

機械学習を用いた飛跡画像処理による 光電子追跡型 X 線偏光計の感度向上

(TK+19, NIMA or arXiv:1907.06442)

内容

1. 天体 X 線偏光観測の現状
2. 機械学習による飛跡画像処理および偏光情報抽出

北口 貴雄¹

玉川 徹¹, 岩切 渉², 榎戸 輝揚³, 水野 恒史⁴, Niccolò Di Lalla⁵

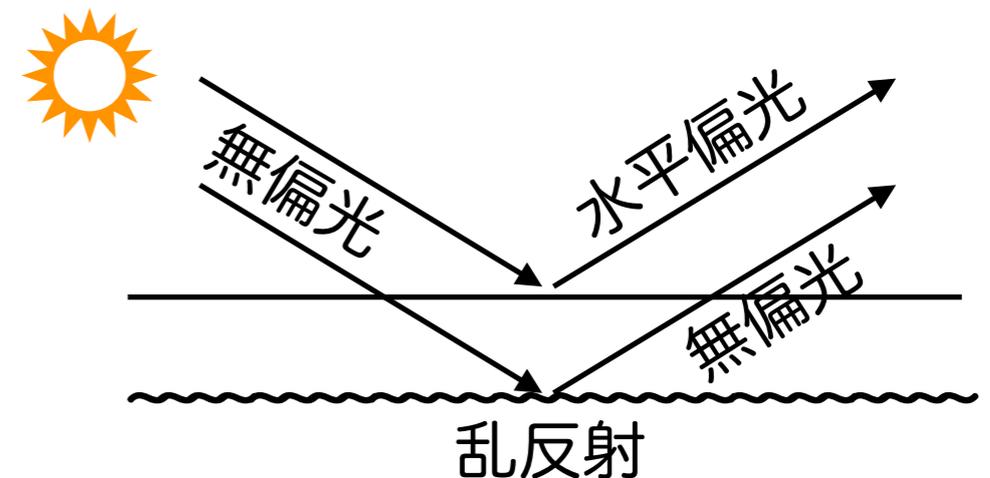
1. 理研, 2. 中央大, 3. 京都大, 4. 広島大, 5. INFN-Pisa

2019年12月7日

1. 天体 X 線偏光観測の現状

天体 X 線偏光観測の意義

- X 線偏光による天体観測はほぼ未開拓
 - 70 年代に、かに星雲から 1 検出 (Weisskopf+76)
 - それ以降、偏光計を搭載する衛星なし
- 可視光での例 — 水中をクリアに見透す
 - 水面で散乱した光を、偏光フィルタでカット
 - 逆に水平偏光を抽出して、水面の状態を探查可
- 天体への応用例
 - 強磁化プラズマは天然の偏光フィルタ
 - 中性子星表面からの放射は、100% 偏光?
 - BH の強重力場および回転により偏光
 - BH 回転の測定
 - 超新星残骸の衝撃波での磁場整列度
 - 粒子加速モデルの制限



光電効果を使った偏光測定

- 光電子追跡型 偏光計

- X線と物質の反応は、光電効果が支配的
- K殻光電子の射出方位角 ϕ は、X線の偏光角に依存

$$\frac{d\sigma}{d\Omega} = 4\sqrt{2}r_0\alpha^4Z^5 \left(\frac{m_e c^2}{h\nu}\right)^{7/2} \frac{\sin^2\theta}{(1-\beta\cos\theta)^4} \cos^2\phi$$

$\propto \cos^2\phi$ (散乱角 θ に依存しない)

- X線入射方向に射影した2次元追跡でよい

- 偏光検出効率: $\mu = \text{サインカーブの高さ} / \text{オフセット}$

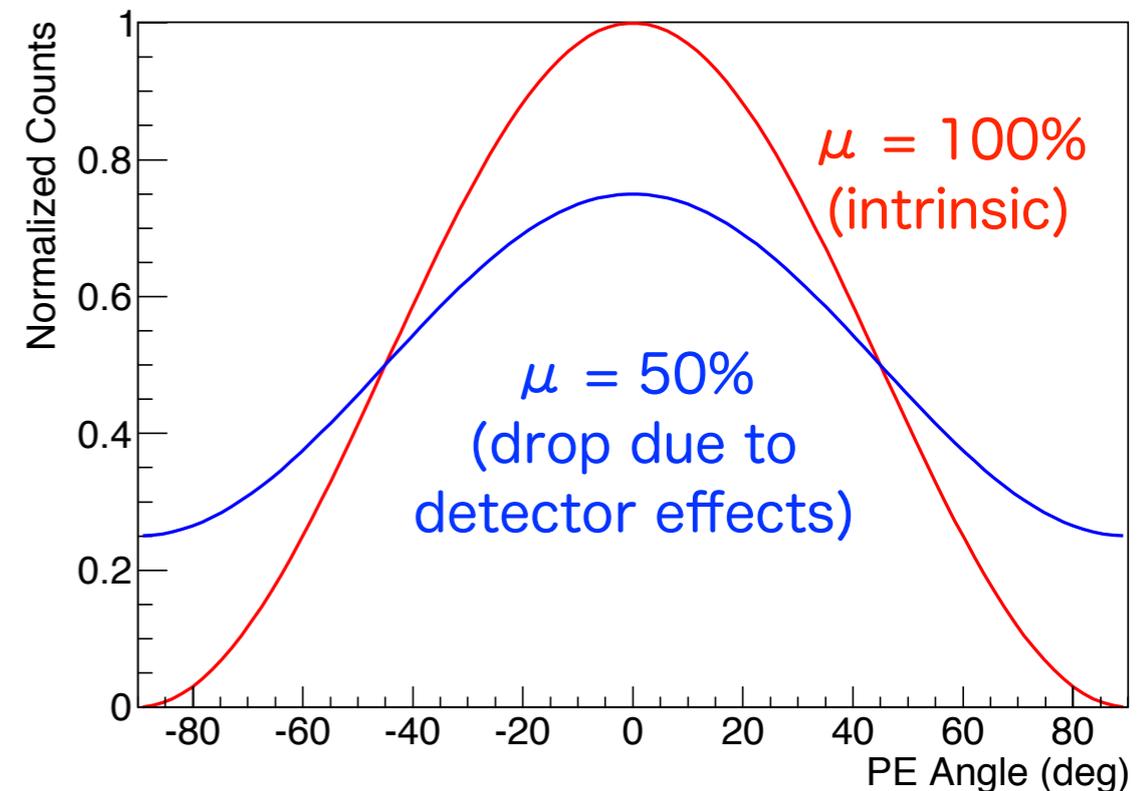
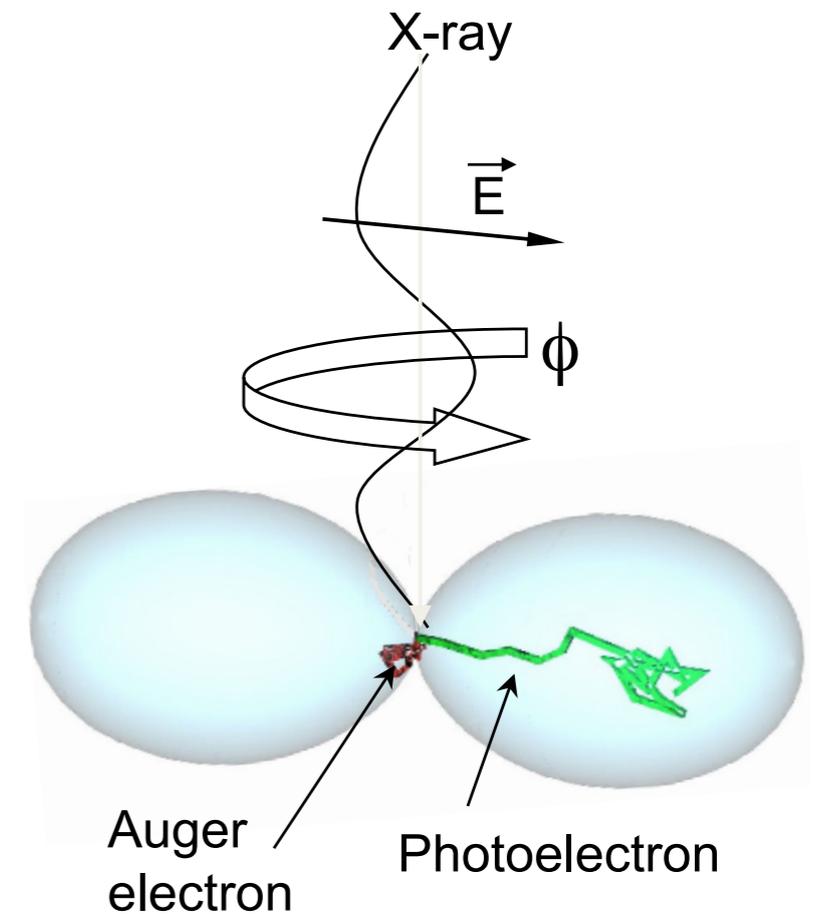
- intrinsic $\mu = 100\%$

- 最小検出偏光度 (MDP)

- $$\text{MDP} = \frac{4.29}{\mu\sqrt{N}} \quad (\text{N: 光子数})$$

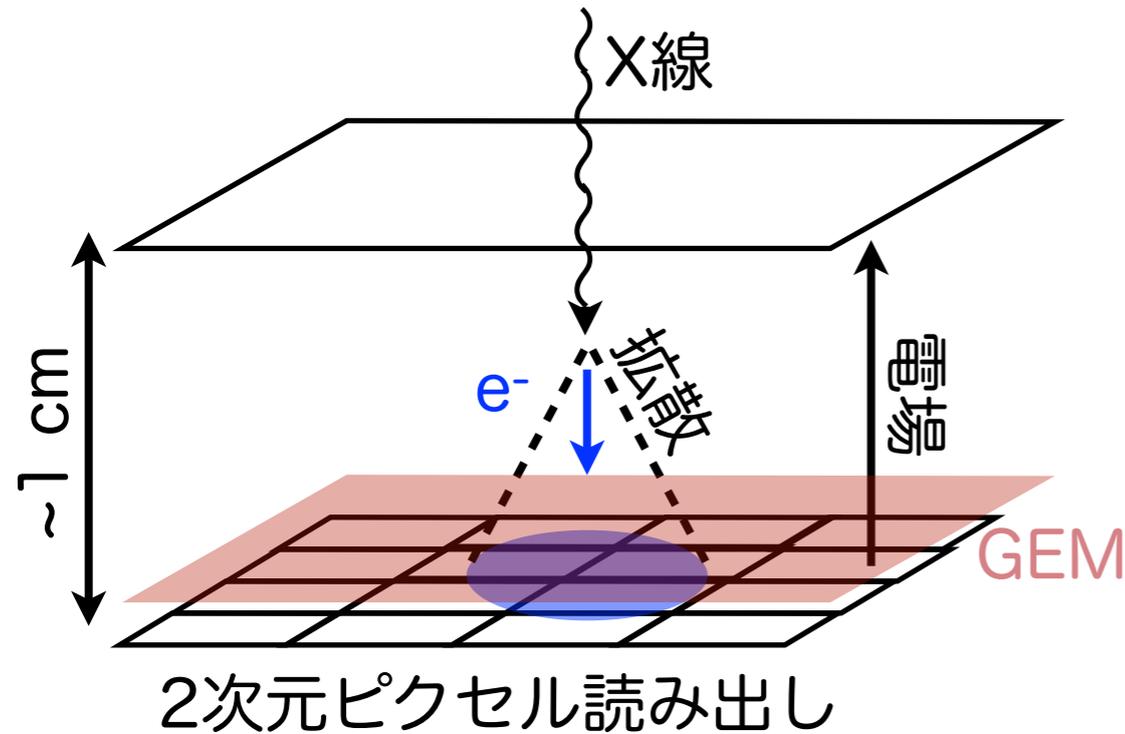
- μ を上げれば、比例して感度は向上

- ▶ 有効面積、観測時間の増加では、その平方根でしか感度は上がらない
- ▶ 飛跡画像から精確に ϕ を求めるアルゴリズムが必須



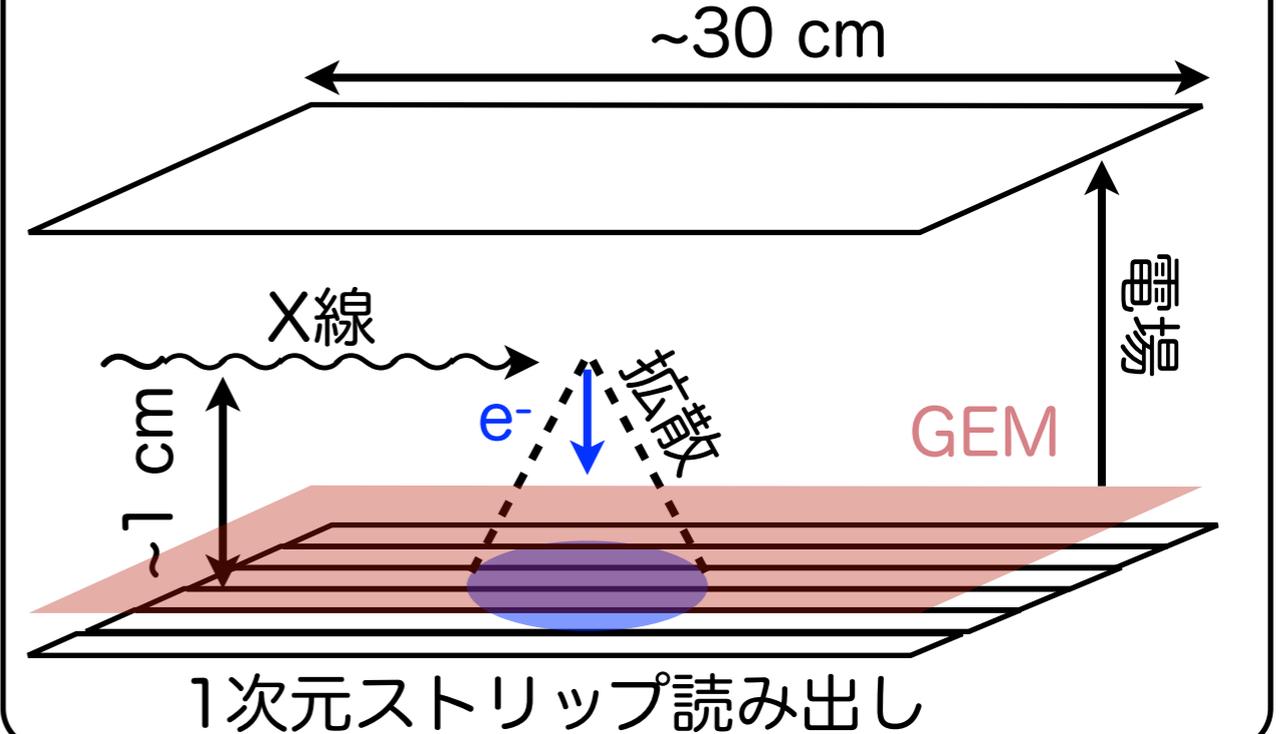
2つの光電子追跡型 X線偏光計

天体撮像偏光計



- 電子ドリフトは入射X線と平行
- 2次元読み出しで飛跡画像を撮像
- 薄い (~1 cm) 有感層 (低検出効率)
 - 電子拡散を抑えるため
- NASA/MSFC および INFN-Pisa が提案する衛星計画 IXPE (Imaging X-ray Polarimetry Explorer)

