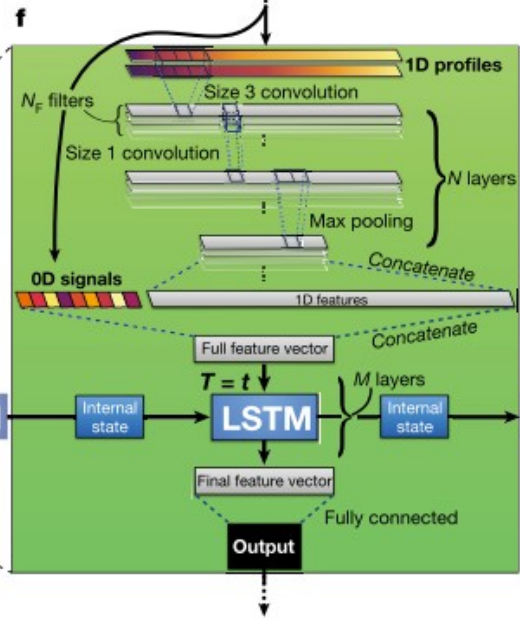
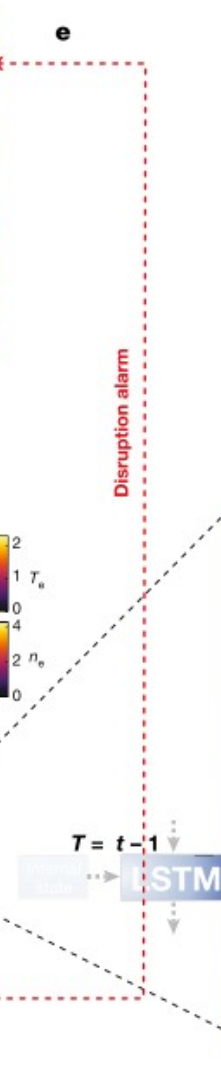
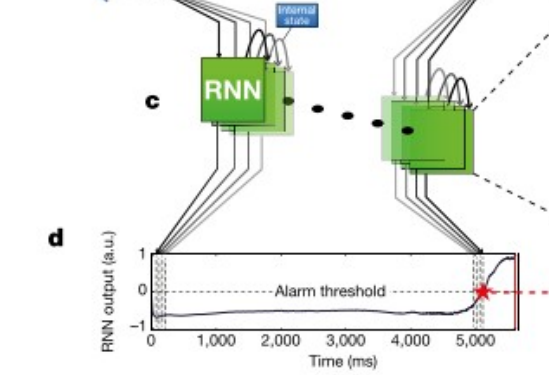
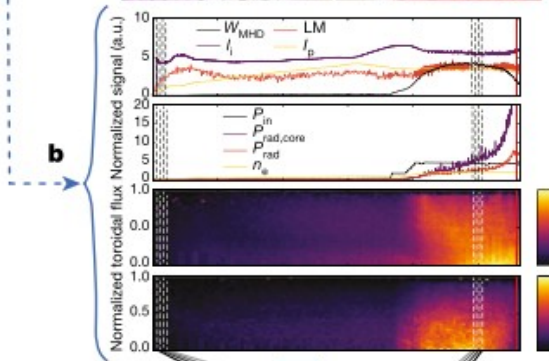
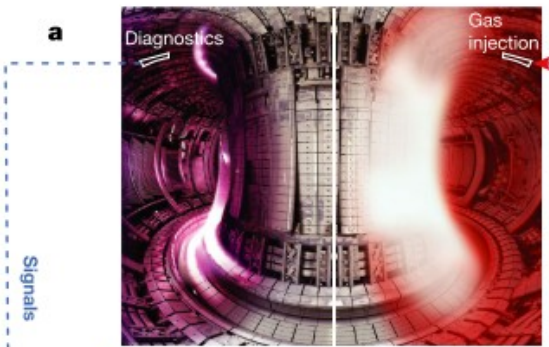
学習・最適化





# ディストラクション予測

Kates-Harbeck, J., Svyatkovskiy, A., & Tang, W. (2019). Nature, 568(7753), 526–531

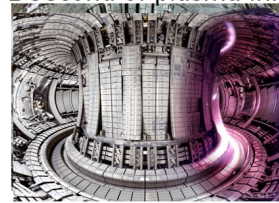




# データ駆動科学でできるようになってきたこと

## シミュレーションの高速化

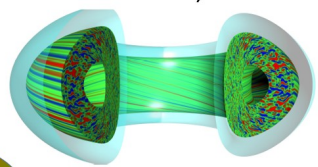
1 second of plasma in..



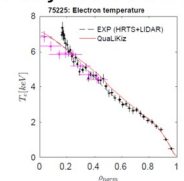
1 second on 1 core!



