

# 高輝度LHC-ATLAS実験に向けた 深層学習による陽子対衝突点高速再構成

神戸大学 修士1年 笹田真宏  
2025/2/18 ICEPPシンポジウム



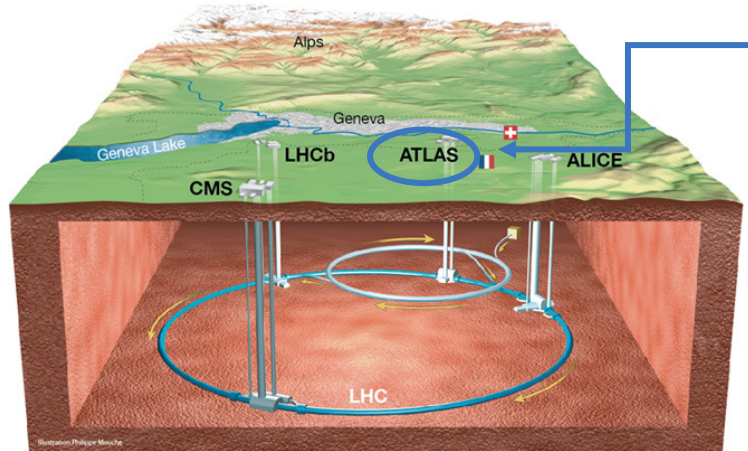
# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

# 高輝度LHC-ATLAS実験

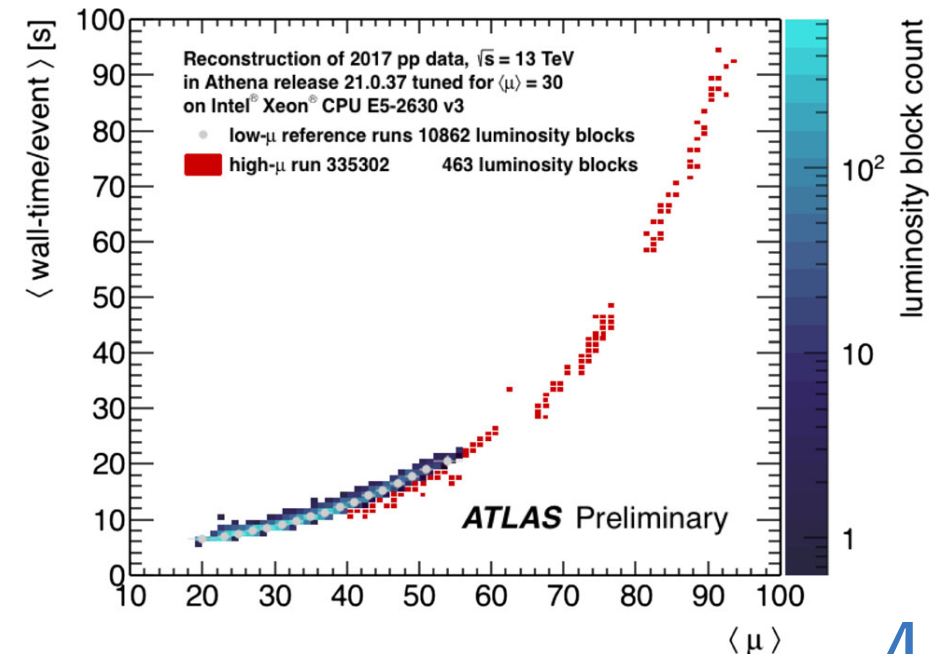


## LHC-ATLAS実験

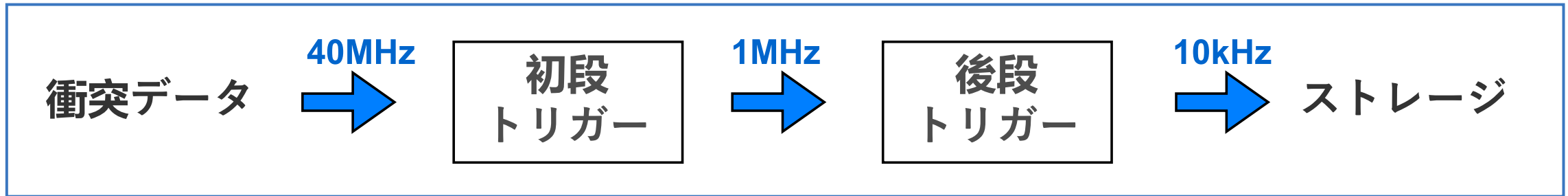
- LHCで行われる実験プロジェクト
- 陽子を世界最高エネルギーまで加速し衝突させる
- ヒッグス粒子の精密測定や「標準理論」を超える新物理の探索

## 高輝度化アップグレード(2030年)

- 陽子の衝突頻度を上げることが計画されている
  - HL-LHCで予想されるイベントに対し、トリガーシステムでは多くの演算能力が必要！
- **トリガーアルゴリズムの改良が重要**



# オンライントリガーシステム



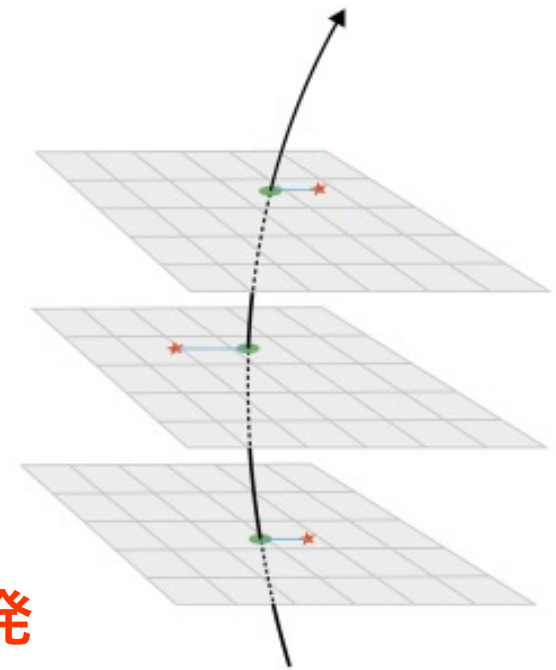
## ◆ 初段トリガー

- ・ ハードウェアベースの高速な事象選別

## ◆ 後段トリガー

- ・ ソフトウェアベースの詳細な事象選別
- ・ 荷電粒子の飛跡再構成
  - ・ Fast TrackとPrecision Trackingの2ステップ
  - ・ 内部検出器のヒット情報を結ぶ