天体非撮像偏光計

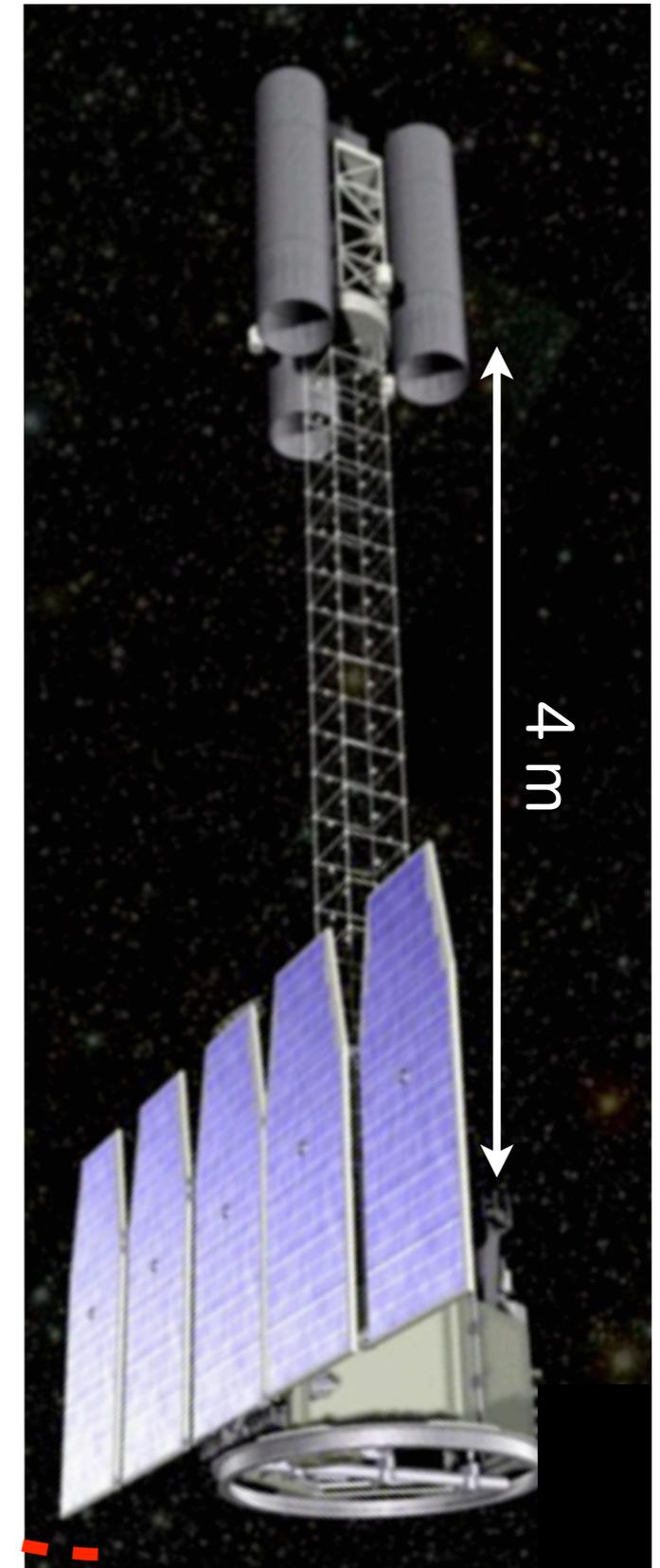
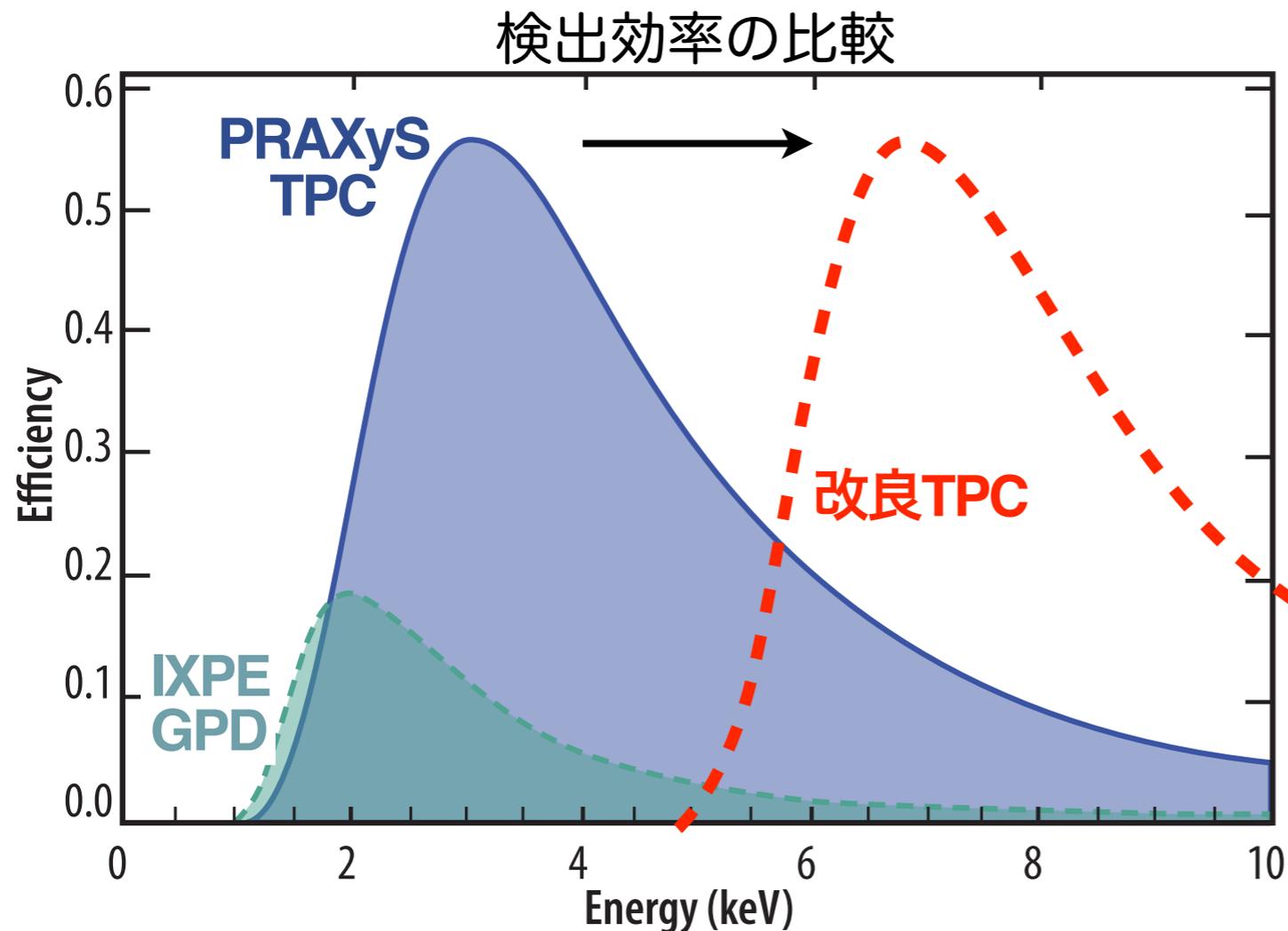


- 電子ドリフトは入射X線と垂直
- 1次元読み出しと TPC 技術で撮像
- 厚い (~30 cm) 有感層 (高検出効率)
 - 電子拡散は検出深さに依らず ~1cm
- NASA/GSFC および理研が提案する衛星計画 PRAXyS (Polarimeter for Relativistic Astrophysical X-ray Sources)

どちらも DME ガス封入、理研/SciEnergy 製 GEM 採用
NASA は 2016 年に2つのX線偏光計を概念検討に選択

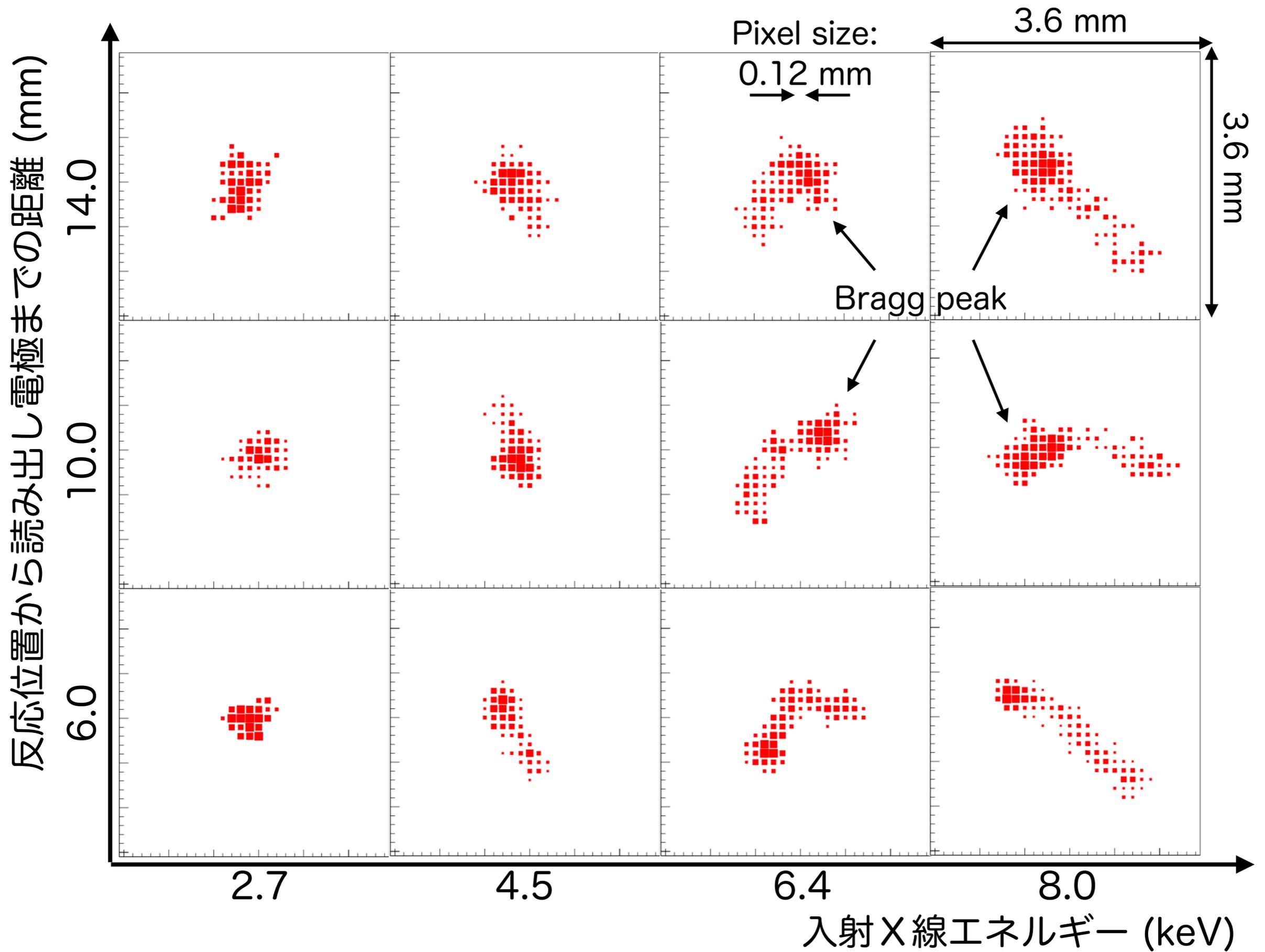
IXPE 当選、PRAXyS 落選

- NASA は 2017 年始めに IXPE を次期打上衛星に選定
 - 衛星搭載品はほぼ完成、較正試験を継続中
 - 2021 年に打ち上げ予定
- 日本の X 線偏光グループは IXPE に合流
 - 理研は GEM を提供 (製作および送付済)
 - 偏光データ解析アルゴリズムを提供
- TPC 偏光計
 - 高エネルギー X 線に感度拡張へ (武田講演)

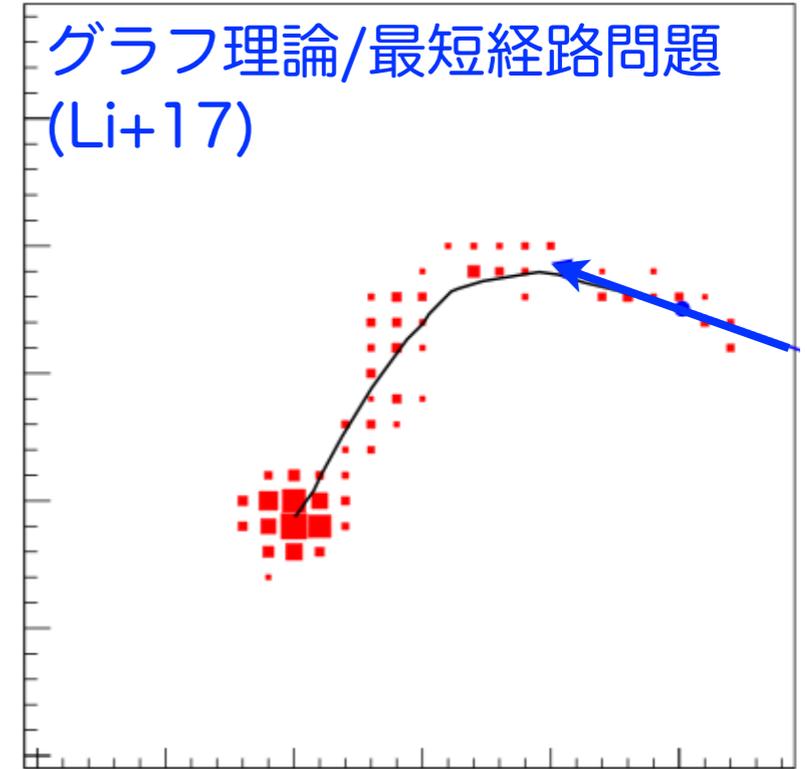
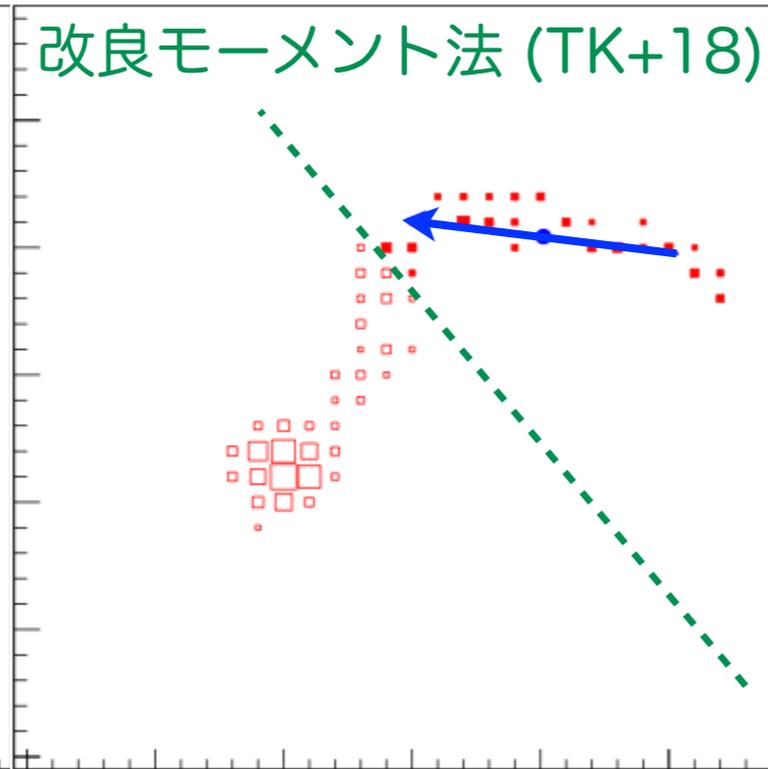
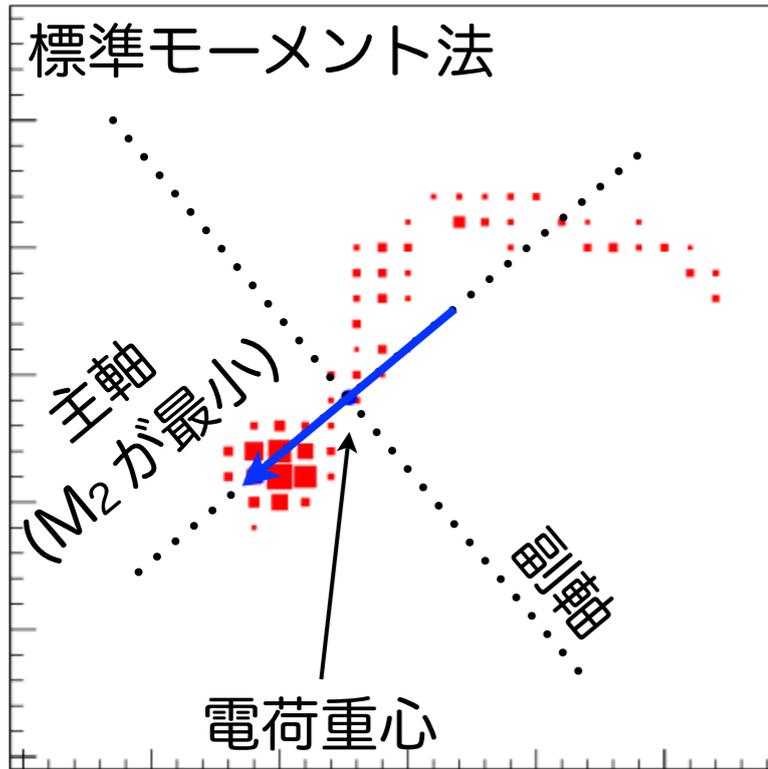


