MVA MET calibration with NN

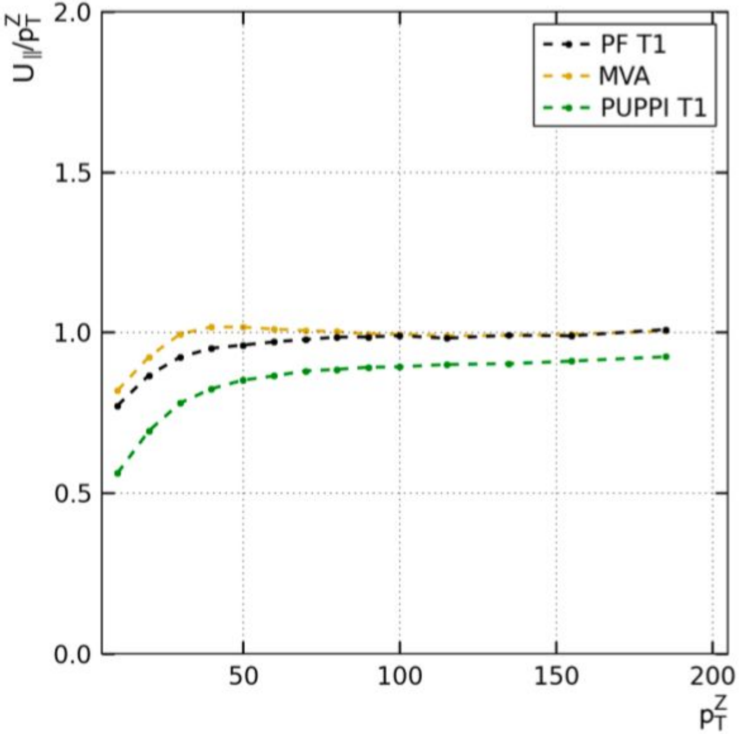
Tanja Kopf, Stefan Wunsch, Roger Wolf, Günter Quast
tanja.kopf@cern.ch, stefan.wunsch@cern.ch

KIT ETP / CERN SFT

Former work in our group

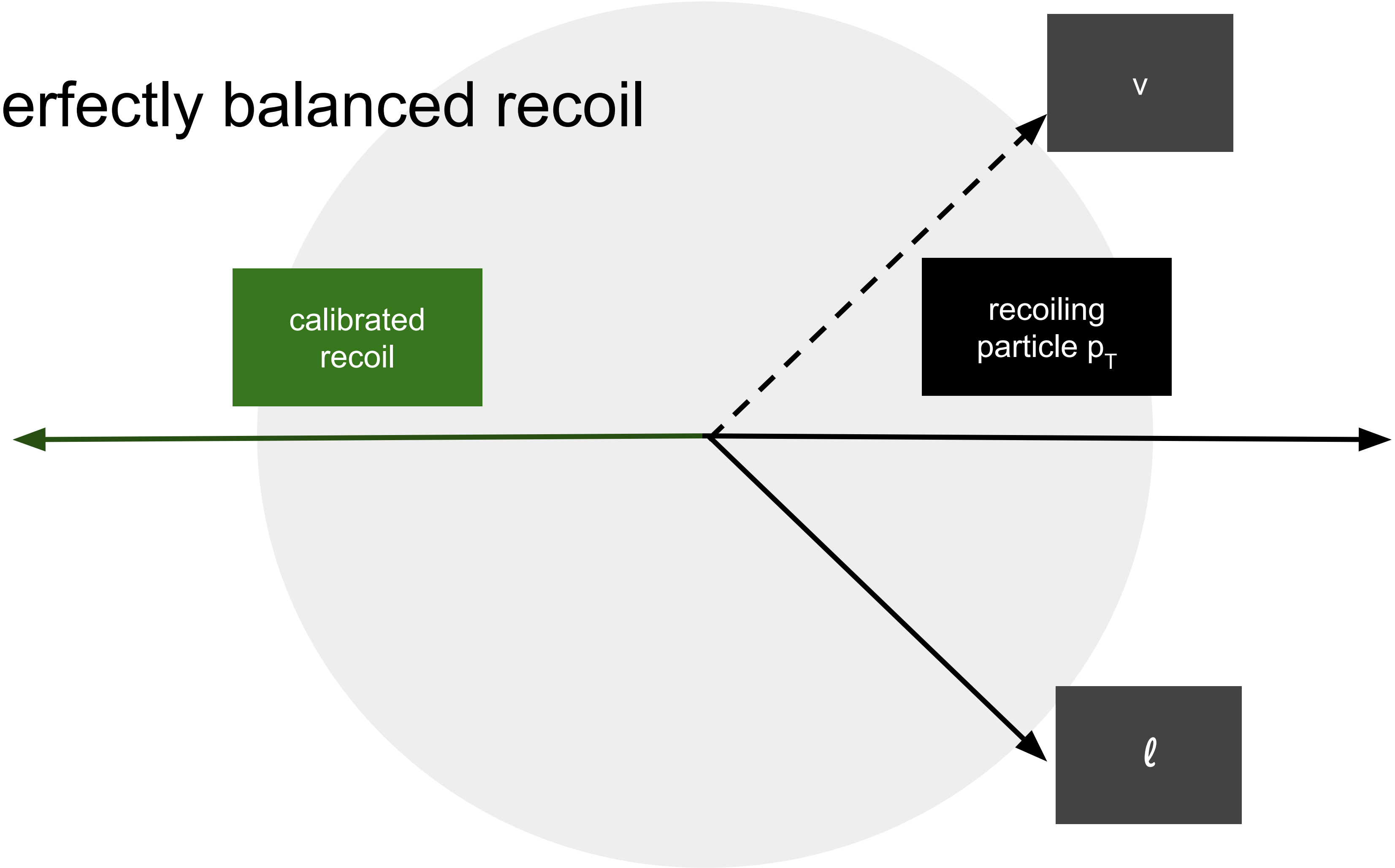
- in our group have been MVA MET approaches by Raphael Friese with BDTs
- our goal is to improve this approach with neural networks (NN)
- performance of NN exceeds performance of BDT