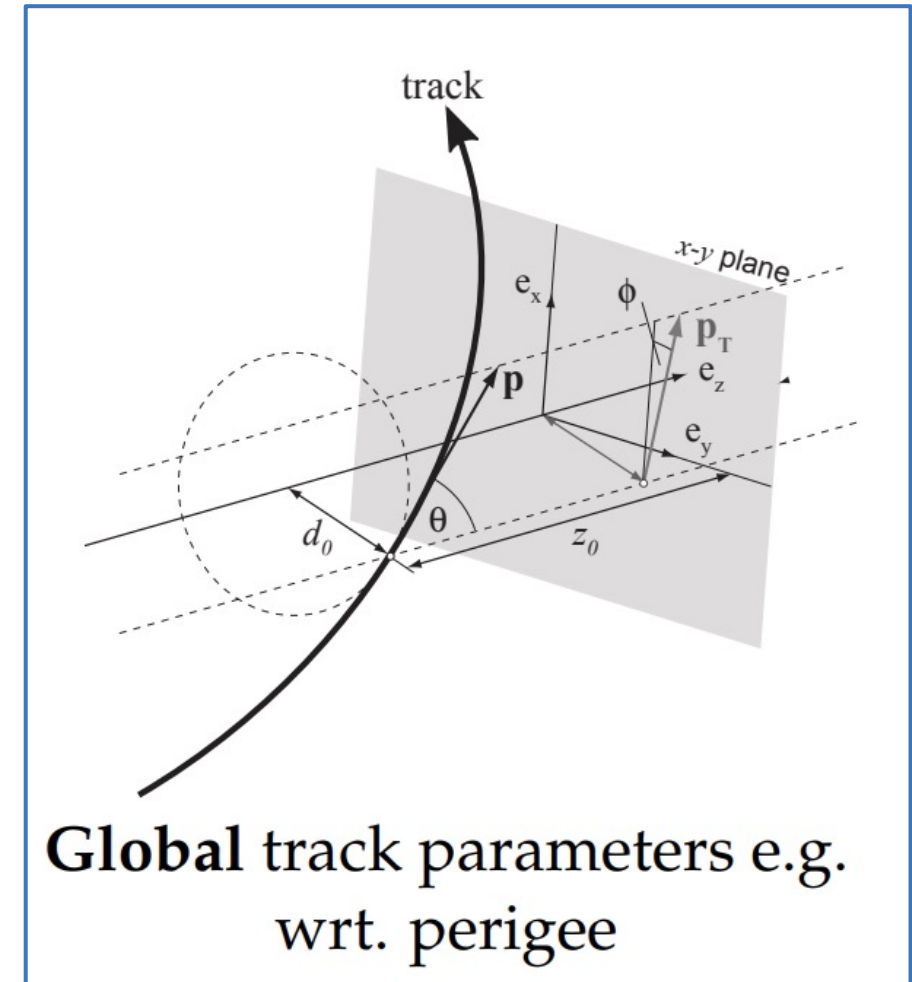
**本研究：再構成された飛跡を用いた新しいアルゴリズムの開発**



# トラックパラメータ

飛跡再構成によって再構成されるトラックは  
いくつかのパラメータで表される

- **インパクトパラメータ**
  - $d_0$ : ビーム軸と最接近点の横方向の距離
  - $z_0$ : ビームスポットからの縦方向の距離
- 方位角  $\phi$ , 極角  $\theta$
- 曲率  $\frac{q}{p}$
- 擬ラピディティ  $\eta$
- フィット精度  $\chi^2$