2. 機械学習を用いた光電子飛跡画像処理 (TPC 偏光計で取得した画像を基に)

光電子飛跡の実測例



これまでの角度再構成法



• 2つの角度再構成法

- モーメント法

- ▶ 2次モーメントを最小にする主軸の角度
- ▶ 曲がった飛跡部分をカット

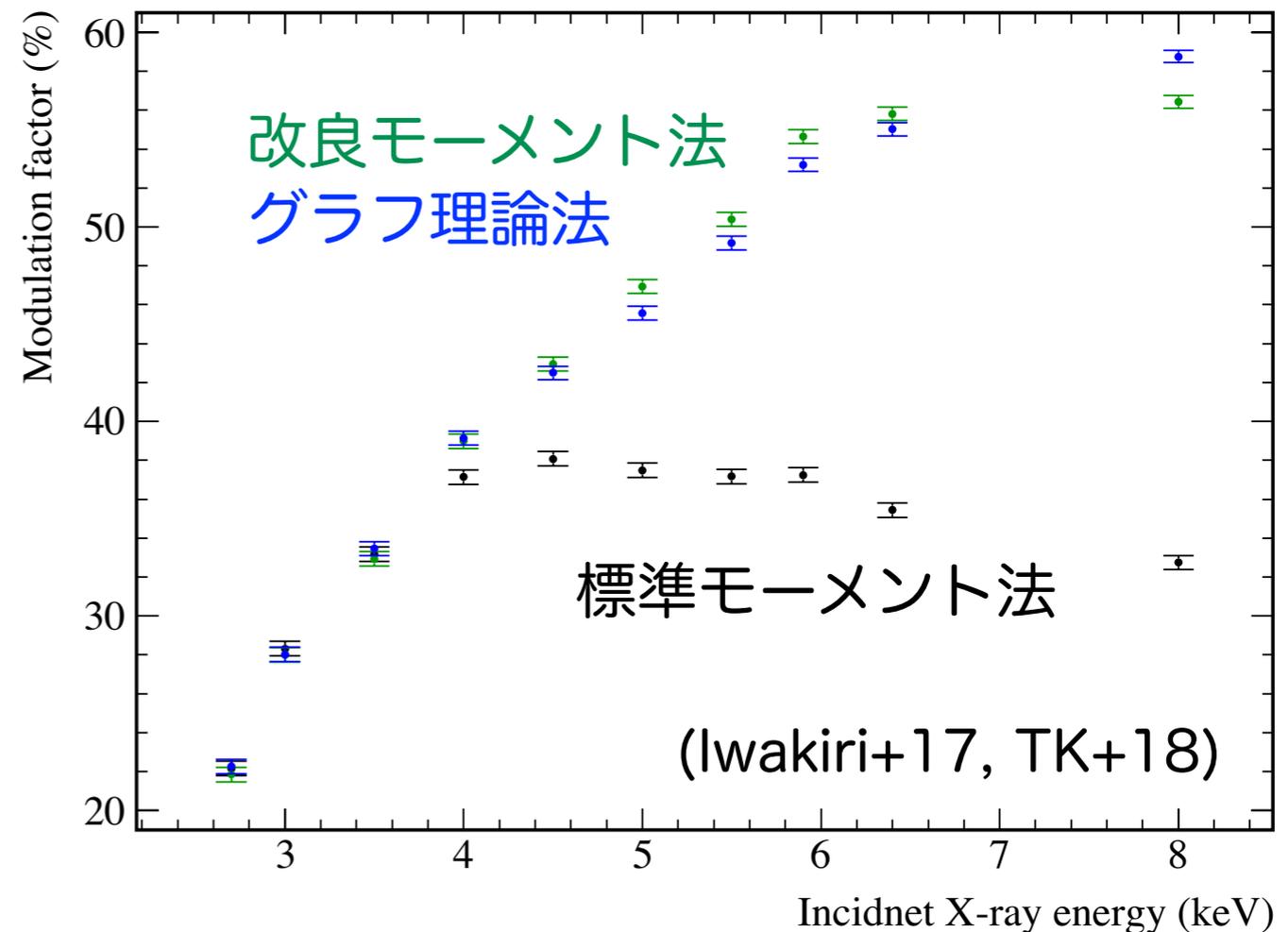
- グラフ理論法

- ▶ 2点の最短距離が最長になる経路、周辺電荷分布の重心でスムージング
- ▶ 始点からの距離を重みにして、モーメント法で角度導出

- 結局モーメントを使うため、同じ μ

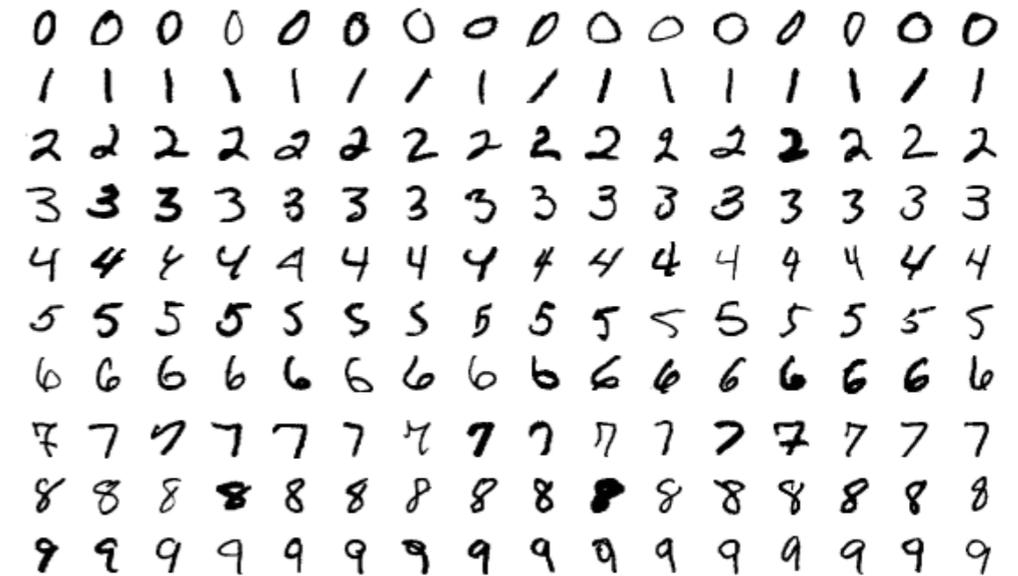
• 第3の方法: 機械学習を試行

- 画像認識が得意

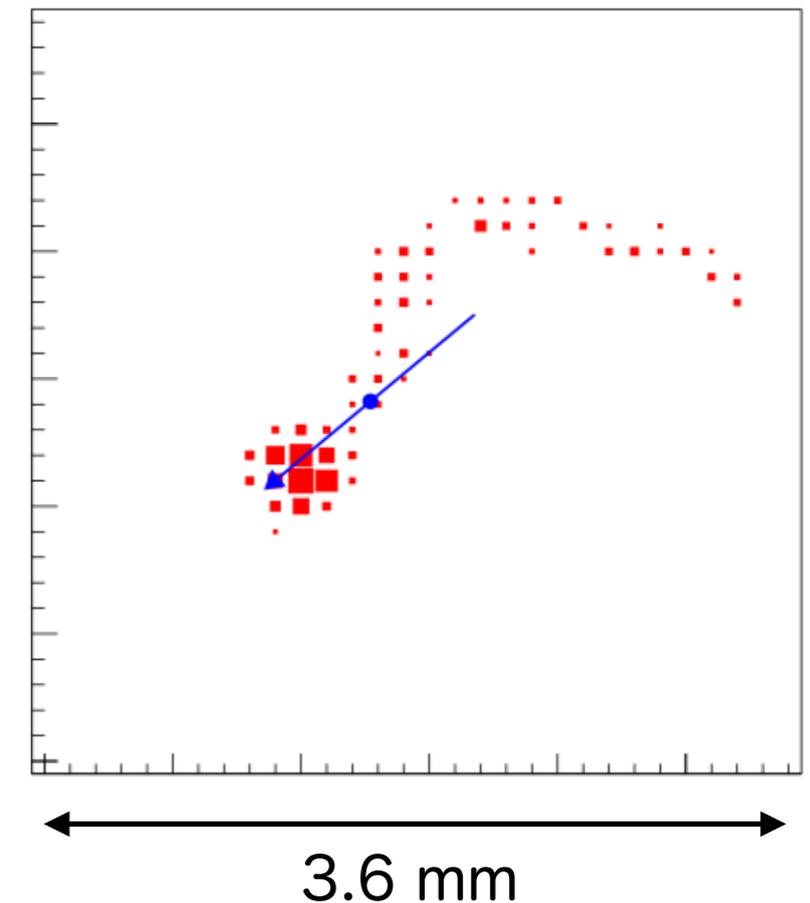


MNIST ベンチマークからの思いつき

- MNIST: 手書き数字データベース
 - 機械学習のベンチマークとして有名
 - ▶ 様々なサンプルコードが利用可能
 - 入力画像: 28x28 画素, 60,000 枚
 - 出力: 数字 (0-9)

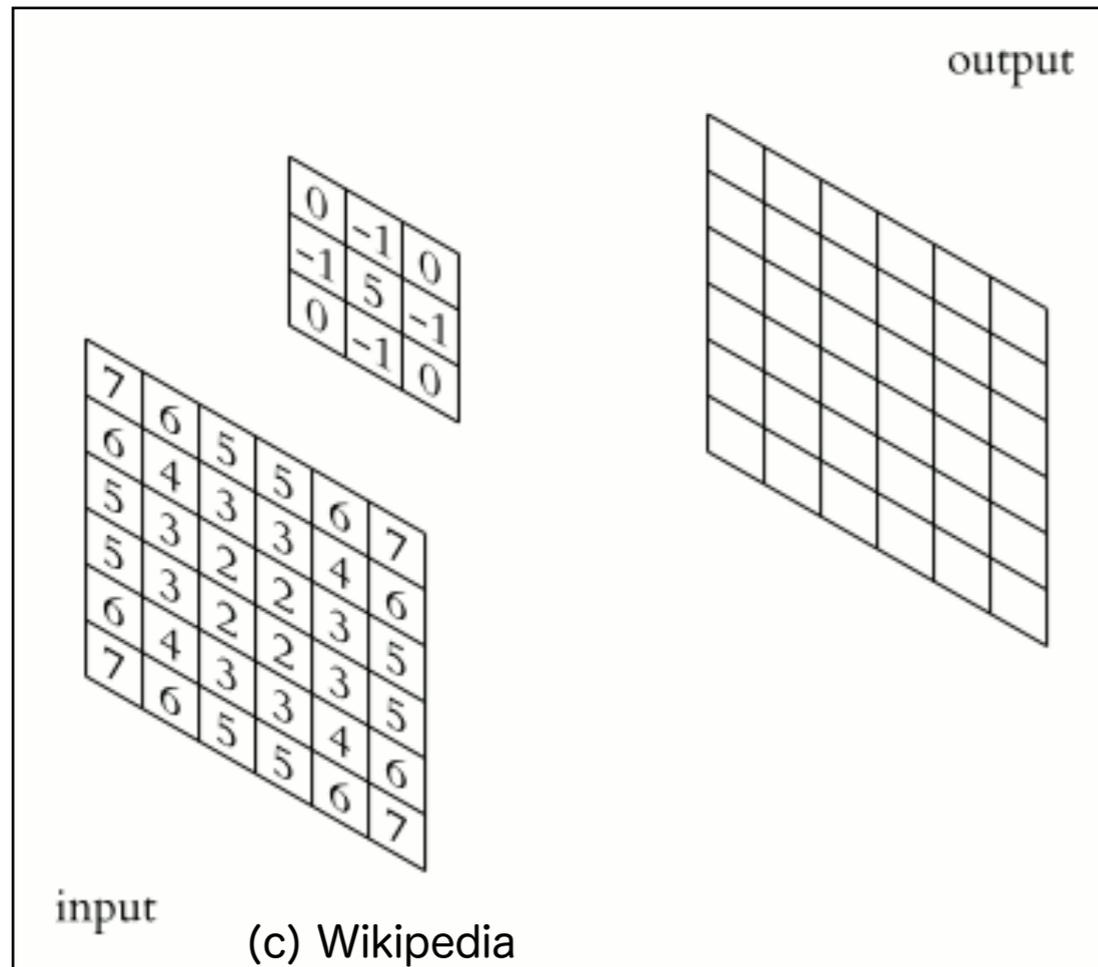


- 我々の電子飛跡画像
 - 入力画像: 30x30 画素
 - 出力: 36 等分した角度ビン
- 畳み込みニューラルネットワーク (CNN)
 - 視覚野の働きを模擬
 - 最高スコアのの一つ (誤認率: 0.2%)
 - 画像認識/選別が得意



ネットワークの構築

- TensorFlow (ver. 1.8.0) を利用
 - Google が開発中のオープンソースソフトウェア
 - 畳み込みニューラルネットワーク: 視覚野の働きを模擬
 - 多クラス分類を採用
- 入力 \vec{q} : 一様連続X線による 30x30 画素の飛跡画像
- 出力 $\vec{\phi}$: 角度を 36 等分した確率分布
- 訓練データ: MC シミュレーションで 0.5-12 keV の飛跡画像を作成



Layer #	Layer name	Image size
1	Input	30×30×1
2	Conv-3-16	30×30×16
3	Conv-3-16	30×30×16
4	Conv-3-16	30×30×16
5	Conv-3-16	30×30×16
6	Maxpool-2	15×15×16
7	Conv-3-32	15×15×32
8	Conv-3-32	15×15×32
9	Conv-3-32	15×15×32
10	Maxpool-2	8×8×32
11	Conv-3-64	8×8×64
12	Conv-3-64	8×8×64
13	Maxpool-2	4×4×64
14	Conv-3-128	4×4×128
15	Maxpool-2	2×2×128
16	FC-512	512
17	FC-512	36×3
18	Output	