In the right handed response plot MVA represents the BDT result

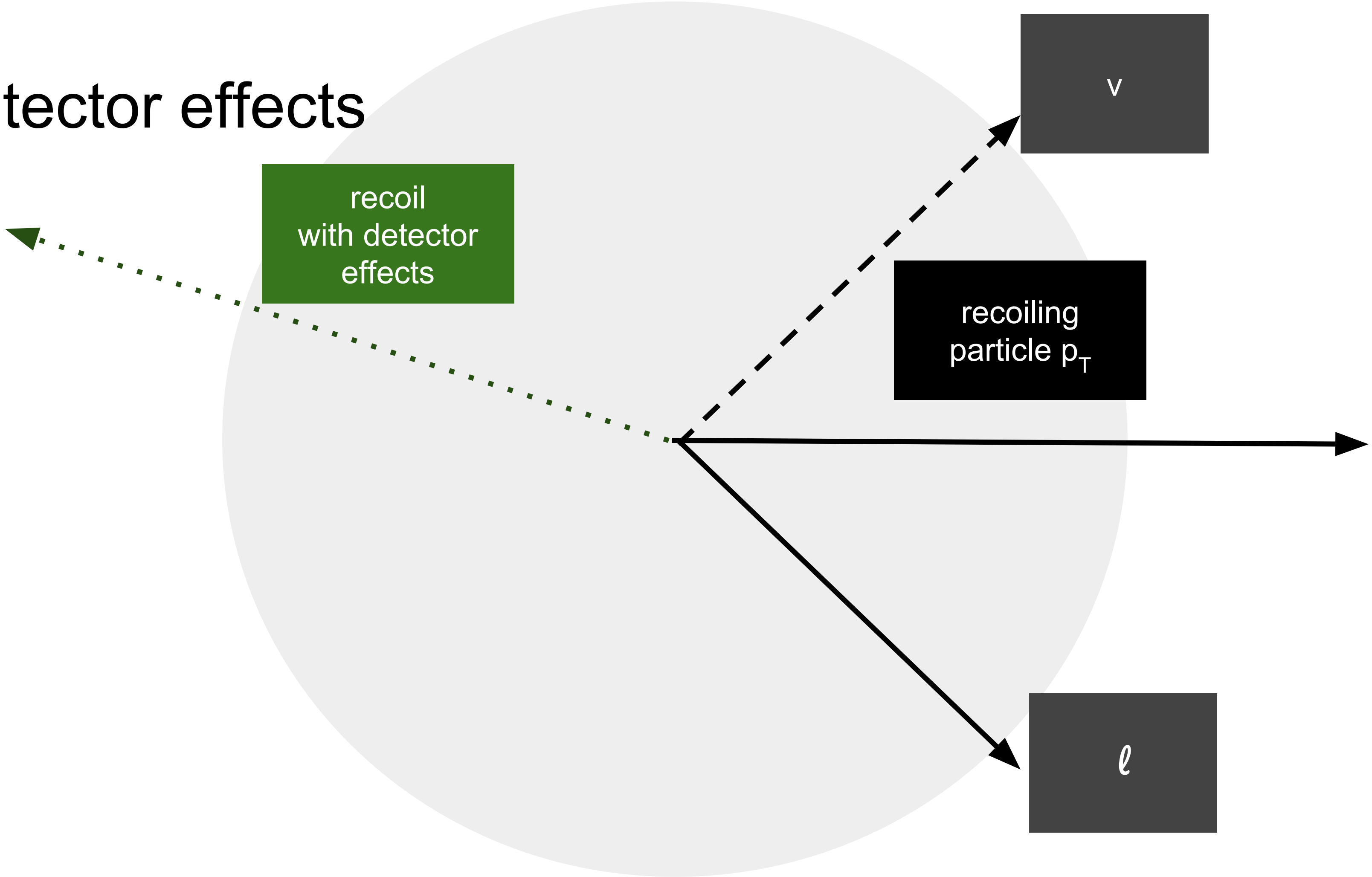


Plot: https://publikationen.bibliothek.kit.edu/1000070623

Perfectly balanced recoil



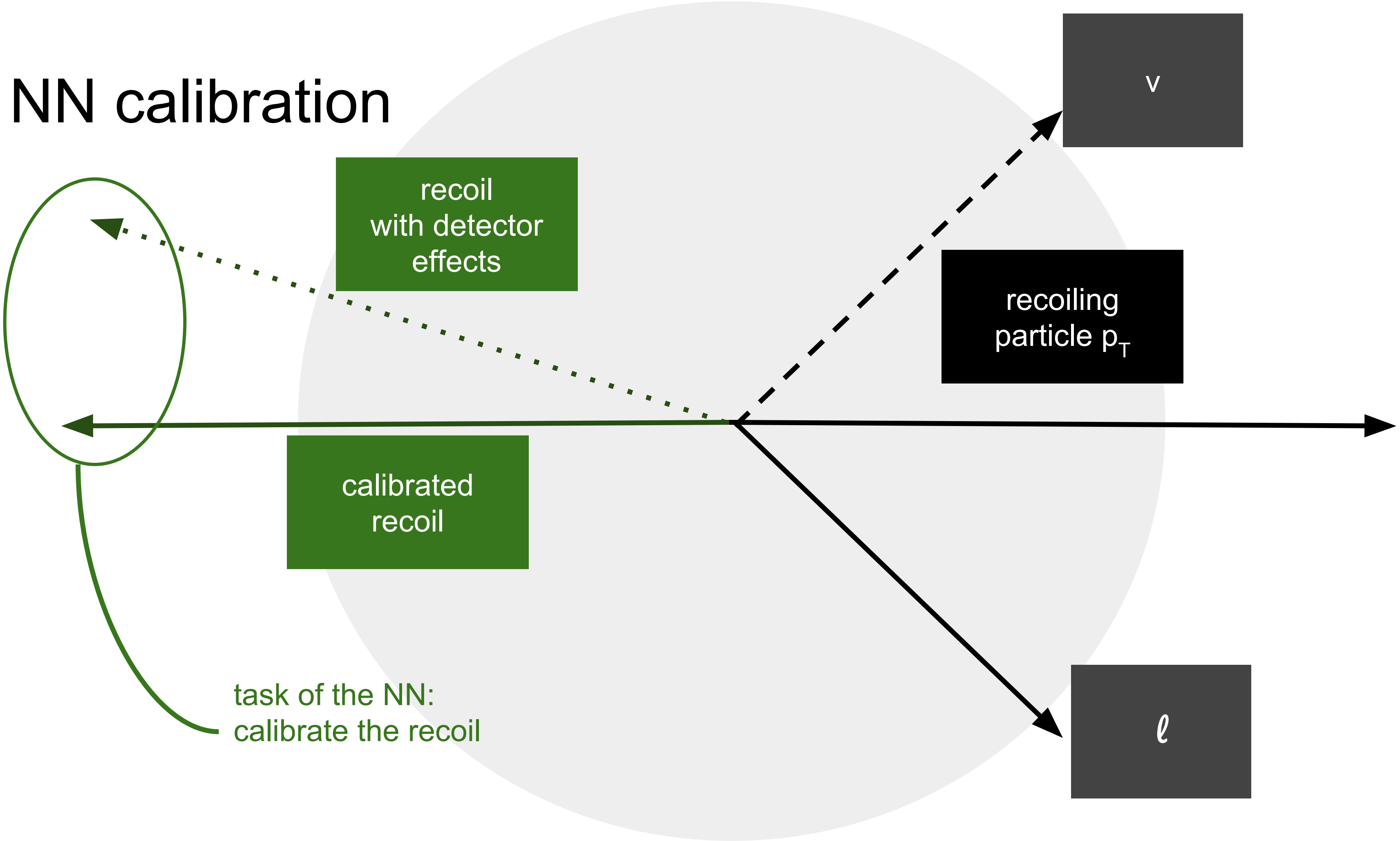