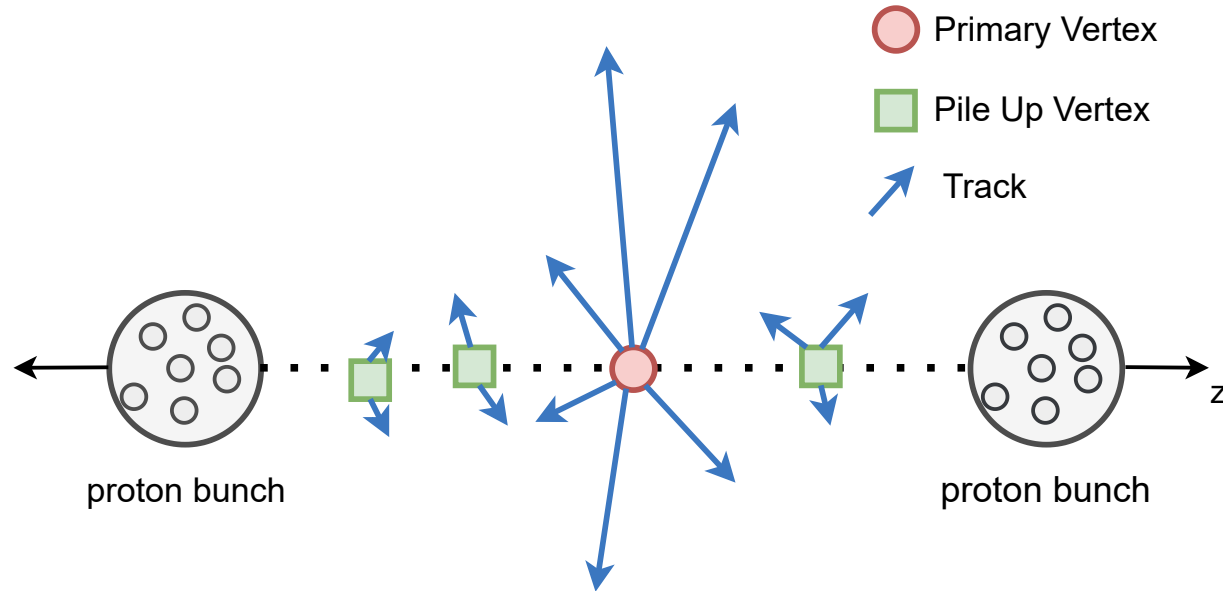
本研究では、飛跡再構成後のアルゴリズムを扱う

# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

# Primary Vertexの高速再構成

陽子のバンチが交差すると多数(約200個)の陽子が衝突する



陽子の衝突点をVertexと呼ぶ

**Primary Vertex(PV):**

陽子が最も高い横運動量で  
非弾性散乱したVertex

**Pile Up Vertex:**

PV以外のVertex

## 本研究の概要

トリガー上でPVのビーム軸方向の座標を再構成することを目指す

- PVから派生したトラックを選別
- 後段のアルゴリズム(例: METトリガーなど)に活用が可能



# 現行の手法: Adaptive multi-vertex fitter (AMVF)

現行のオフラインVertex再構成で使われている手法

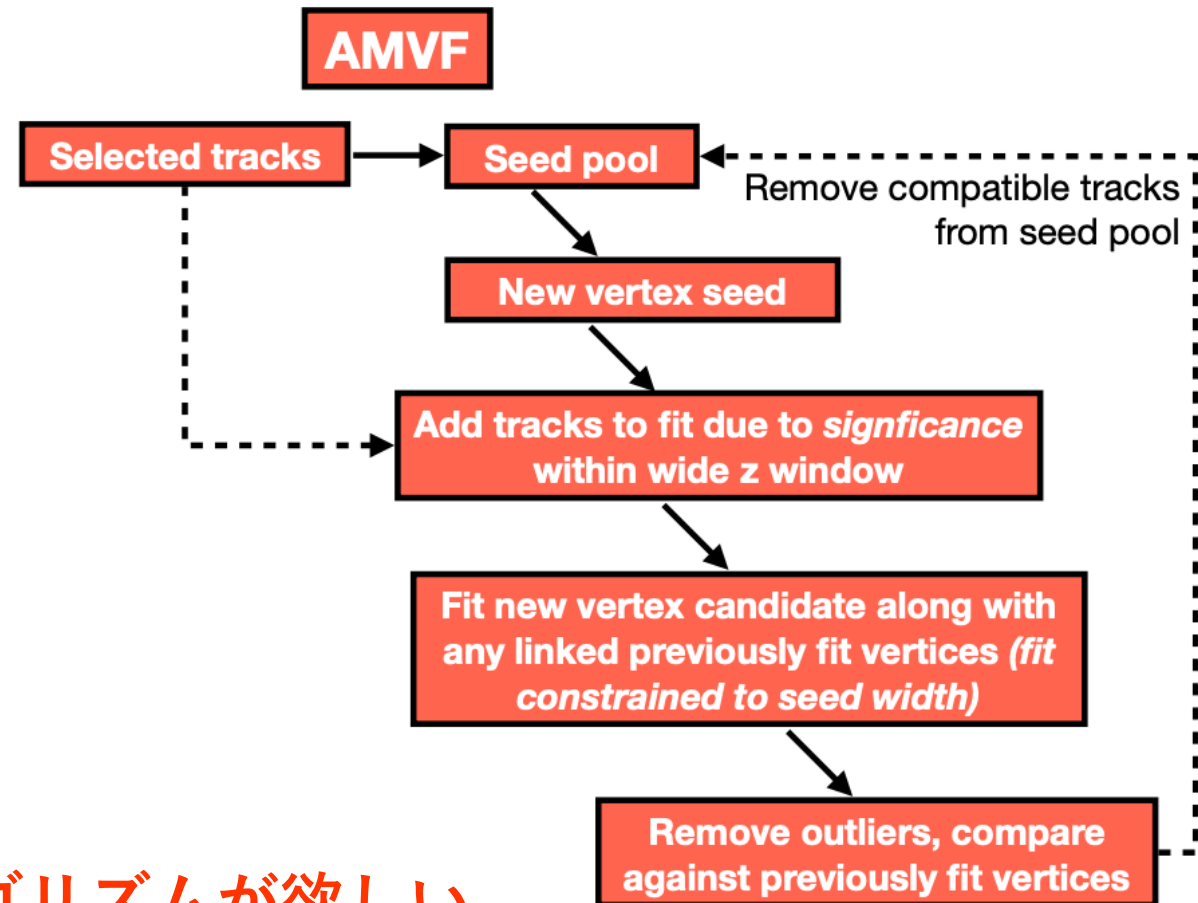
トラックを選別し、  
以下の手順を反復的に実行する

- Vertexのシード位置を選択
- Vertexのフィット
- 不適合トラックの除去

反復が進むにつれ、適合性の低い  
トラックを除去し、高いトラックの  
重みを強くしていく

性能は高いが時間がかかる

→ **トリガーで使える高速な再構成アルゴリズムが欲しい**



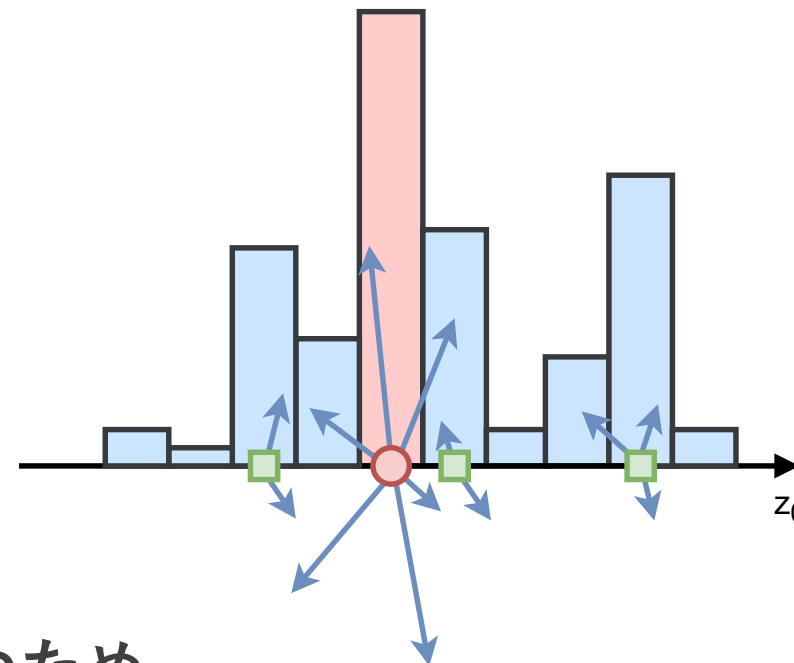
# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