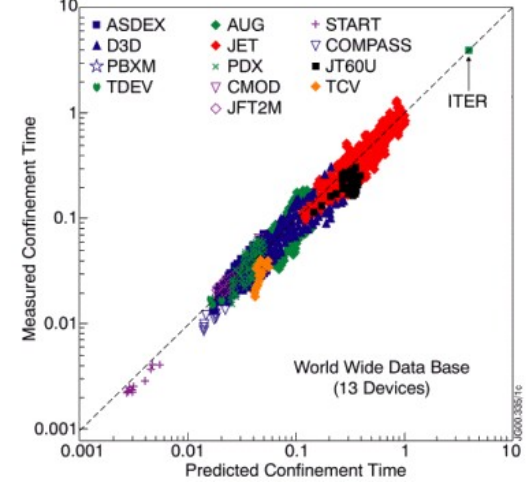
1 month on 10,000 cores



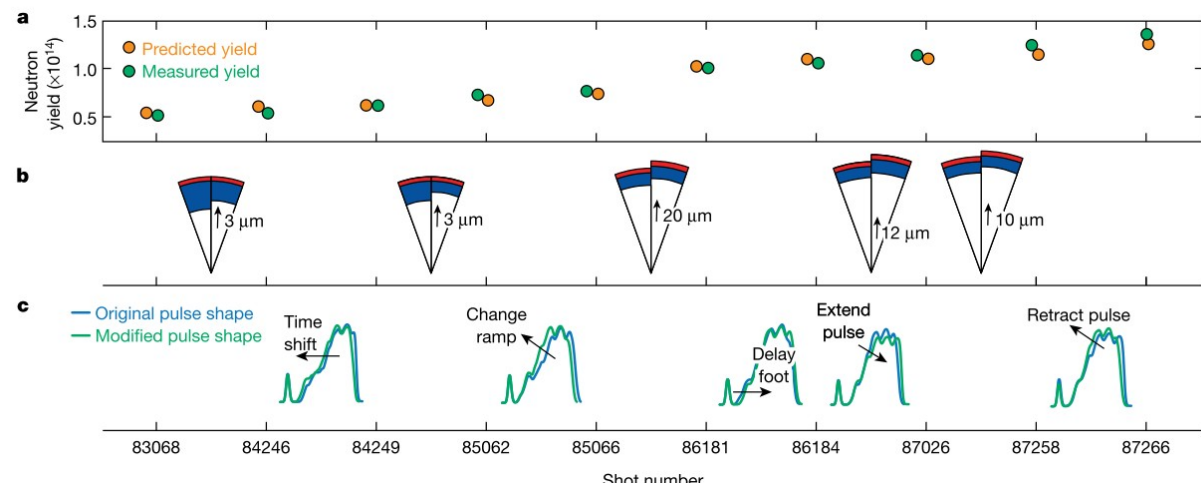
1 day on 24 cores



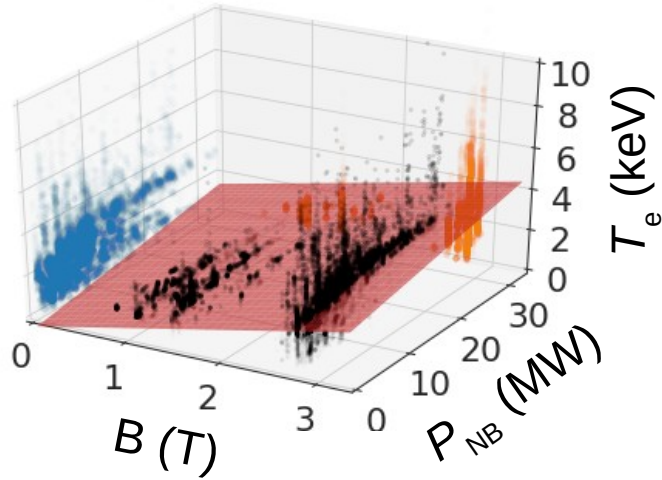
## 装置設計・運転のための予測



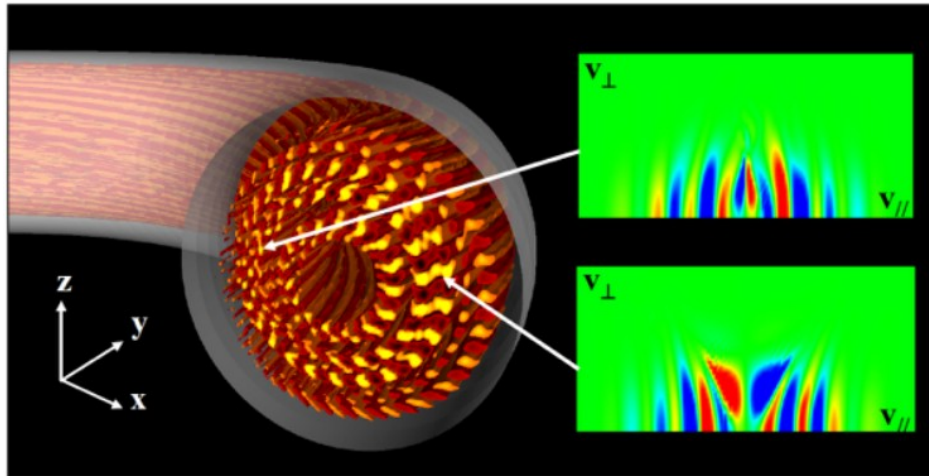
## 学習と実験の繰り返しによる最適パラメータの探索



# データ駆動科学でできるようになってほしいこと 大規模データからのパターンの発見



低次元データ：  
相関・パターンはだいたいわかる

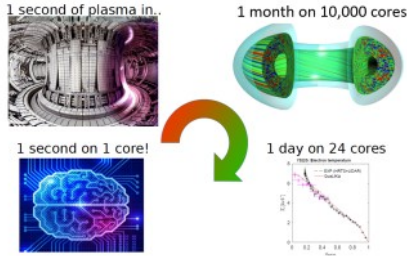


高次元データ  
例：乱流シミュレーション  
 $10^{13}$  データポイント  $\approx$  40TB

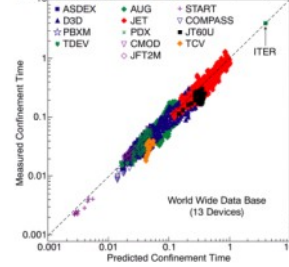
データがそもそもどんな様子なのかもよくわからない

# データ駆動科学でできるようになってきたこと

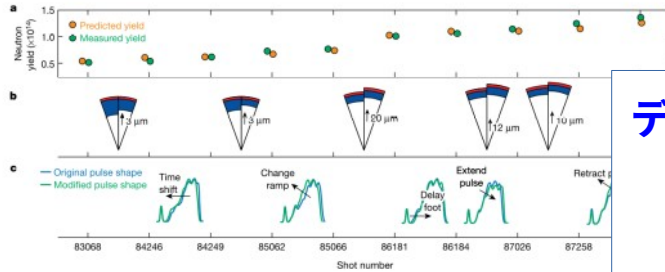
## シミュレーションの高速化



## 装置設計・運転のための予測

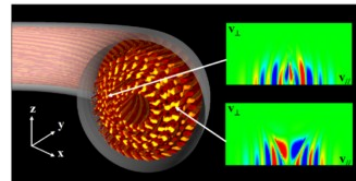


## 学習と実験の繰り返しによる最適パラメータの探索

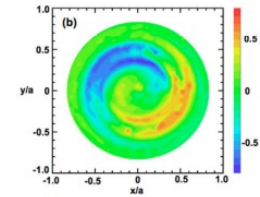


## 高次元データからの知見の抽出

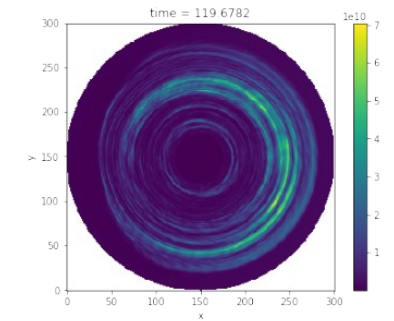
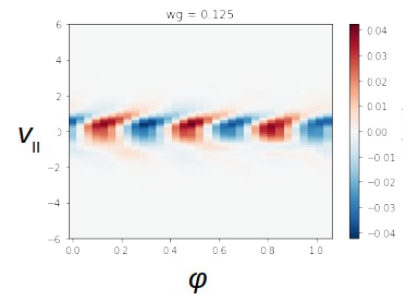
# データ駆動科学でできるようになってほしいこと



ジャイロ運動論 5次元シミュレーションから、  
径方向に伝播する波を抽出できた…?



Inagaki, S., Tamura, N., Tokuzawa, T., Ida, K., Kobayashi, T., Shimozuma, T., ... Itoh, S. I. (2012). Nuclear Fusion, 52(2)



# AI の発展がもたらした事実上のメリット

自動でマルチノード計算

```
import numpy as np
```

```
X = np.random.randn(1000, 3333)  
y = np.random.randn(3333, 222)  
X @ y
```

```
import dask.array as da  
from dask.distributed import Client  
from dask_jobqueue import SLURMCluster  
  
if __name__ == '__main__':  
    frame_start = int(sys.argv[1])  
    frame_end = int(sys.argv[2])  
  
    print("Preparing workers.")  
  
    nb_workers = 6  
    cluster = SLURMCluster(processes=1,  
                           cores=10,  
                           memory="48GB",  
                           project='GT5DSTC',  
                           name='worker',  
                           queue='S-M',  
                           walltime='24:00:00',  
                           interface='ipogif0',  
                           job_extra={'-o dask%j.out', '-e dask%j.out'},  
                           dashboard_address=':1234')
```

```
X = da.random.randn(1000, 3333)  
y = da.random.randn(3333, 222)  
X @ y
```