Detector effects



recoil
with detector
effects

recoiling
particle p_T

NN calibration



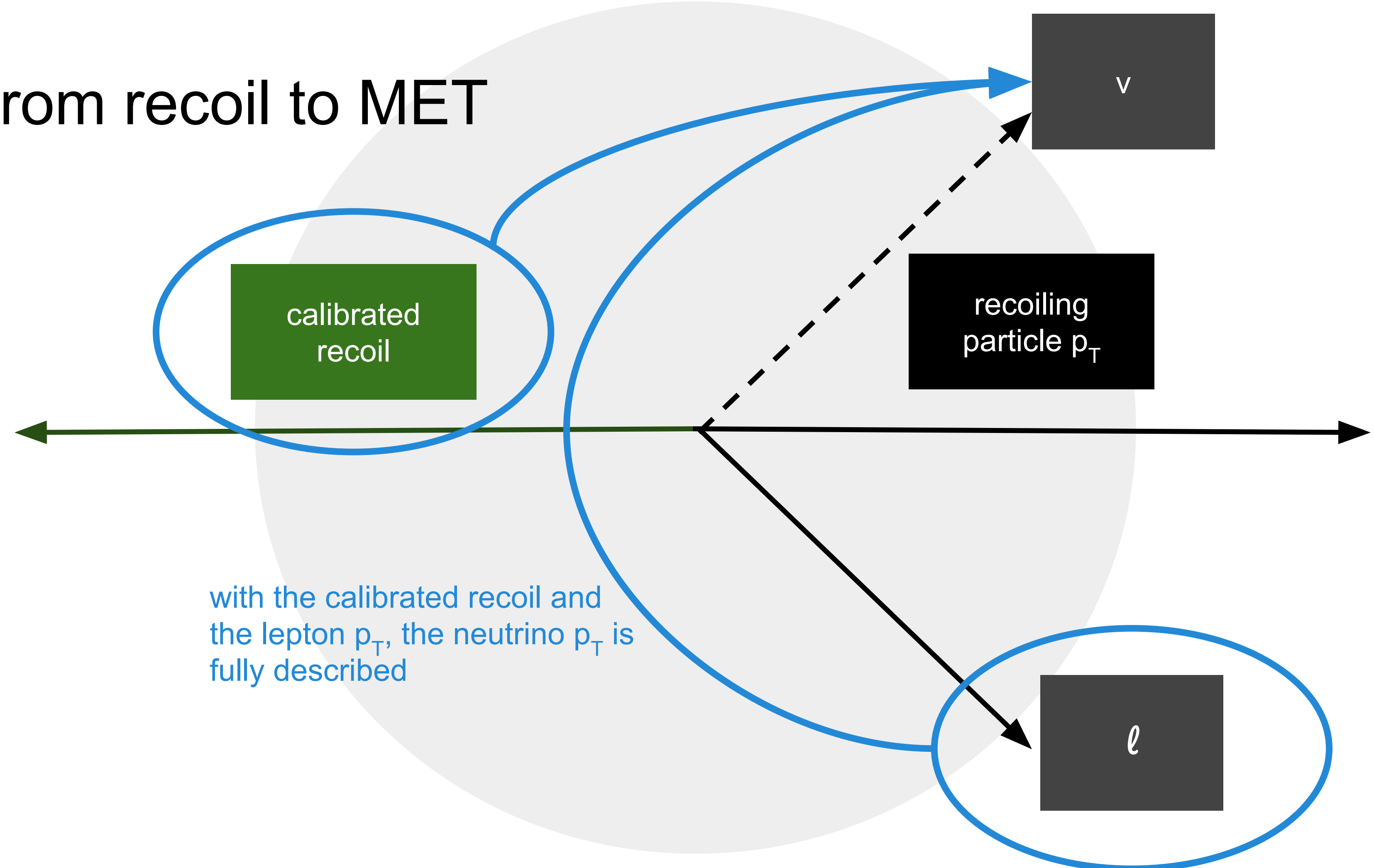
recoil with detector effects

recoiling particle p_T

calibrated recoil

task of the NN: calibrate the recoil