ネットワークの構築

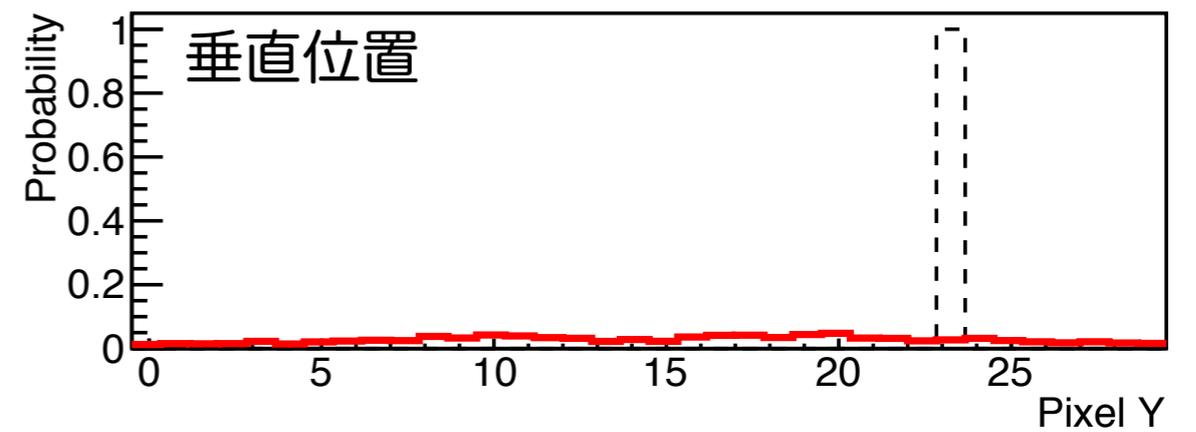
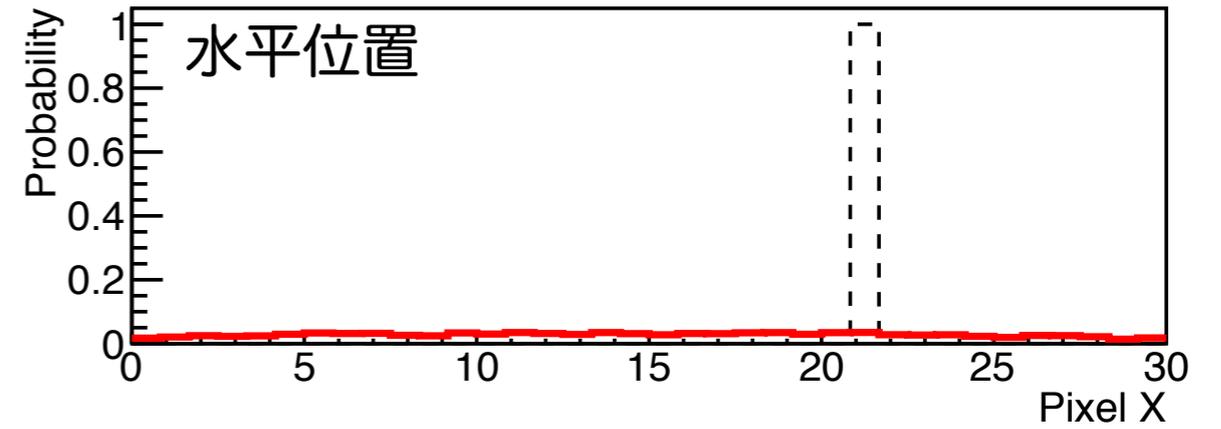
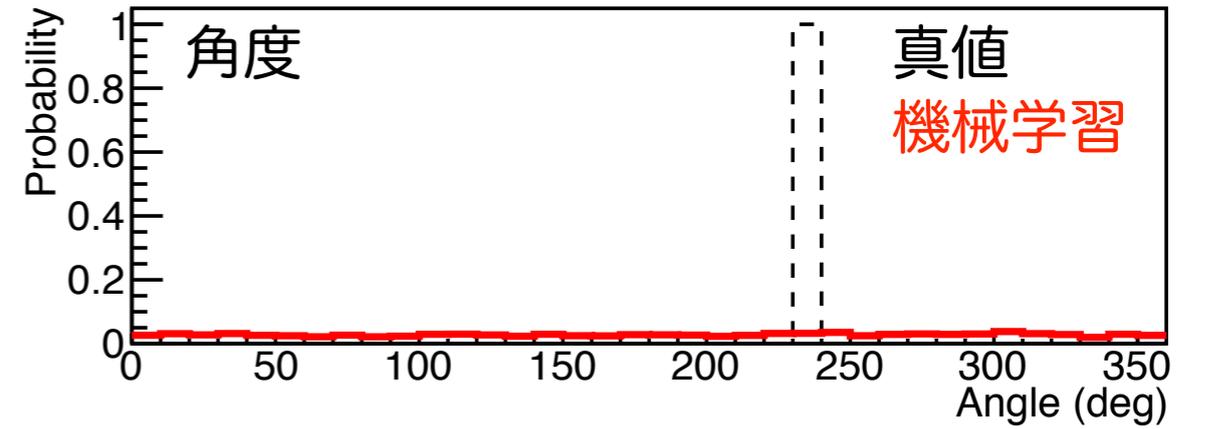
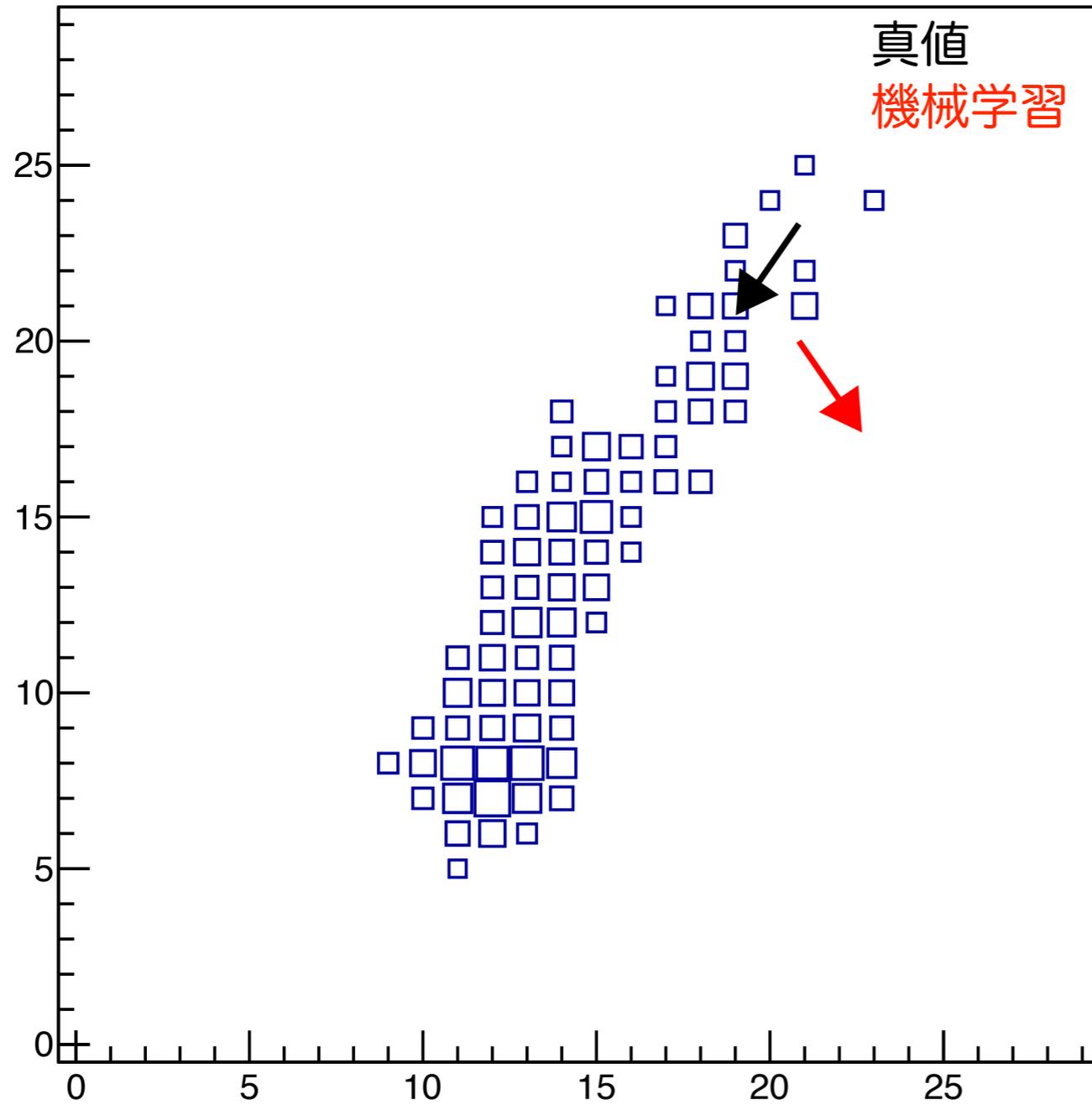
- TensorFlow (ver. 1.8.0) を利用
 - Google が開発中のオープンソースソフトウェア
 - 畳み込みニューラルネットワーク: 視覚野の働きを模擬
 - 多クラス分類を採用
- 入力 \vec{q} : 一様連続 X 線による 30x30 画素の飛跡画像
- 出力 $\vec{\phi}$: 角度を 36 等分した確率分布
- 訓練データ: MC シミュレーションで 0.5-12 keV の飛跡画像を作成
- 訓練: スペクトルフィットと同じ
 - $\vec{\phi} = F(\vec{q}; w)$ (w はパラメータ、約 74 万コ)
 - 勾配法を用いて、 $\vec{\phi}$ と $\vec{\phi}_{\text{true}}$ の交差エントロピー

$$H = \sum_i \phi_{\text{true},i} \log \phi_i \text{ を最小に}$$

(ネットワーク中の演算は、入力に対して微分可が必要)

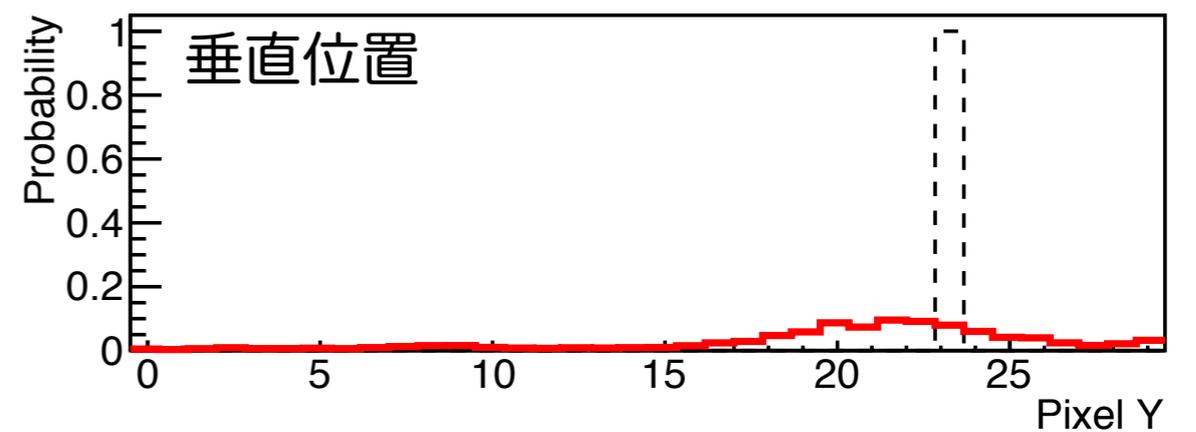
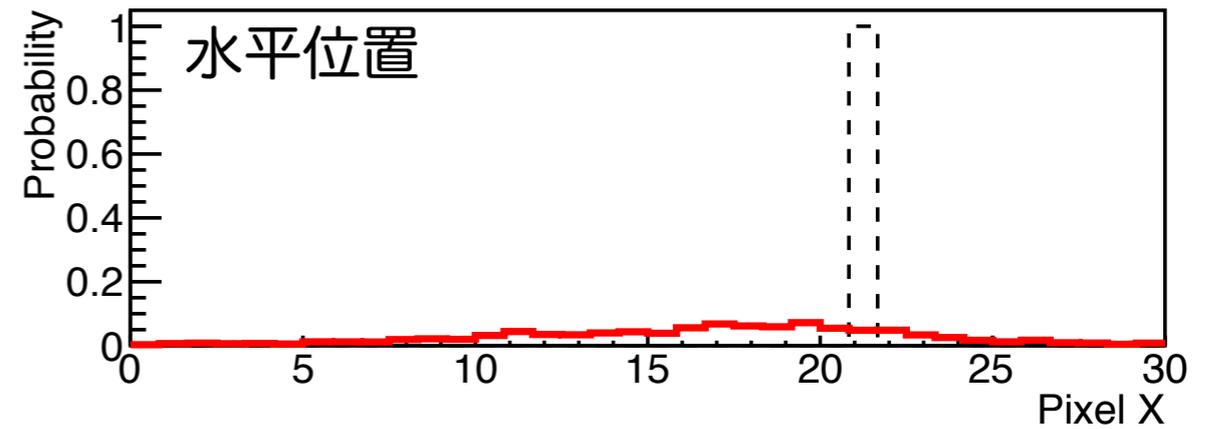
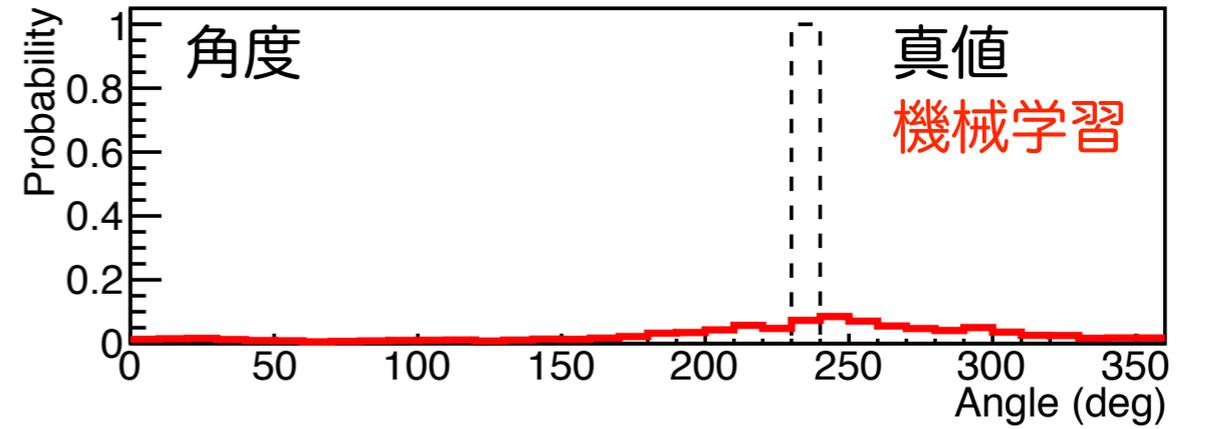
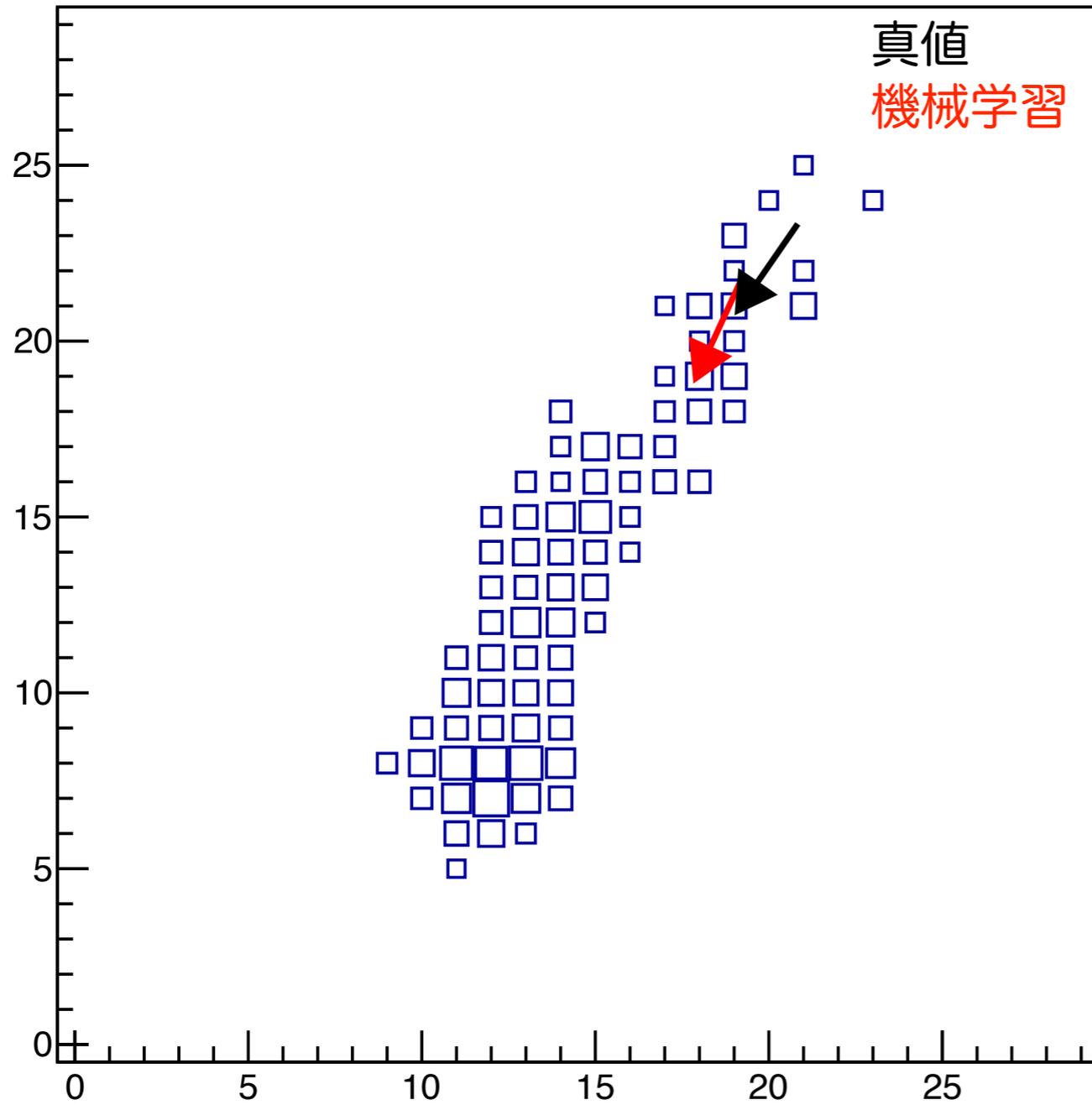
Layer #	Layer name	Image size
1	Input	30×30×1
2	Conv-3-16	30×30×16
3	Conv-3-16	30×30×16
4	Conv-3-16	30×30×16
5	Conv-3-16	30×30×16
6	Maxpool-2	15×15×16
7	Conv-3-32	15×15×32
8	Conv-3-32	15×15×32
9	Conv-3-32	15×15×32
10	Maxpool-2	8×8×32
11	Conv-3-64	8×8×64
12	Conv-3-64	8×8×64
13	Maxpool-2	4×4×64
14	Conv-3-128	4×4×128
15	Maxpool-2	2×2×128
16	FC-512	512
17	FC-512	36×3
18	Output	