# PVの高速再構成手法

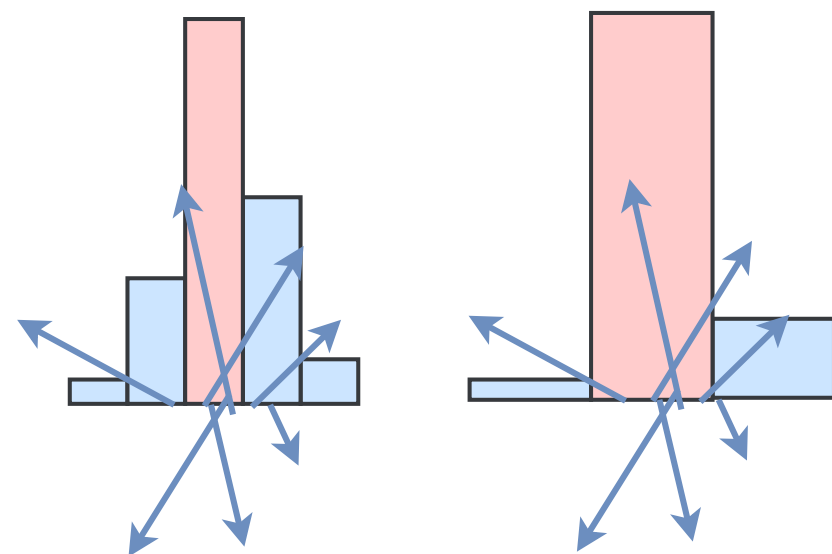
シンプルで高速な方法:

- ヒストグラムのbin幅を設定する
- トラックの横運動運動量 $p_T$ で足し合わせる
- 最も高いbinにPVが入っているとす



## bin幅と再構成精度

トラックは有限の分解能を持つため、**bin幅によって精度が変わる**

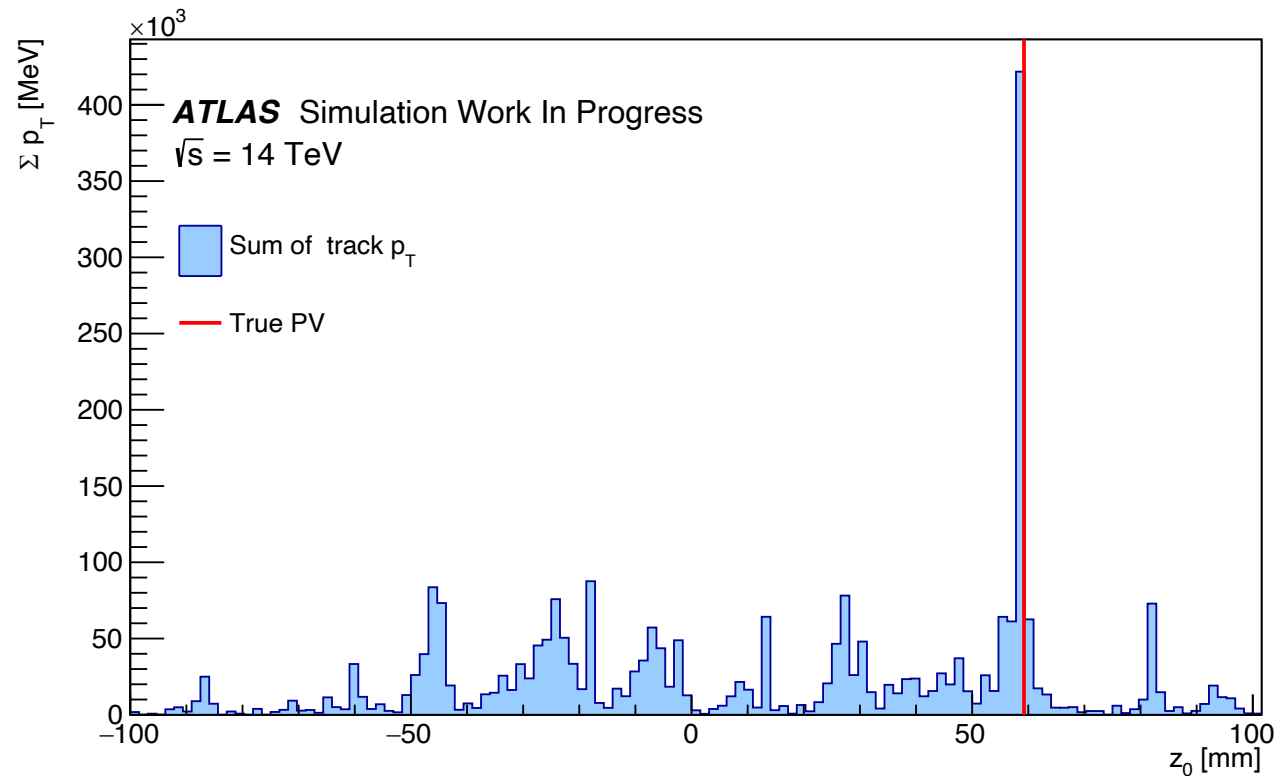


	トラック	分解能	特徴
広いbin	多い	悪い	再構成しやすいが パイルアップが混ざりやすい
狭いbin	少ない	良い	PVからのトラックを取り逃す 可能性があり、再構成が難しい