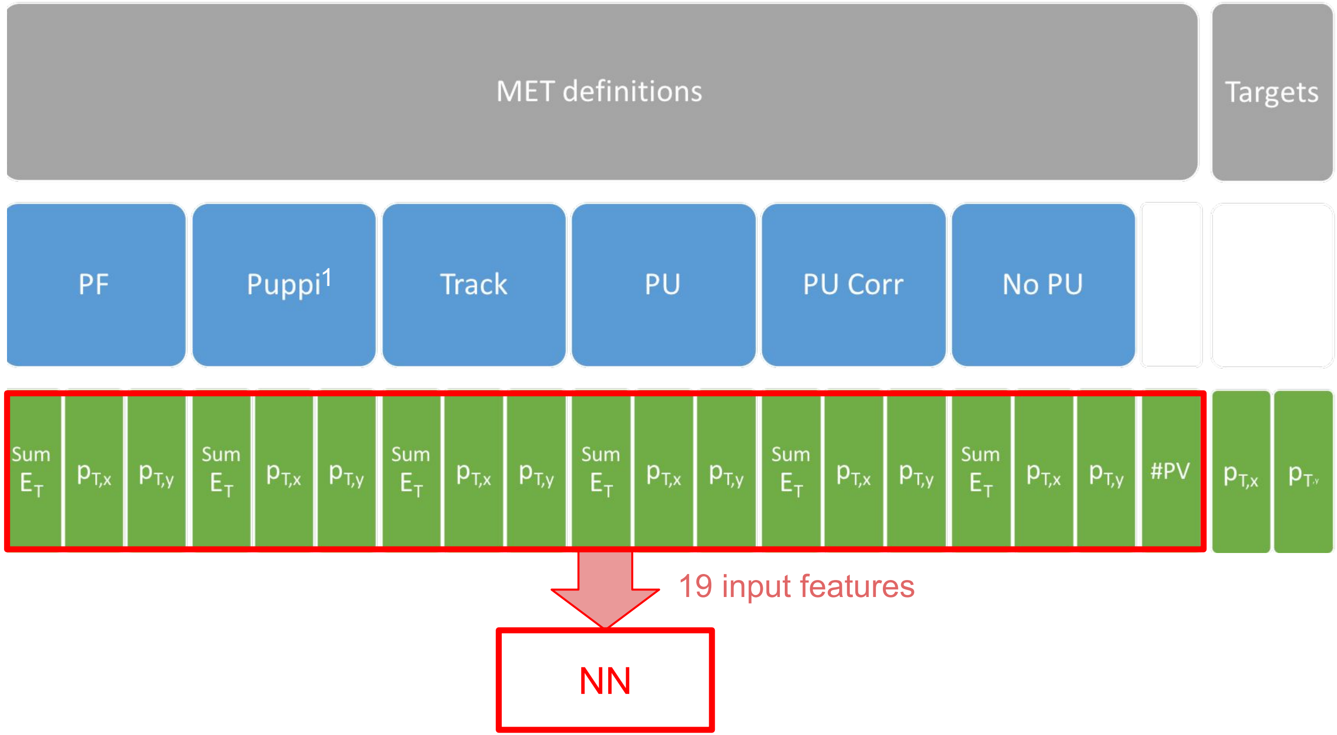
From recoil to MET



with the calibrated recoil and the lepton p_T, the neutrino p_T is fully described

MVA approach

Input and output features



MET definitions							Targets	
PF	Puppi[1]	Track	PU	PU Corr	No PU			
Sum E_T, $p_{T,x}$, $p_{T,y}$	Sum E_T, $p_{T,x}$, $p_{T,y}$	Sum E_T, $p_{T,x}$, $p_{T,y}$	Sum E_T, $p_{T,x}$, $p_{T,y}$	Sum E_T, $p_{T,x}$, $p_{T,y}$	Sum E_T, $p_{T,x}$, $p_{T,y}$	#PV	$p_{T,x}$	$p_{T,y}$