機械学習の途中結果



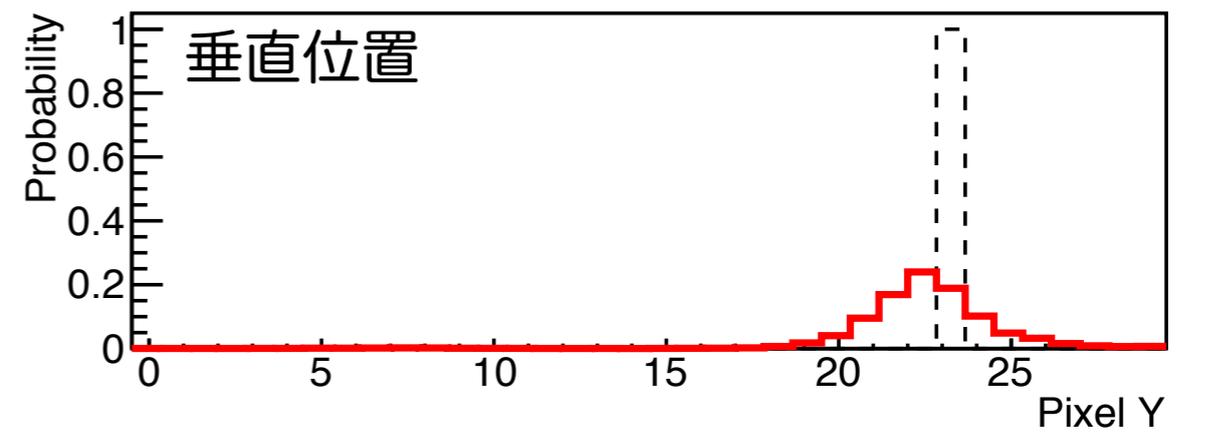
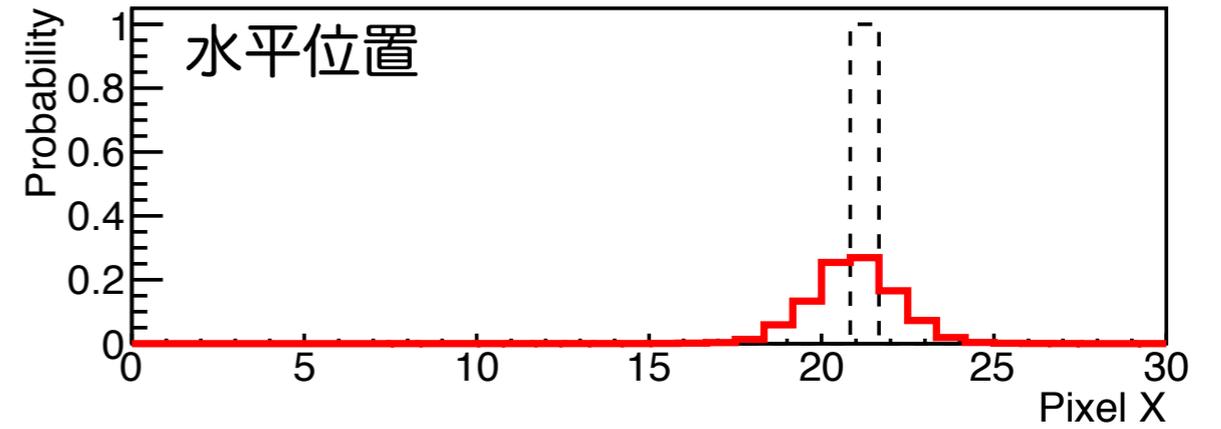
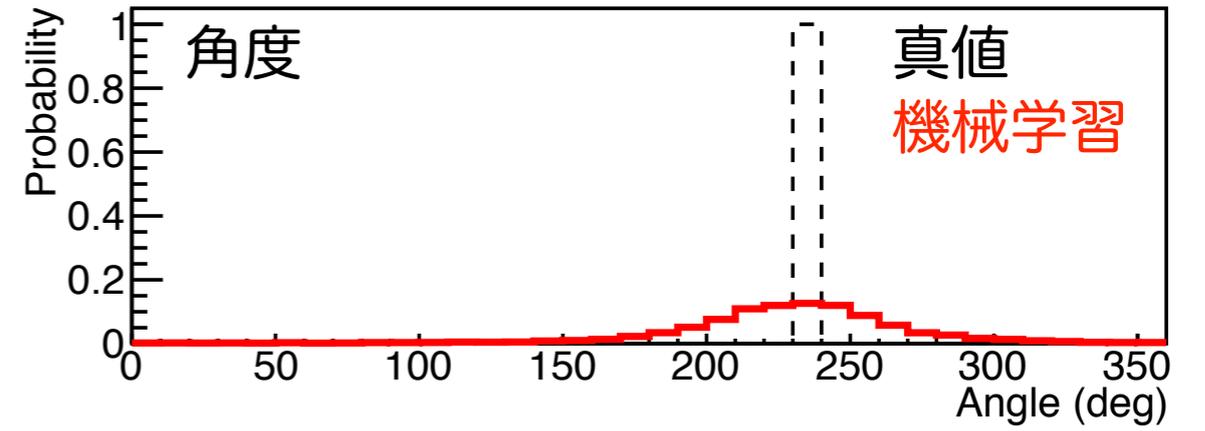
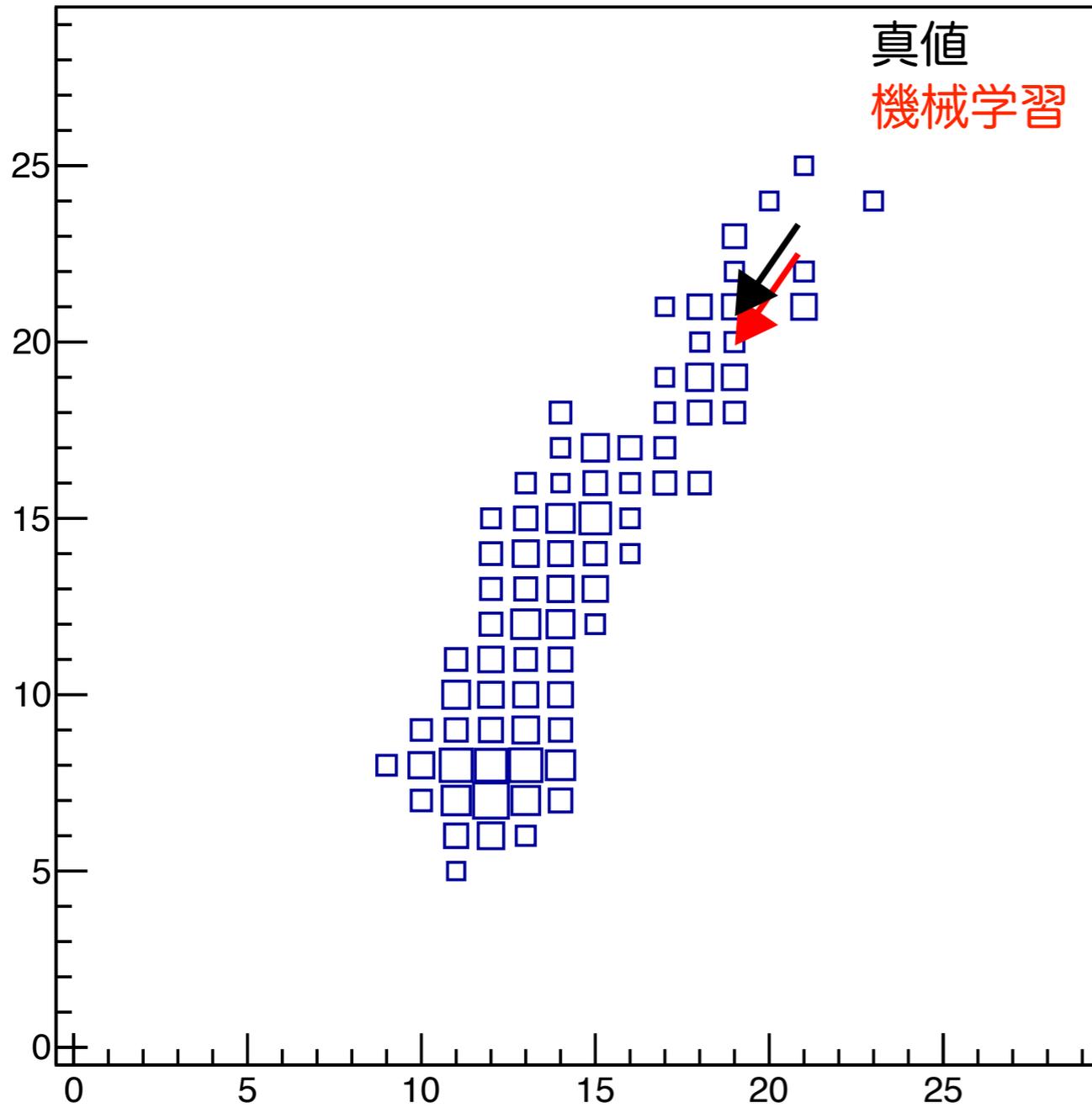
予測角度および位置は、確率分布の
最大値を含むビン中央値