# 再構成結果

あるイベントでの成功例

- パイルアップ<sup>200</sup>のMCシミュレーションデータ
  - -200~200mmを256分割し、 $p_T$ で足し合せ (図は $\pm 100$ mmで拡大)
- 最も高いbinの中にTruthのPVが入っている



出力:

57.8125 ~ 59.375 mm

Truth PV:

59.235 mm

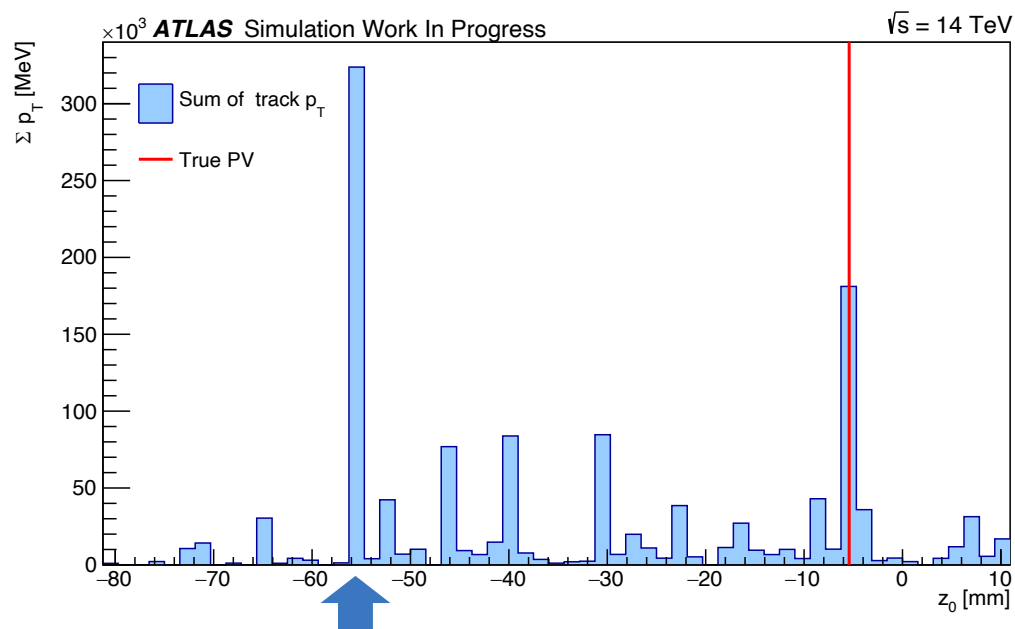
# この手法の問題点

非常に大きな $p_T$ を持ったトラックの影響が大きい

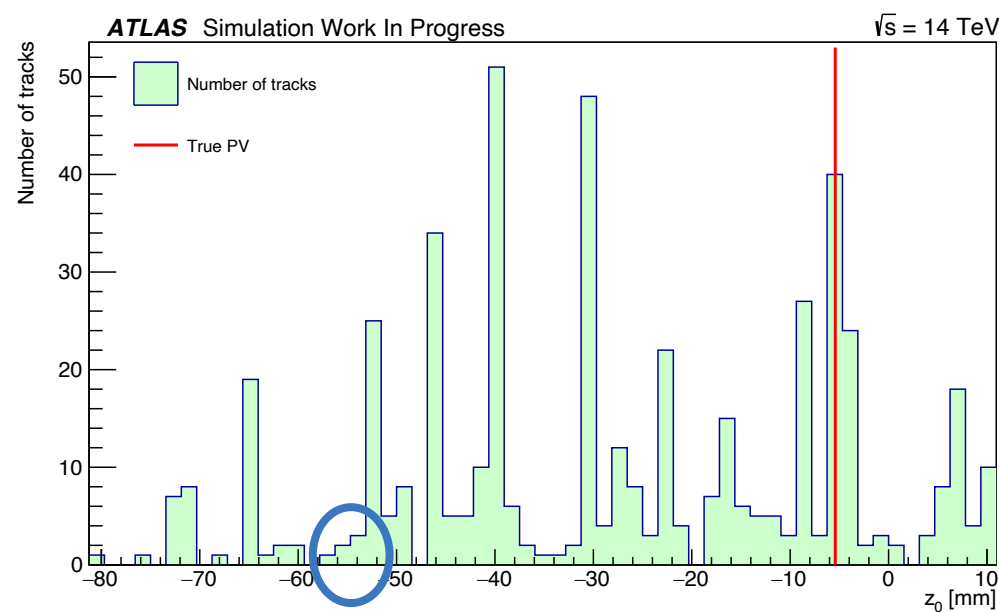
これは、再構成に失敗して大きな $p_T$ を持ってしまったトラックと考えられる

$p_T$ の合計

トラック数



再構成されたPVの位置



この位置はほとんどトラックがない

トラックの精度など、 $p_T$ 以外の情報を考慮したアルゴリズムを開発する必要がある

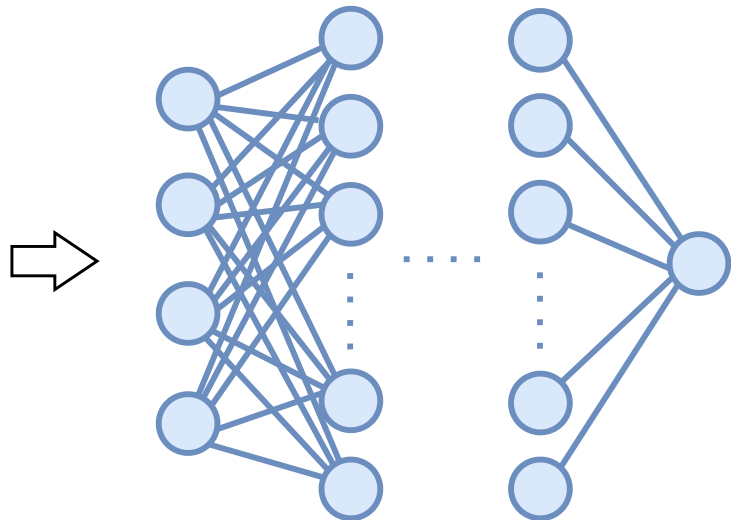
# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