機械学習の途中結果



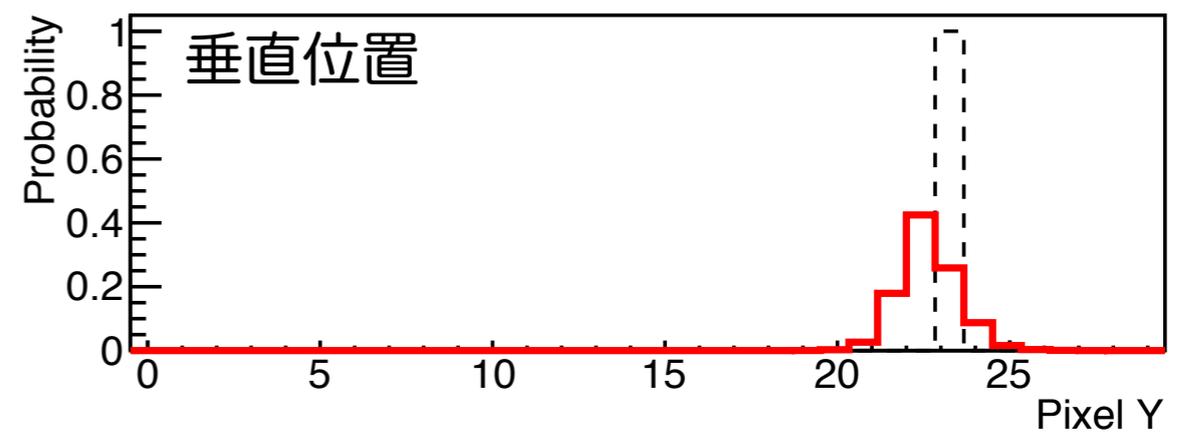
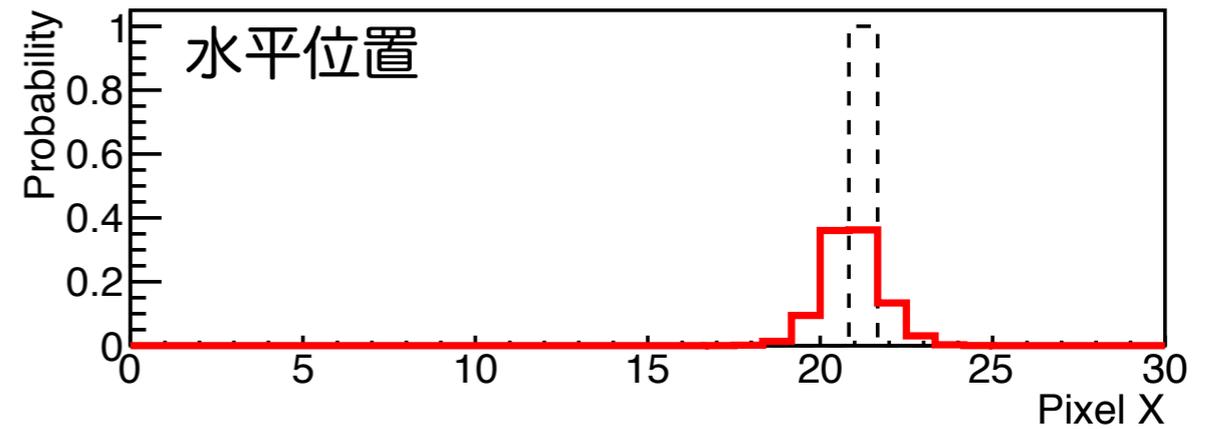
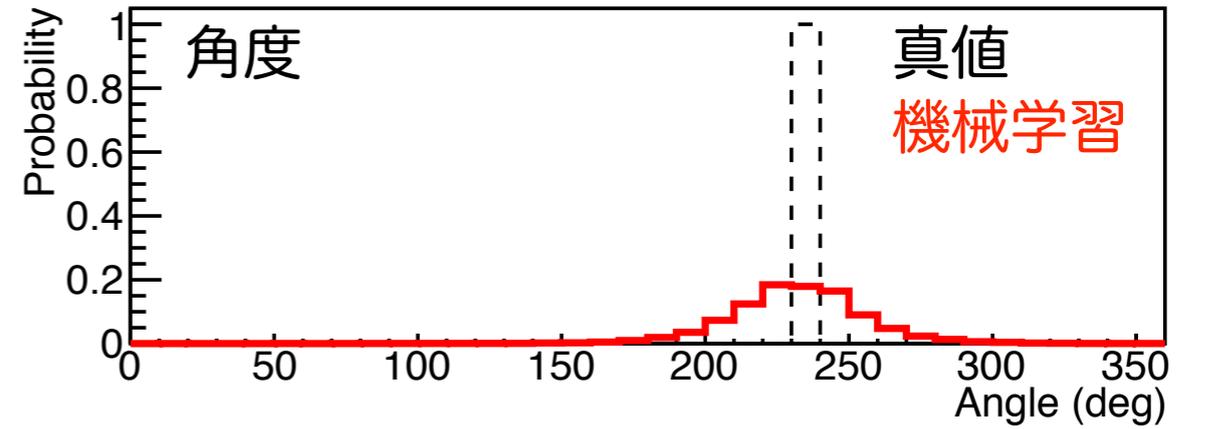
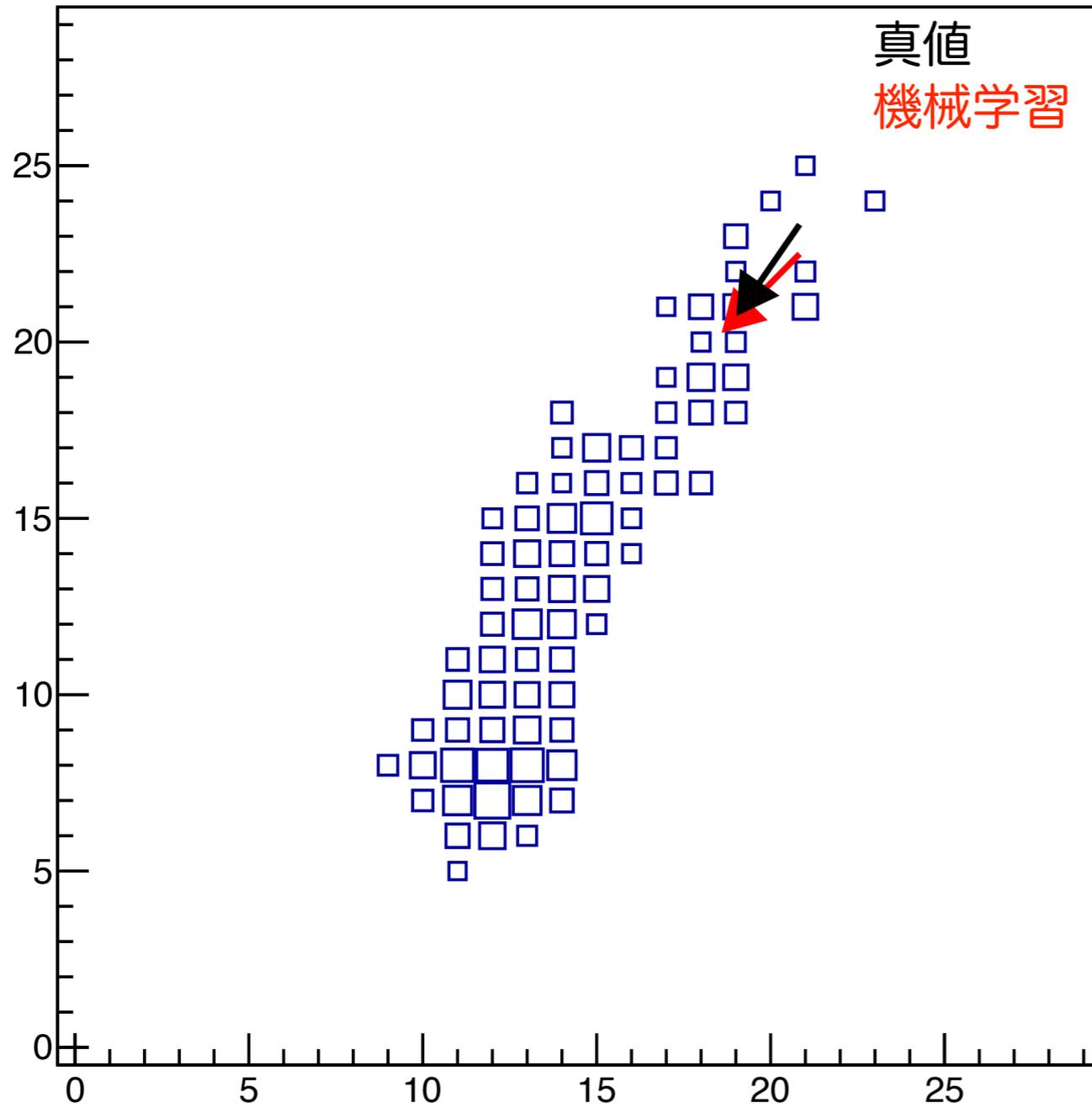
予測角度および位置は、確率分布の
最大値を含むビン中央値

機械学習の途中結果



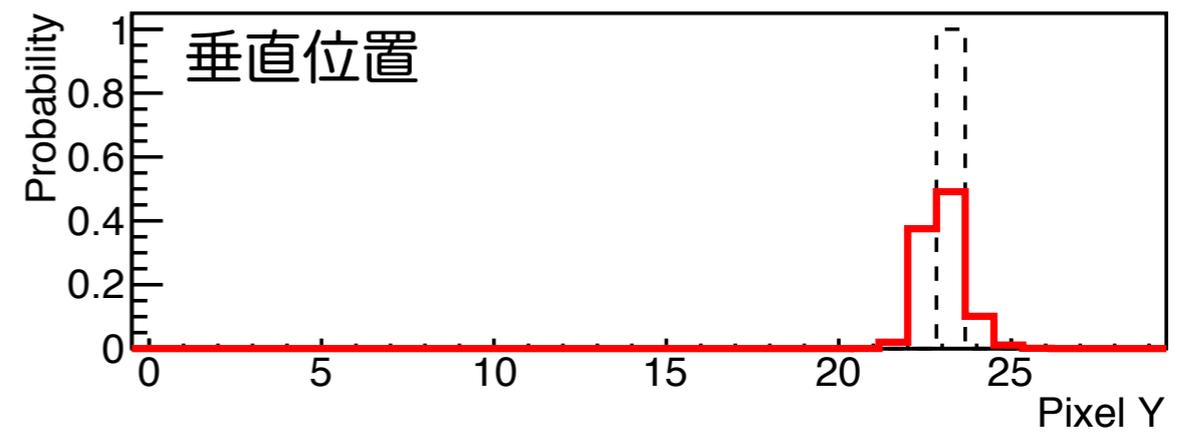
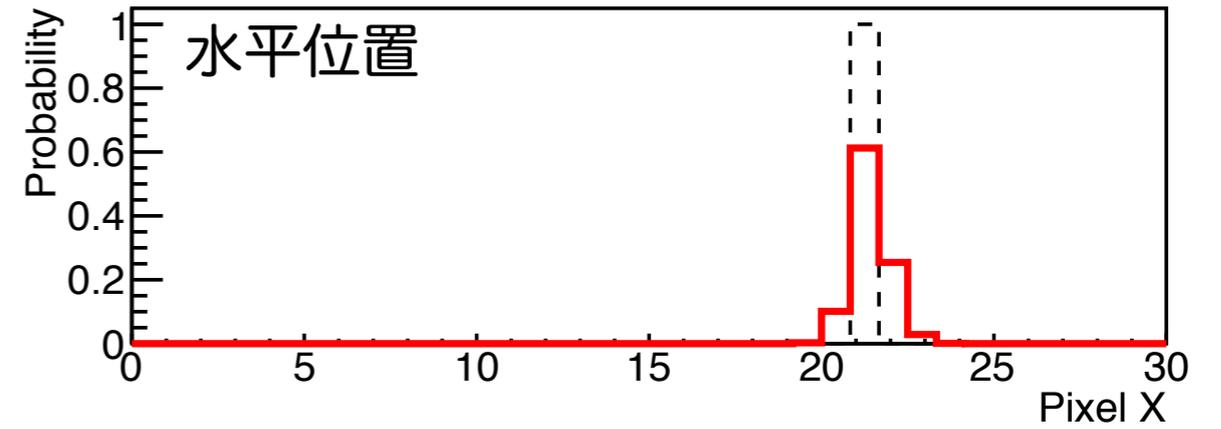
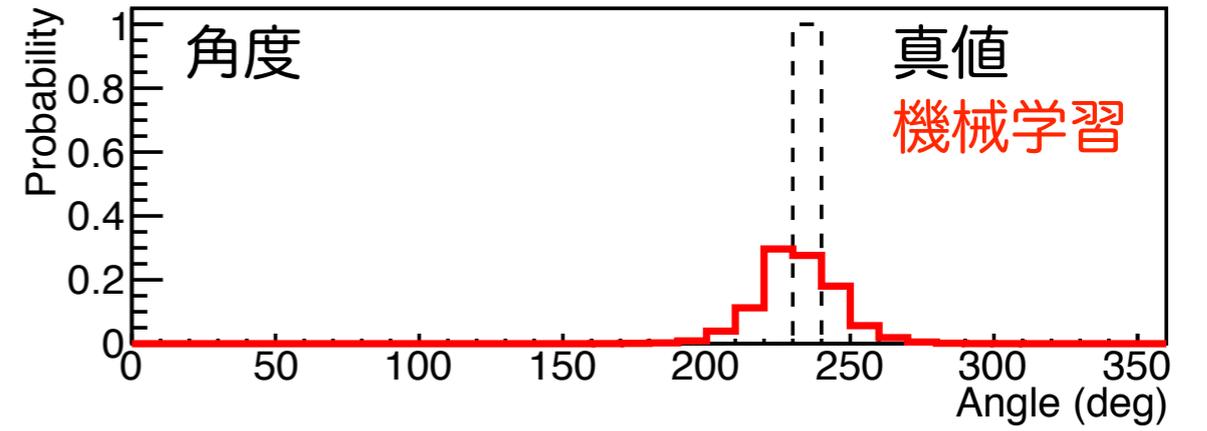
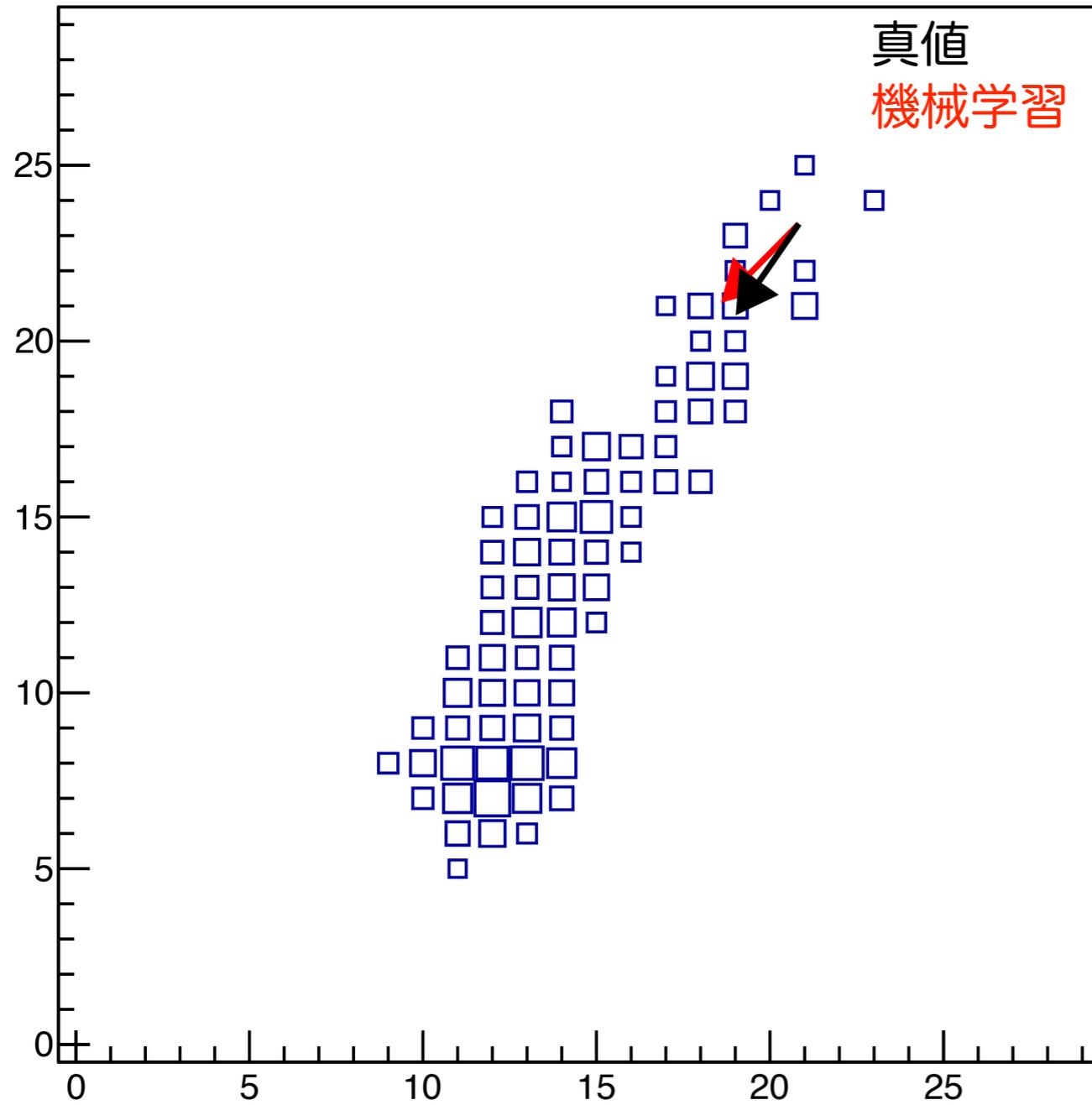
予測角度および位置は、確率分布の
最大値を含むビン中央値

機械学習の途中結果



予測角度および位置は、確率分布の
最大値を含むビン中央値

機械学習の途中結果



予測角度および位置は、確率分布の
最大値を含むビン中央値

学習の過程

- 学習法

- 損失関数 (L) を定義し、それを最小化

$$L = H + \lambda_2 \sum w^2$$

- 正規化項: 重みの2乗和 (過学習を防止)
- ミニバッチ学習: 訓練データ 360 万画像から 360 枚をランダムかつ重複なしに選んで入力 (学習の効率化)