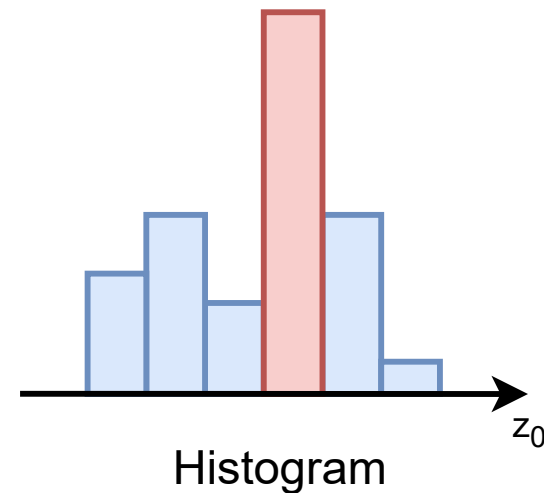
# 機械学習による再構成手法

学習モデル:

Track features  
( $p_T, \eta, d_0, \chi^2, \sigma_{d0}, \sigma_{z0}$ )



Weight  
⇒



PV

入力: 6ノード

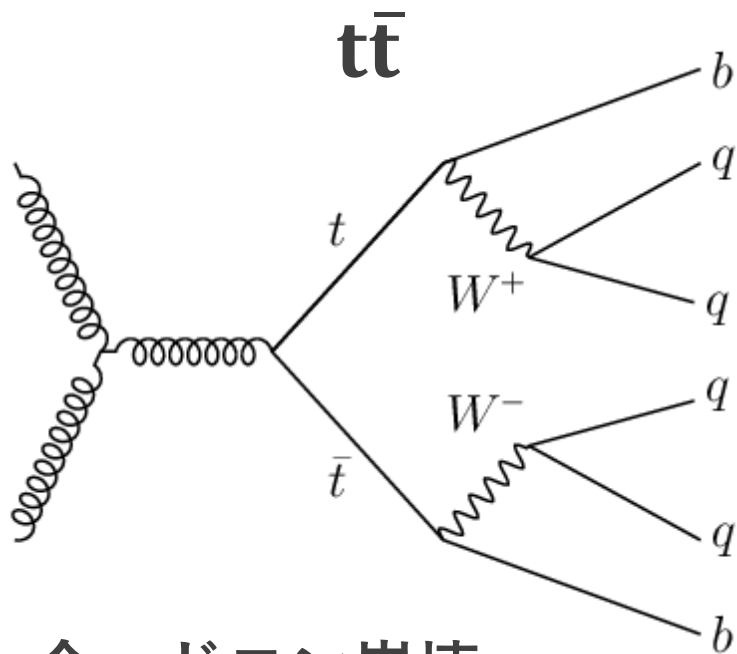
MLP: 5層  
活性化関数: ReLU

Softmaxを利用して  
最大値の位置を出力

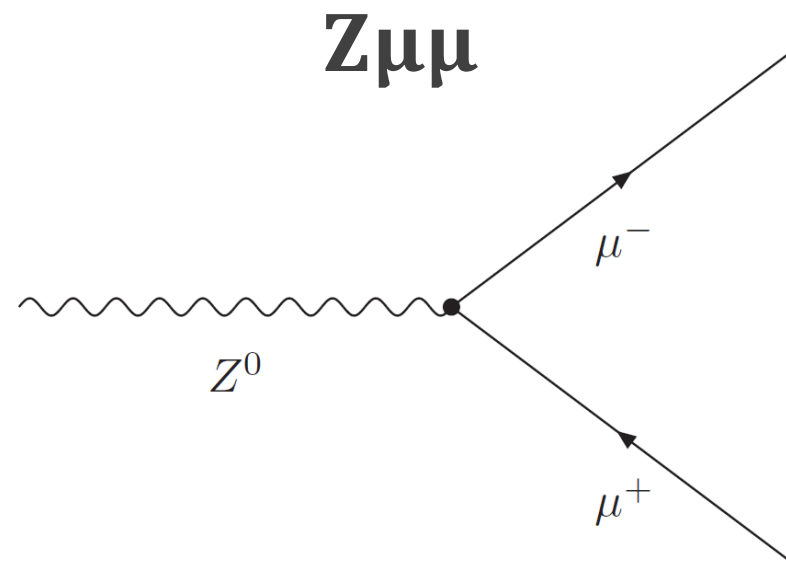
- 損失関数: Truthとヒストグラムの最大値の位置との平均二乗誤差 (MSE)
- データセット: パイルアップ<sup>o</sup>200, MCシミュレーション, 1万イベント
- ヒストグラムのbinを変えて、それぞれ別々に学習