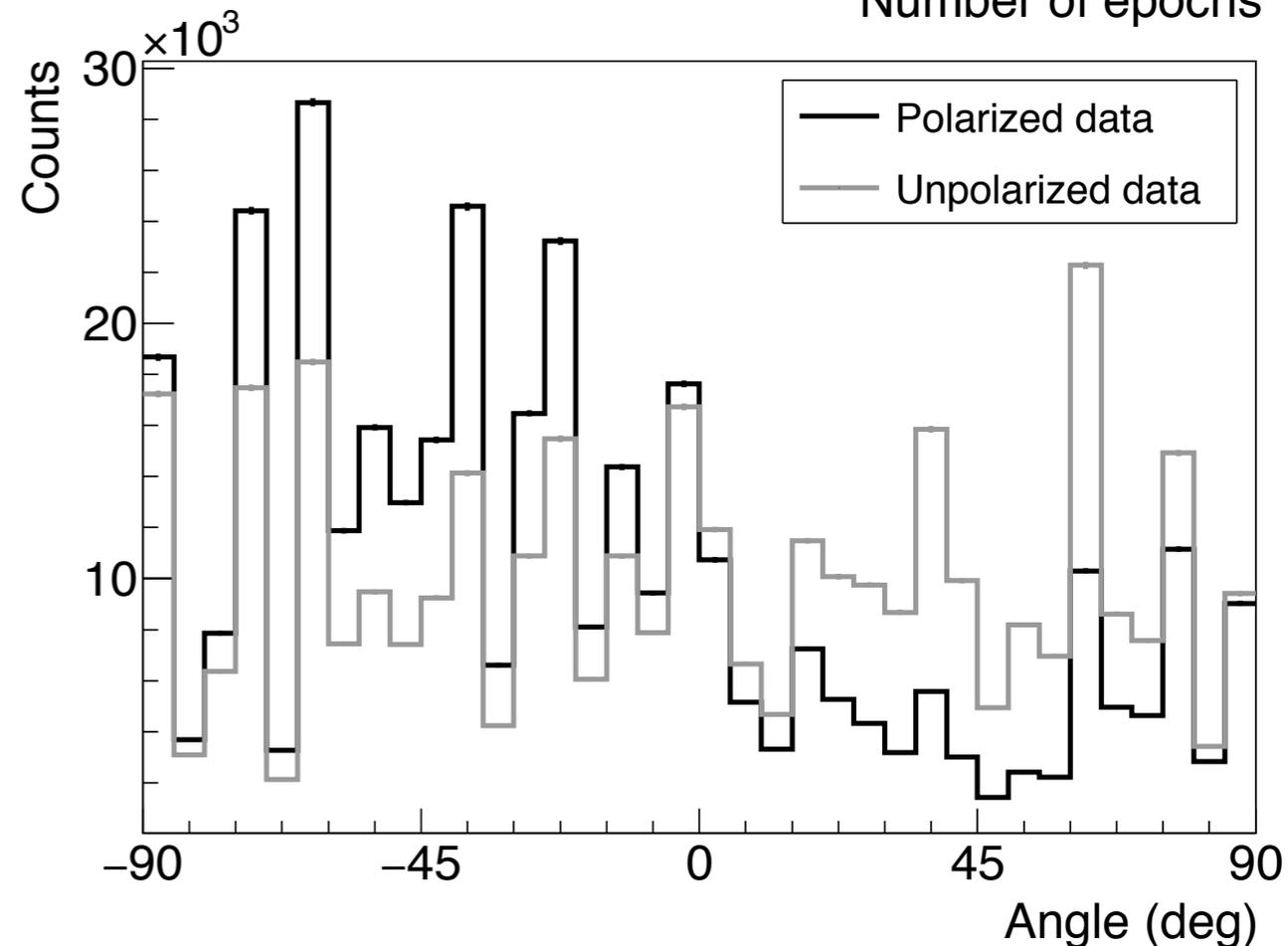
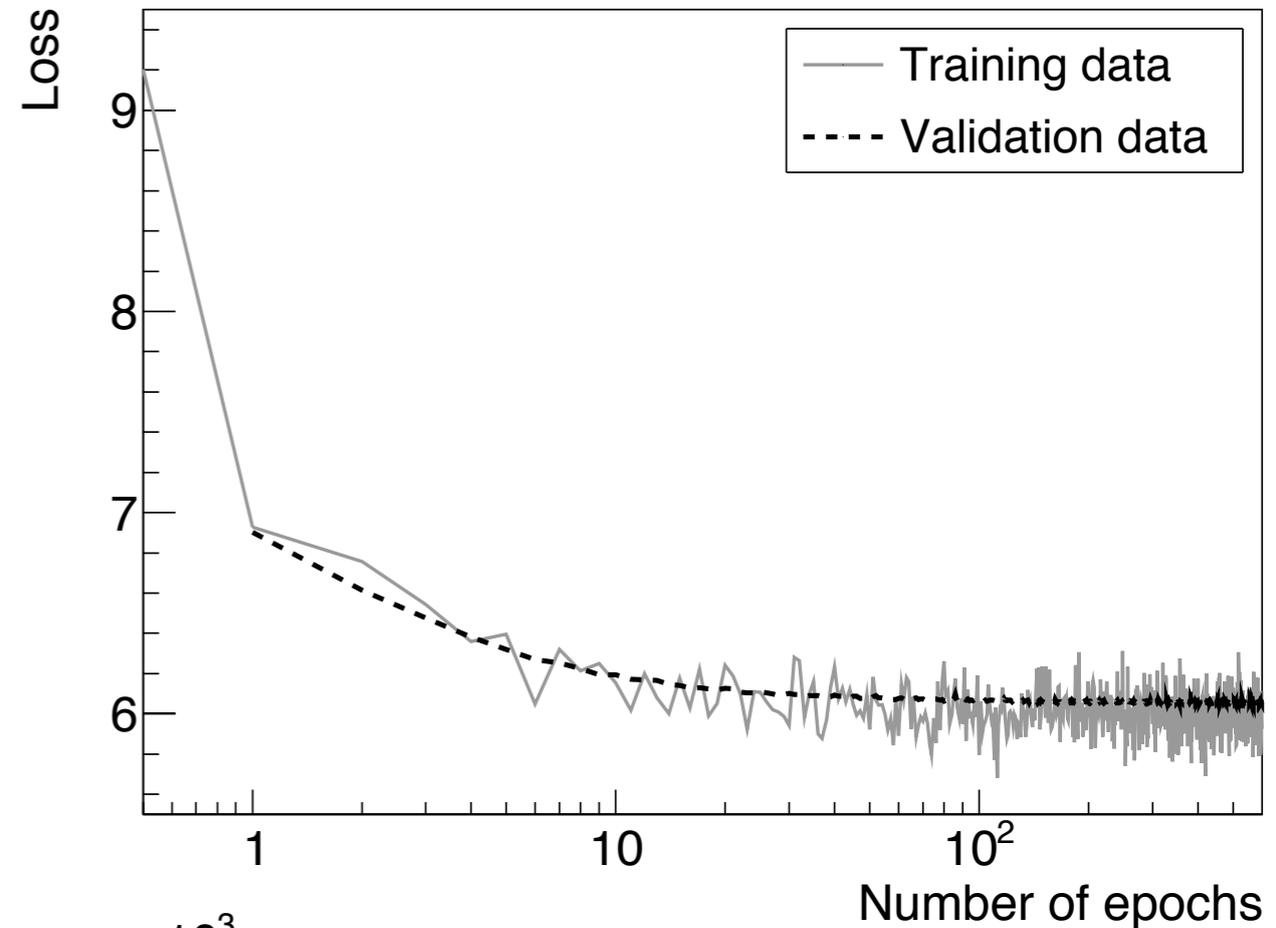
- GPU (NVIDIA GeForce GTX 1080) で

1 日の間、並列計算

- 2560 コア、1.6 GHz、8 GB メモリ
- 最小の L を持つネットワークを保存

- 8 keV テストデータに適用 (学習に未使用)

- 2つのデータ: 無偏光・-45 deg に偏光
- 無偏光カーブがフラットではない



学習の過程

学習法

- 損失関数 (L) を定義し、それを最小化

$$L = H + \lambda_2 \sum w^2$$

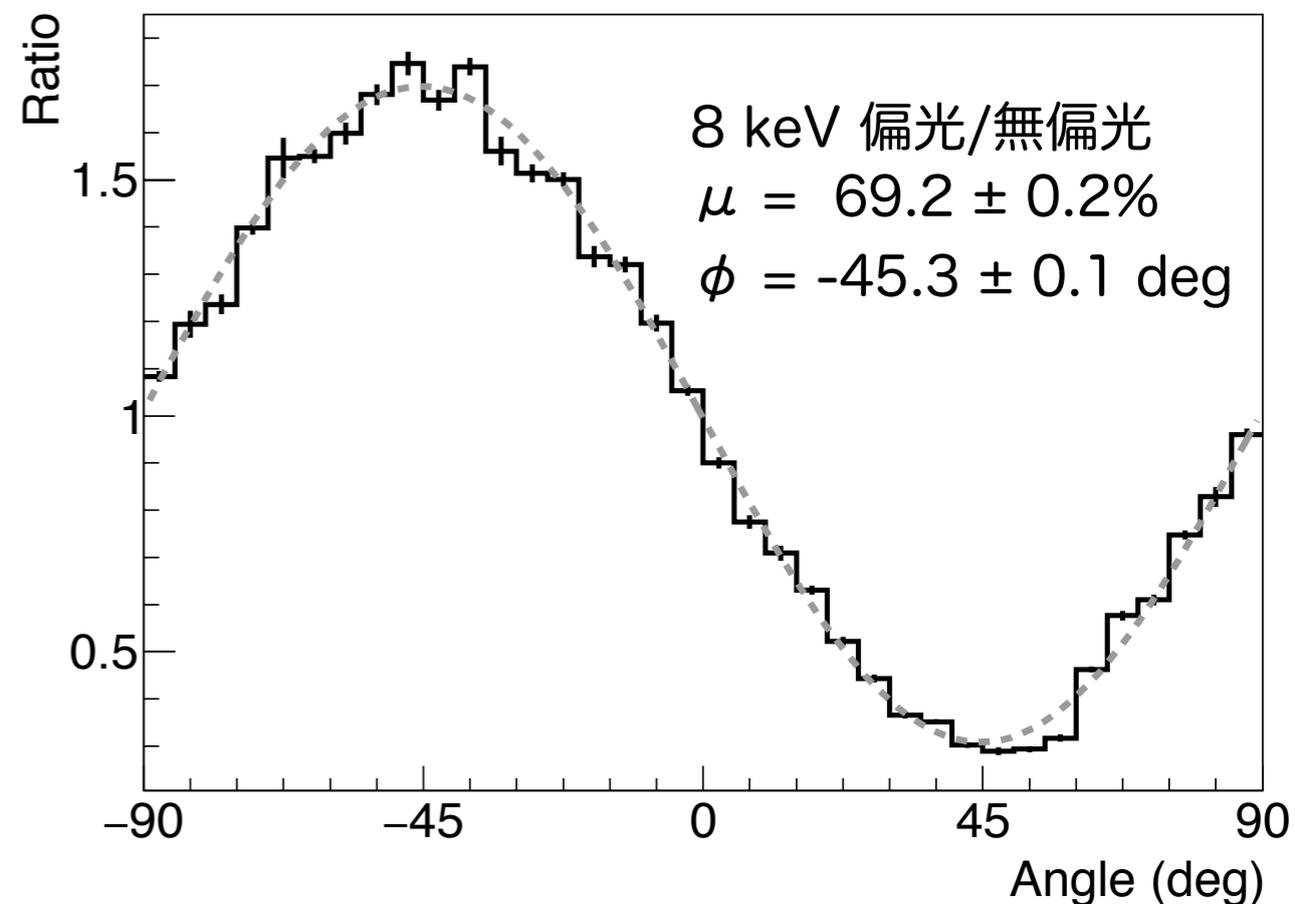
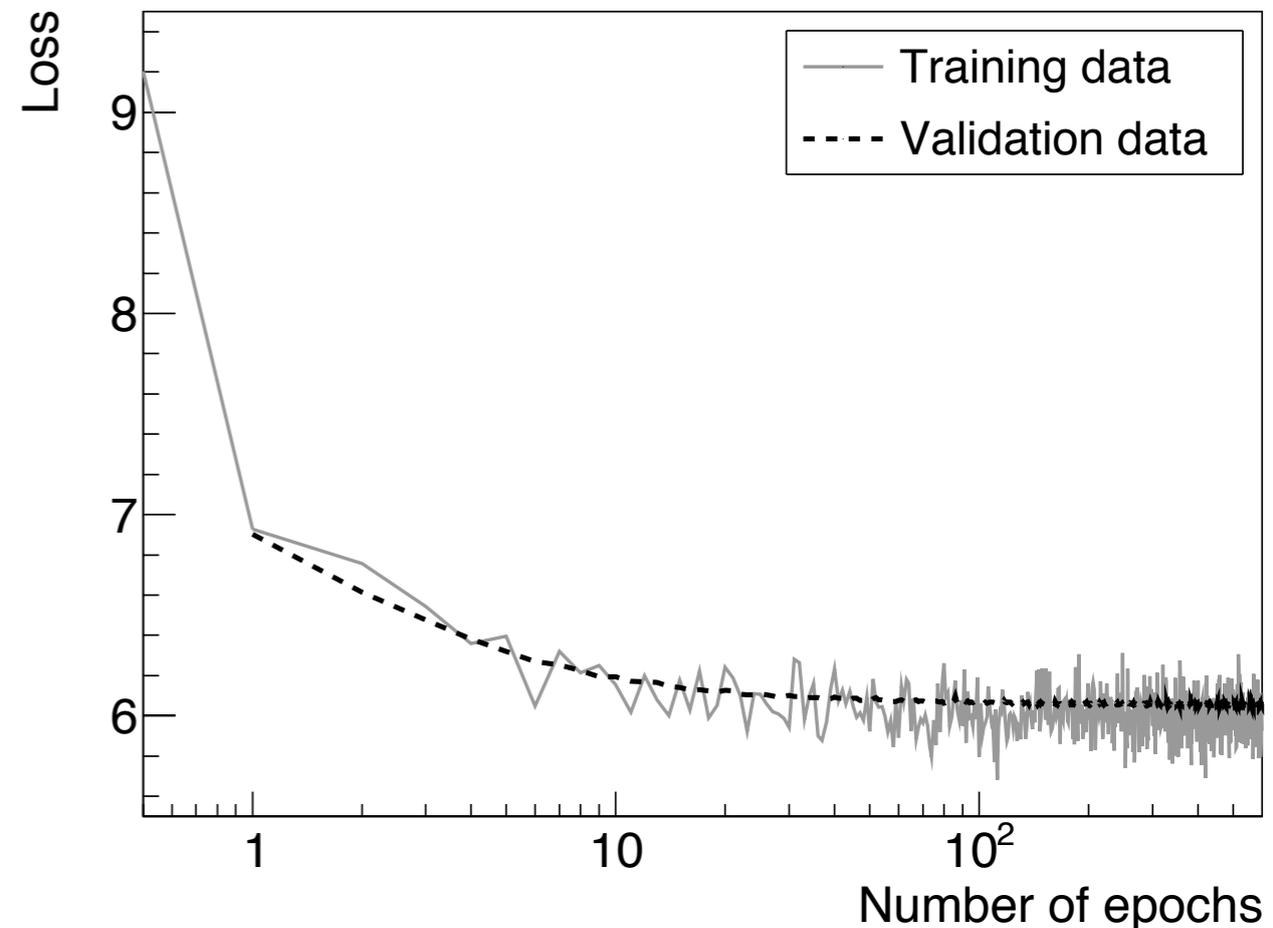
- 正規化項: 重みの2乗和 (過学習を防止)
- ミニバッチ学習: 訓練データ 360 万画像から 360 枚をランダムかつ重複なしに選んで入力 (学習の効率化)

GPU (NVIDIA GeForce GTX 1080) で 1 日の間、並列計算

- 2560 コア、1.6 GHz、8 GB メモリ
- 最小の L を持つネットワークを保存

8 keV テストデータに適用 (学習に未使用)

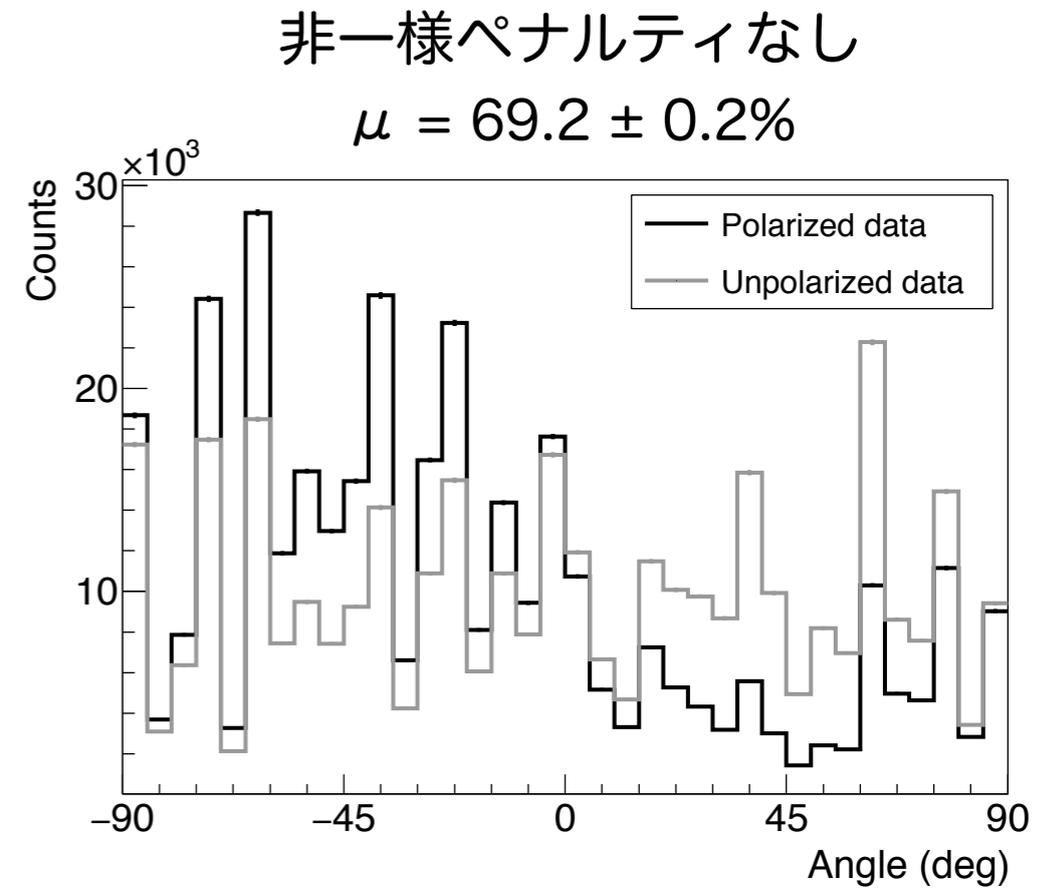
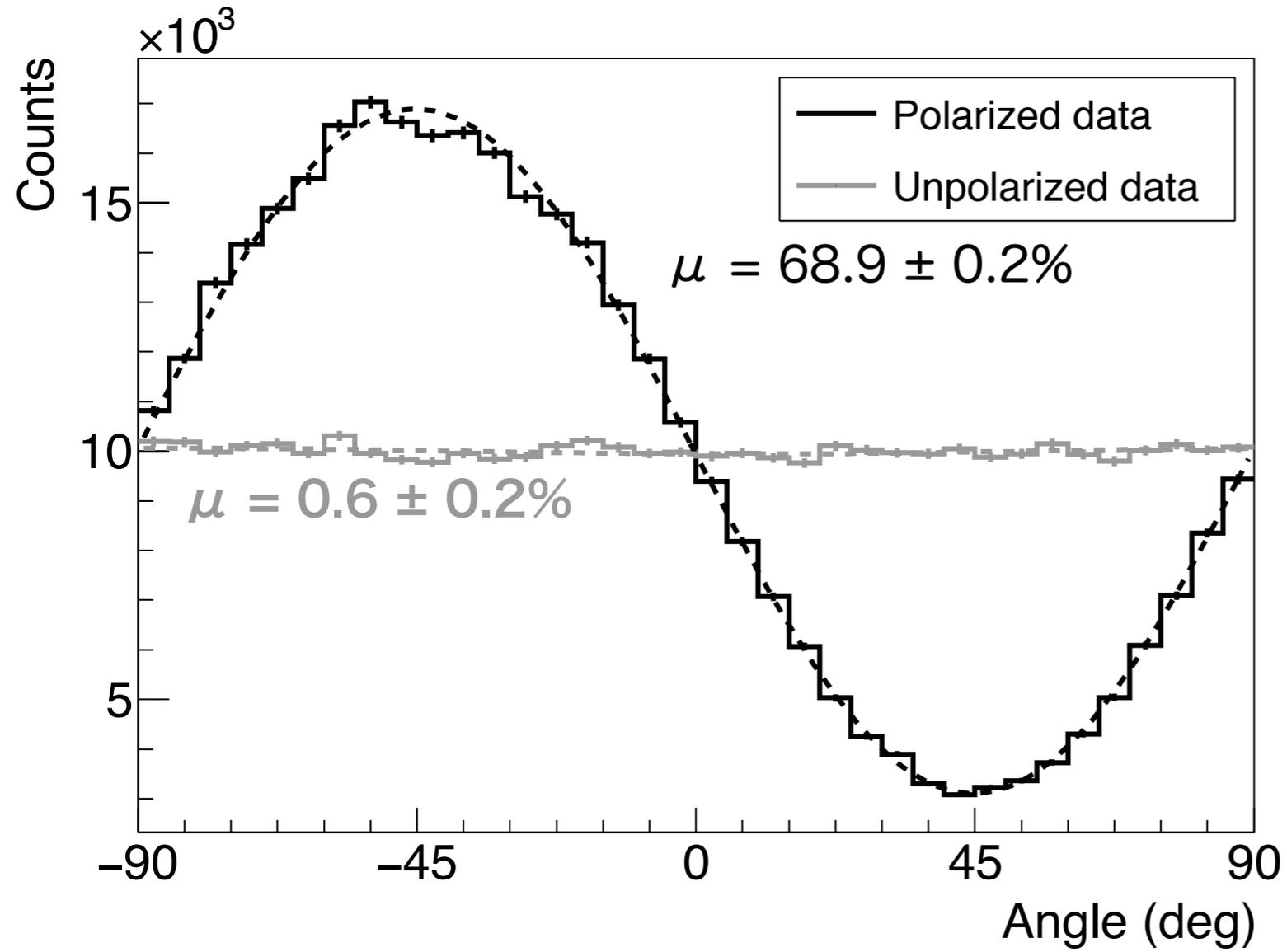
- 2つのデータ: 無偏光・-45 deg に偏光
- 無偏光カーブがフラットではない
- 偏光/無偏光カーブは、正弦波に
 - ▶ 偏光検出効率: $\mu = 69.2 \pm 0.2\%$
- 検出器の一樣回転でも、正弦波に



無偏光カーブを一様に

- 無偏光カーブがフラットなら、偏光解析が簡単に (Strohmer+17)
 - 偏光応答関数 $X(E', E, \varphi', \varphi) = \varepsilon(E') R(E', E) V(E', \varphi', \varphi)$
 $= \varepsilon(E') R(E', E) \mu(E')$
- 損失関数に、非一様な角度分布へのペナルティ項 (L_{FL}) を追加
 - $L = H + \lambda_2 \sum w^2 + \lambda_{\text{FL}} L_{\text{FL}}$
 - L_{FL} は微分可能な演算子で計算 (誤差逆伝播法の条件)
 - H テスト統計量 を応用
- H テスト統計量
 - 天体からのX/γ線の周期的信号を探す方法
 - m 次までのフーリエパワーの合計
$$Z_m^2(\bar{\varphi}_j) = \frac{2}{N} \sum_{k=1}^m \left[\left(\sum_{j=0}^{N-1} \cos k\bar{\varphi}_j \right)^2 + \left(\sum_{j=0}^{N-1} \sin k\bar{\varphi}_j \right)^2 \right]$$
 - $H_M = \max_{1 \leq m \leq 20} (Z_m^2 - 4m + 4) = Z_M^2 - 4M + 4 \geq 0$
- ミニバッチ 18,000 枚 (GPU メモリで制限) に増加。1 次以上のフーリエパワーの合計が最小になるように学習。

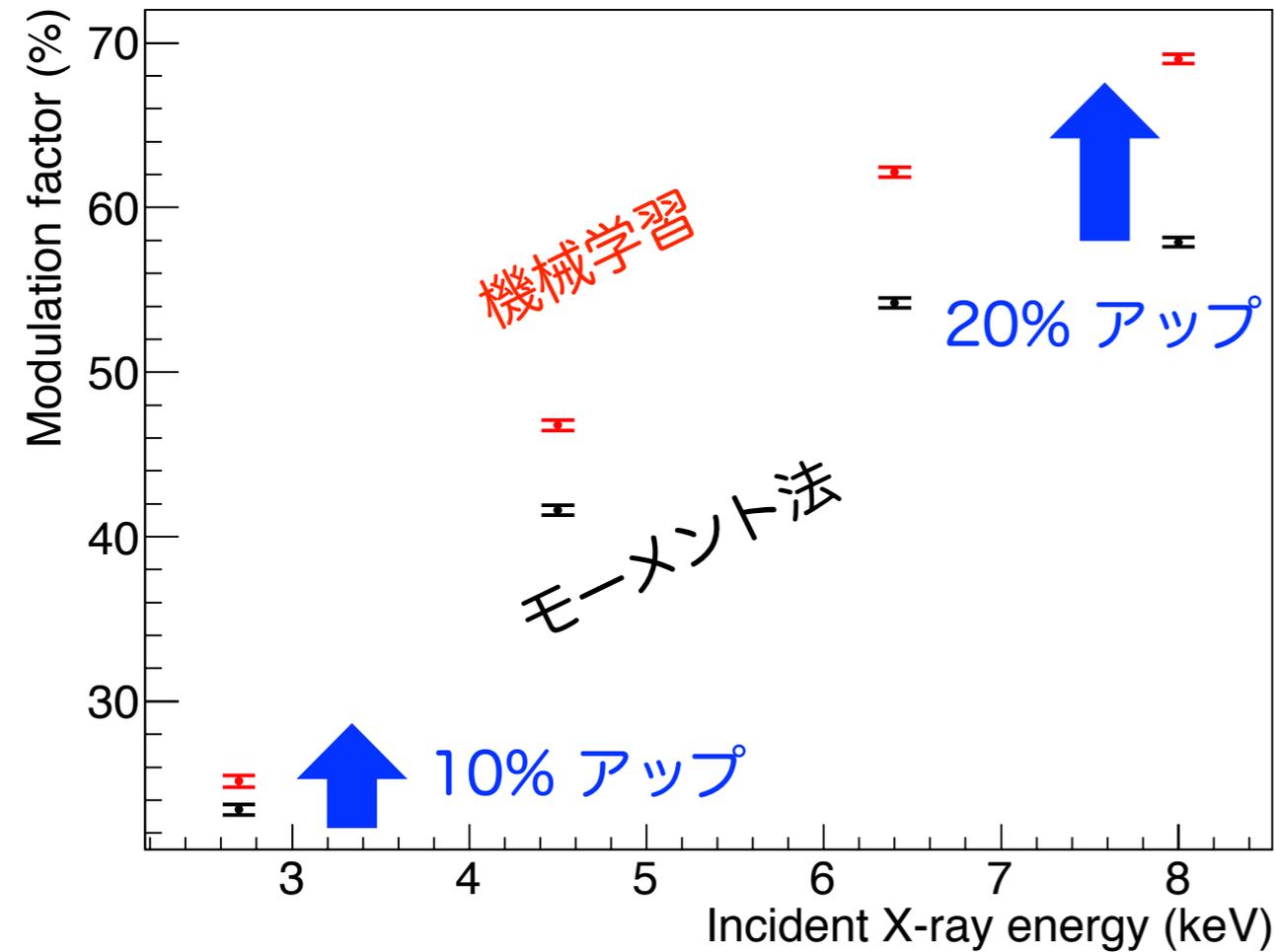
無偏光カーブを一様に



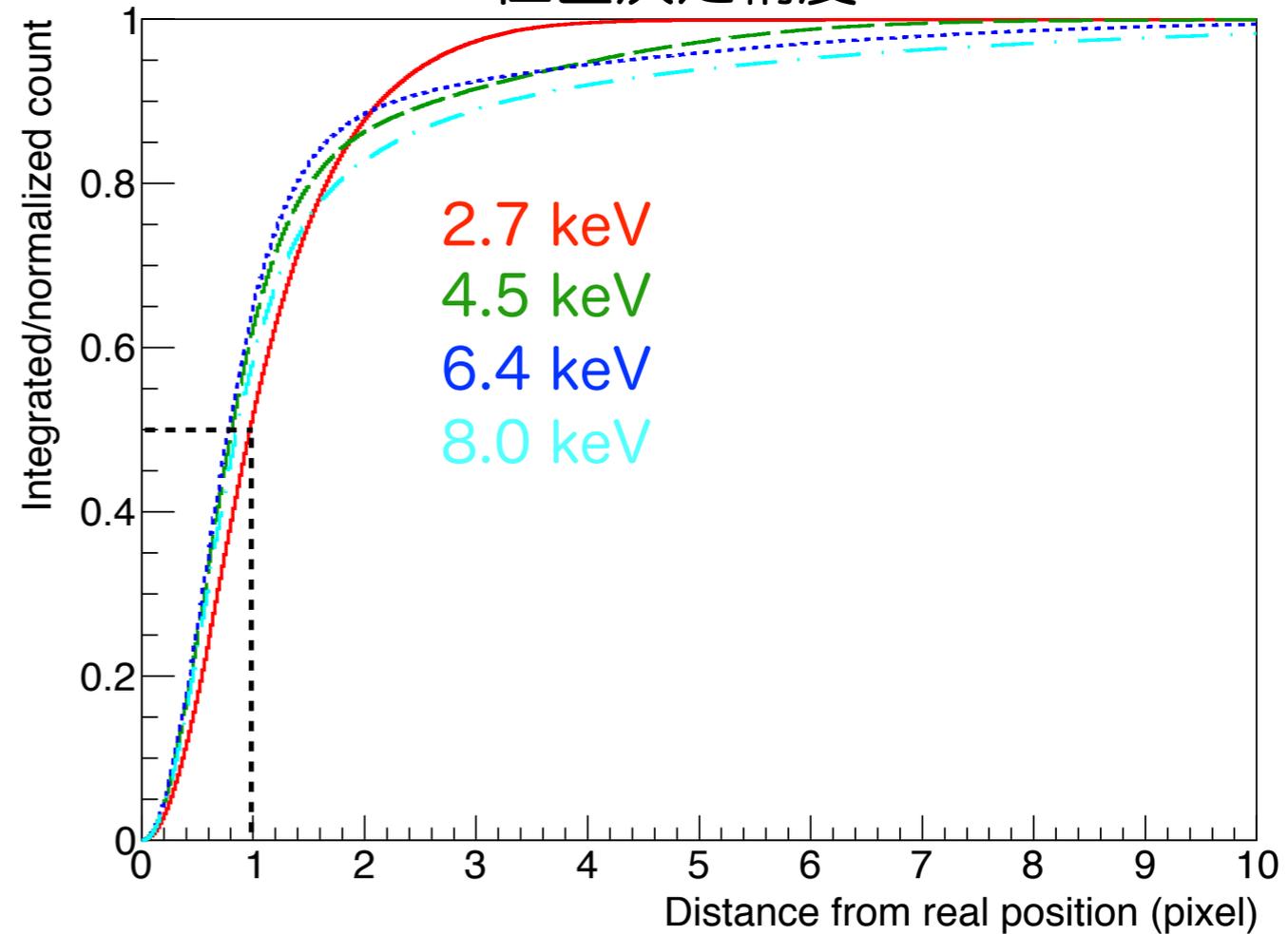
機械学習の結果

単色X線テストデータ 36万画像を処理

偏光検出効率



位置決定精度



偏光検出効率 μ x 10-20% 向上

位置決定精度 HPD < 2.0 pixel ~ 240 μ m

1週間の観測時間が5日に短縮可

なぜ機械学習は偏光感度を改善したか？

- 今までのイメージモーメント法
 - 角度 $\phi = F(M_{00}, M_{11}, M_{02}, M_{20})$
 - 30x30 画素を **4 つ**のモーメントまで情報の次元を下げて、それらと ϕ との相関で計算
- 機械学習
 - 30x30 画素を **512 ノード**まで次元を下げて、(最後尾の全結合層で) それらと ϕ との相関を学習
 - 相関を取る情報量が圧倒的に多く、画像情報を効率的に使用

まとめ

- 天体 X 線偏光を観測する IXPE 衛星を推進中。2021 年に打上予定。光電子の飛跡画像から偏光情報を抽出する方法が肝。
- Google 社が開発中の TensorFlow を用いて、畳み込みニューラルネットワークを構築。TPC 偏光計の飛跡画像を入力し、光電子の射出方向を学習。
 - 訓練/テストデータは、MC シミュレーションで作成
 - 生画像を入力 (事前処理なし)
 - 無偏光カーブをフラットにする損失項として、H テスト統計量を転用
- 偏光検出効率 (μ) は、モーメント法に比べて、x10-20% の改善
- 位置決定精度は 2.0 pixel (or 240 μm) 以下
- IXPE 飛跡画像にも適用し、高エネルギー側で μ は同様に向上。