# データセット

2つのデータセットを用いて学習



- 全ハドロン崩壊
- トラック数が多い



- $Z$ がミューオン対に崩壊
- トラック数が少ない

$t\bar{t}$ は比較的PVの再構成がしやすい例、 $Z\mu\mu$ は難しい例として採用



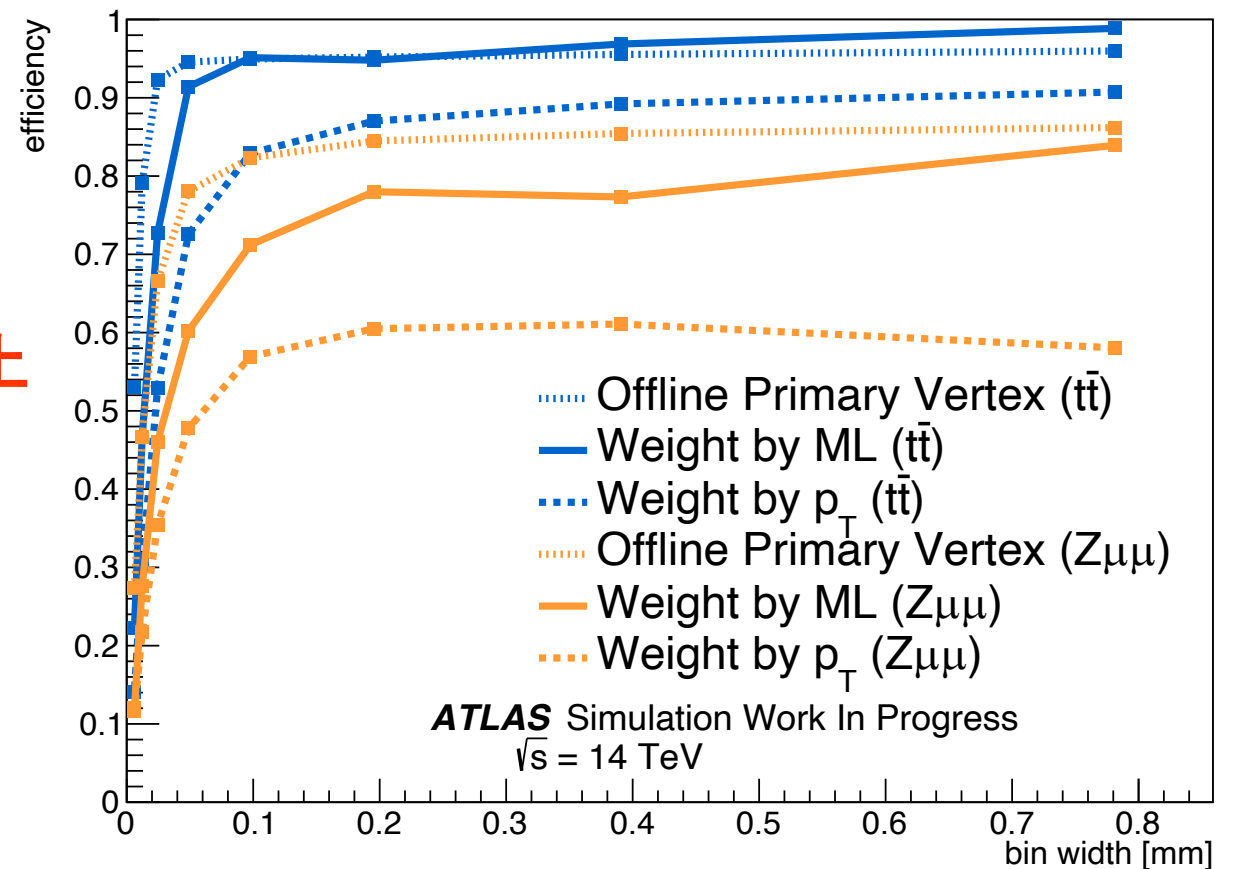
# 性能評価

## Efficiencyでの評価:

- 出力したPVとTruthのPVとの差が横軸のbin幅以内のイベントの割合
- 参考として現行のオフラインアルゴリズムとも比較

トラックの誤差は0.1mm程度  
→ PVも0.1mmを目安に考える

- どちらのデータセットでも  
**機械学習によってEfficiencyが向上**
- $t\bar{t}$ 
  - オフラインに近い性能を示す
- $Z\mu\mu$ 
  - 機械学習によって大幅に向上
  - まだ伸び代がある



# CONTENTS

1. イントロダクション
2. Primary Vertex再構成
3. シンプルな手法
4. 機械学習を用いた手法
5. まとめ

# まとめ

- 高輝度化に向けてトリガーシステムでのPV再構成が有用
- 2つの手法を開発し、PV再構成アルゴリズムの性能を評価した
  - シンプルな手法に比べ、機械学習を活用することで大幅に性能が向上した
  - データセットによって再構成の性能が変化

## 今後の展望

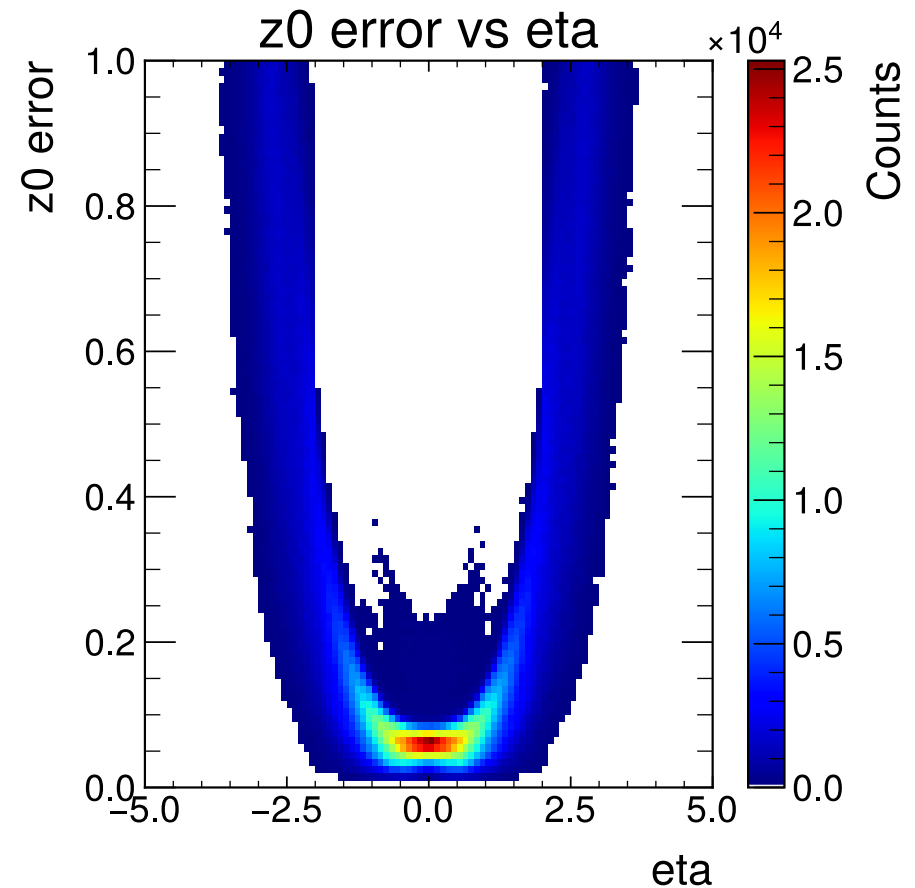
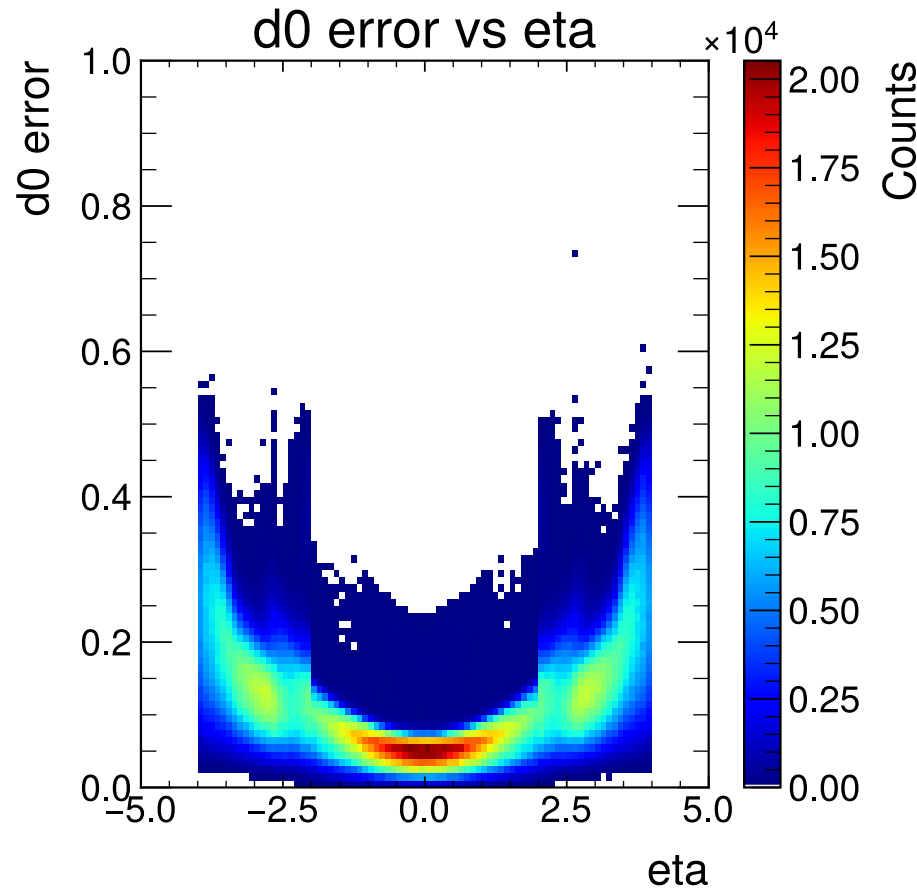
- データ依存が少ない汎用的な機械学習モデルの開発・評価
- 実データ (Run3) での性能評価
- トリガーシステムに適用可能な再構成速度の検討
- 再構成したPVを用いた場合の後段トリガーの性能評価

# Back Up

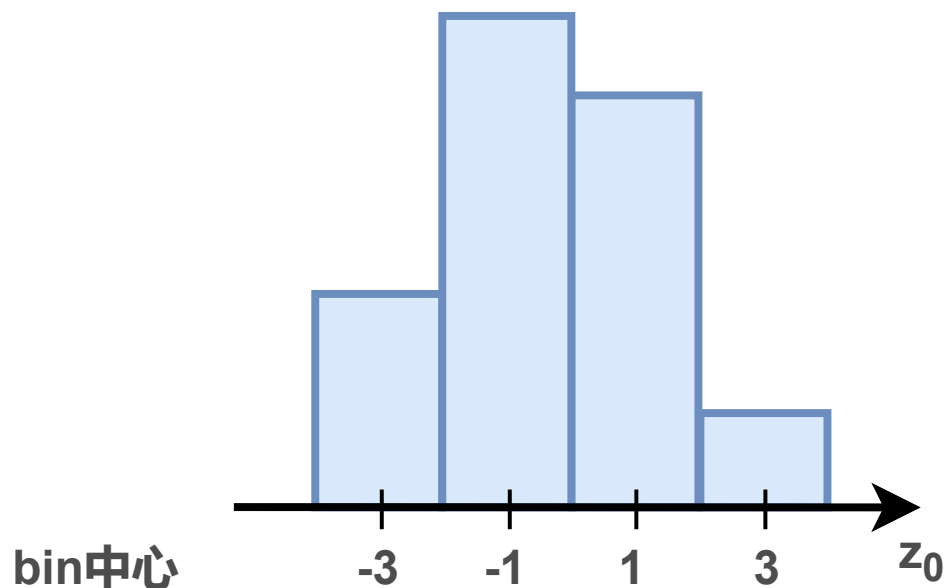
---

# トラックの分布

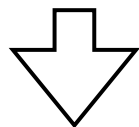
- トラックの $z_0$ の誤差は0.1mm~0.2mmに分布
- 高い $\eta$ では誤差が大きい



# ピーク位置の推定



値



重み

0 1  $10^{-87}$  0

$$\text{Softmax}(u_k) = \frac{\exp(u_k/T)}{\sum_{j=1}^K \exp(u_j/T)}$$

Softmax:

- 入力の値を0~1にスケール
- 合計値は1
- 温度が0に近づくほど急になる

今回は、予測されるbinはほとんど-1

ヒストグラムの最大値を出力するのとほとんど等価と考えて良い

# ハイパーパラメータ

- ニューラルネットワーク
  - インプット6, アウトプット1
  - 隠れ層は5層
    - 32, 128, 256, 128, 32
- 学習率: 0.0001
- バッチサイズ: 32
- Epoch数
  - 最大: 100
  - 5回連続で改善しない場合はEarly stop
  - 実際は20程度で学習が完了