19 input features

NN

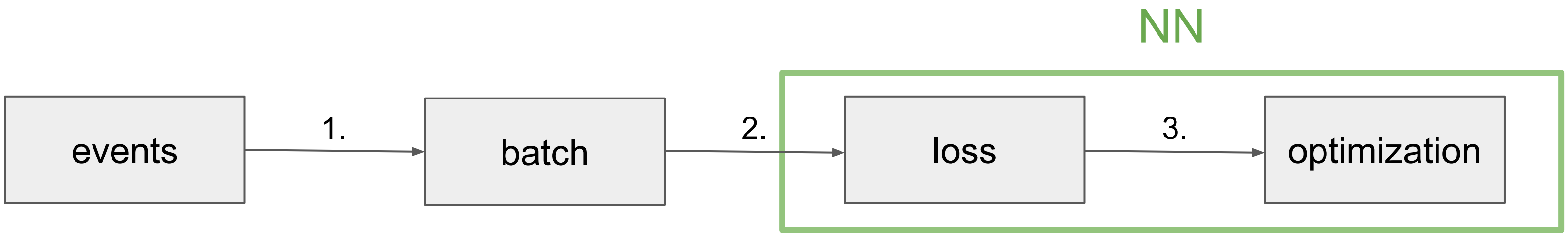
Legend:

Sum E_T: Sum of absolute values of transverse momentum of all particles used in MET definition

($p_{T,x}$; $p_{T,y}$): Transverse momentum in cartesian coordinates

[1] Pileup Per Particle Identification (PUPPI)

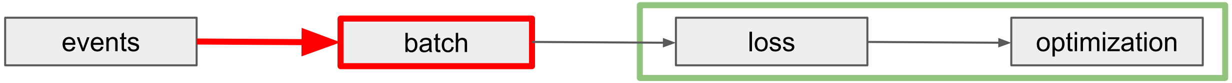
NN workflow for one gradient step



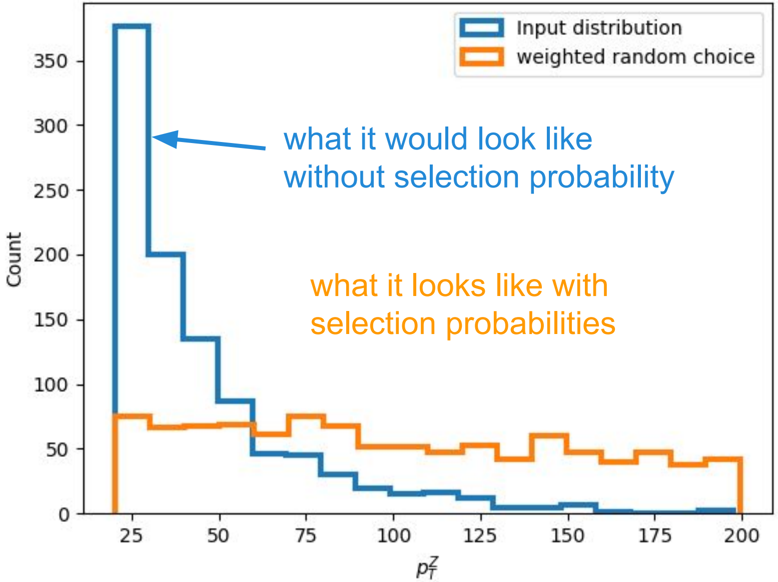
The general NN workflow consists of 3 steps

1. Subsetting the events to a sample on which the NN trains
2. Formulating a loss function to calculate the loss between the prediction and desired output
3. Minimizing the loss

→ For the NN MET approach the **batch selection** and **loss** are individually tailored to the problem



Batch selection



what it would look like without selection probability

what it looks like with selection probabilities

Problem:

p_T-distribution has strong decrease over p_T

→ It's likely that bins with high p_T are empty
→ Reweighting doesn't work

Solution:

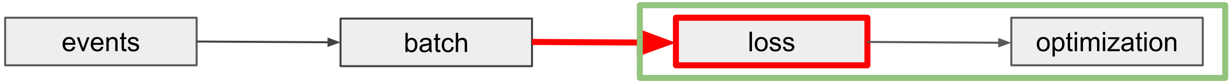
1. Fit Crystal ball function[1] $g(p_T)$ to p_T-distribution

2. Randomly choose subsets of data as batches with **probability p** associated with each event

$$p = \frac{1}{g(p_T)}$$

→ **Get in p_T uniformly distributed batches**

[1] https://arxiv.org/pdf/1002.1850.pdf



Loss function

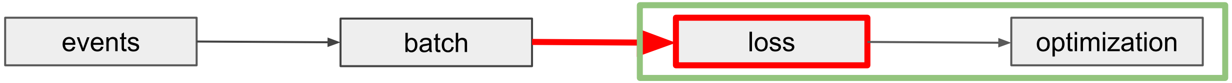
Goal of the NN calibration: best response

Problem:

- Distribution of PF/Puppi response is asymmetric around 1
- Let the NN handle the asymmetry → part of loss

Loss addresses two goals:

- Minimize deviation from response=1
- Minimize asymmetry of response in p_T and #PV bins

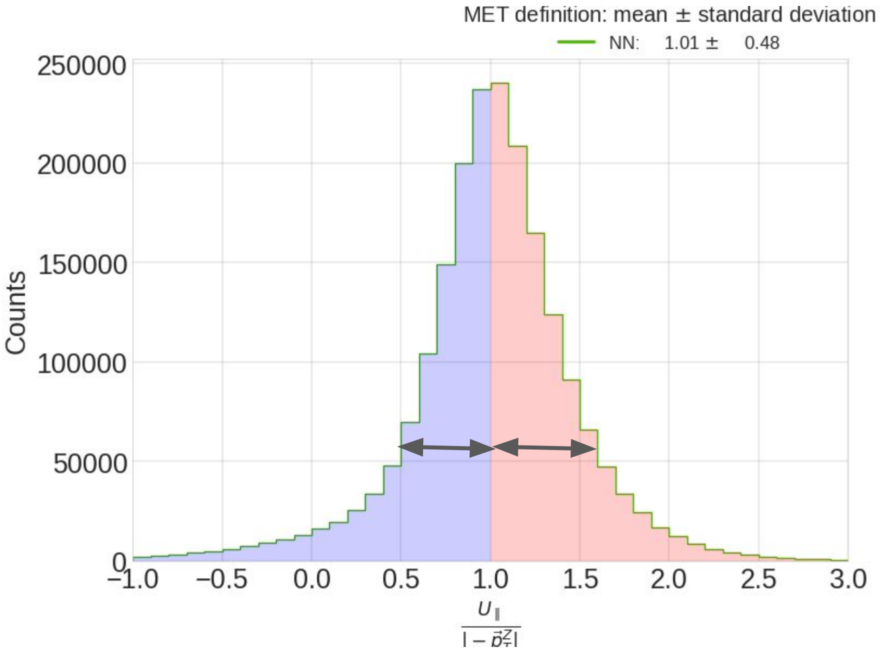


Minimize deviation of response inclusive

Loss l_R for minimize error from response = 1 in batch with size N_B:

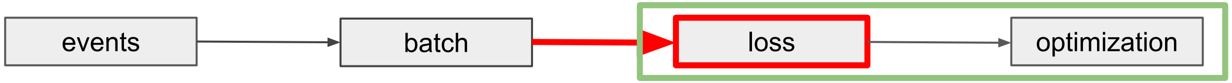
$$l_R = \sum_{i=1}^{N_B} (R-1)^2$$

$$R = \frac{U_\parallel}{-p_T^Z}$$



→ Penalty on response

→ **Minimizes the deviation** inclusive over the whole batch



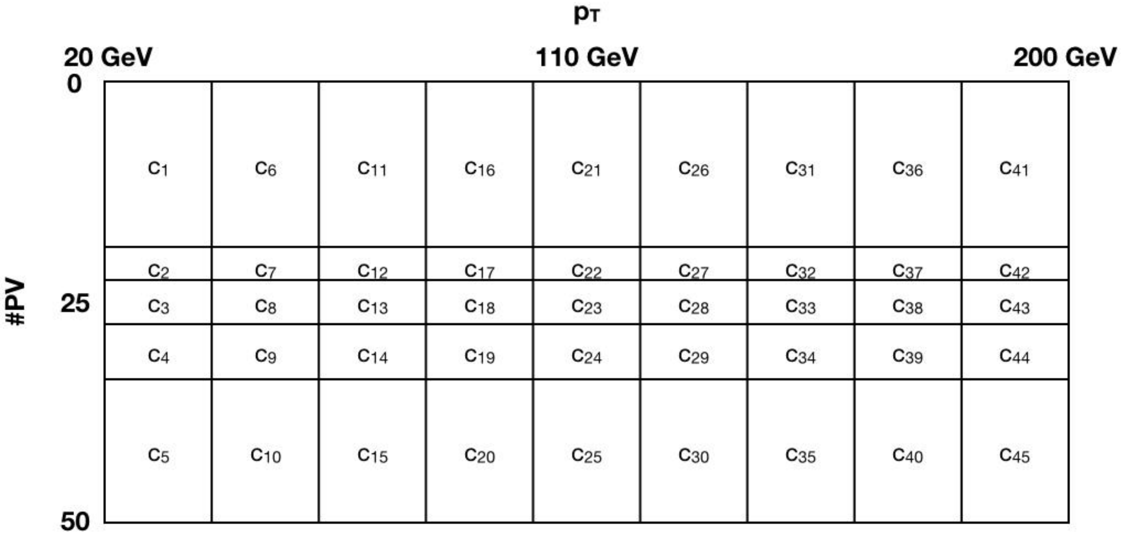
events → batch → loss → optimization

Minimize asymmetric distribution

1. Create 2D binning in batch over p_T and #PV → ensure to have same population in each bin

   a. For p_T the batches are uniformly distributed → uniform binning

   b. For #PV take percentiles with each 20 % of the batch



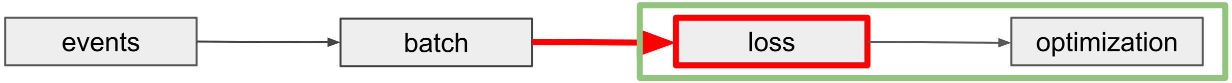
C1	C6	C11	C16	C21	C26	C31	C36	C41
C2	C7	C12	C17	C22	C27	C32	C37	C42
C3	C8	C13	C18	C23	C28	C33	C38	C43
C4	C9	C14	C19	C24	C29	C34	C39	C44
C5	C10	C15	C20	C25	C30	C35	C40	C45

Number of p_T bins:

$$N_{p_T} = 9$$

Number of #PV bins:

$$N_{PV} = 5$$



events → batch → loss → optimization

2D binning in loss

2. Each bin results in its own cost value c_i

3. Sum up over all costs c_i → loss l

$$l = \sum_{i=1}^{N_{p_T} \cdot N_{PV}} c_i$$

$$c_i = \boxed{\sum_j^{N_B} b_{ij} \cdot (R_j - 1)^2} + \boxed{s \cdot \left(\sum_j^{N_B} b_{ij} \cdot max(0, R_j - 1) - \sum_j^{N_B} b_{ij} \cdot max(0, 1 - R_j) \right)^2}$$

loss l_R for minimize error from response = 1

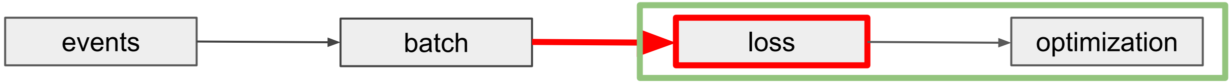
loss l_A minimizes the asymmetry of the distribution in each bin

$$R = \frac{U_\parallel}{p_T^Z}$$

$$s \in \mathbb{R}, \text{ global scale factor}$$

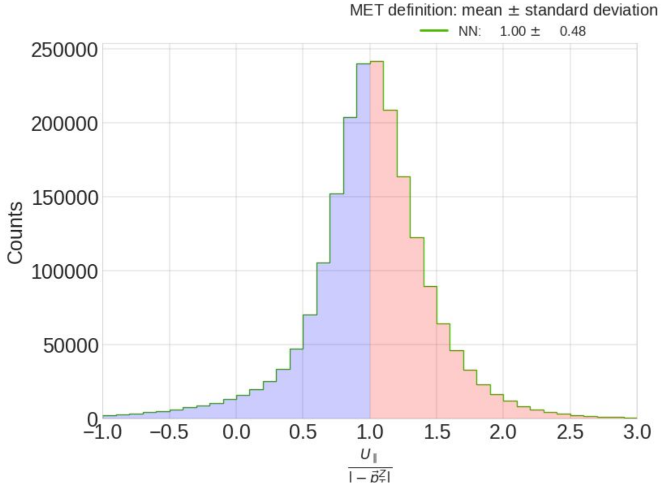
$$b_i \in \{0, 1\}$$

→ **Minimizes the asymmetry** of the distribution in each bin



Cost values

cost value for each bin:

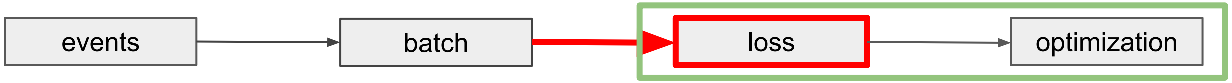


$$c_i = \sum_{j}^{N_B} b_{ij} \cdot (R_j - 1)^2 + s \cdot \left(\boxed{\sum_{j}^{N_B} b_{ij} \cdot max(0, R_j - 1)} - \boxed{\sum_{j}^{N_B} b_{ij} \cdot max(0, 1 - R_j)} \right)^2$$

$$R = \frac{U_\parallel}{p_T^Z}$$

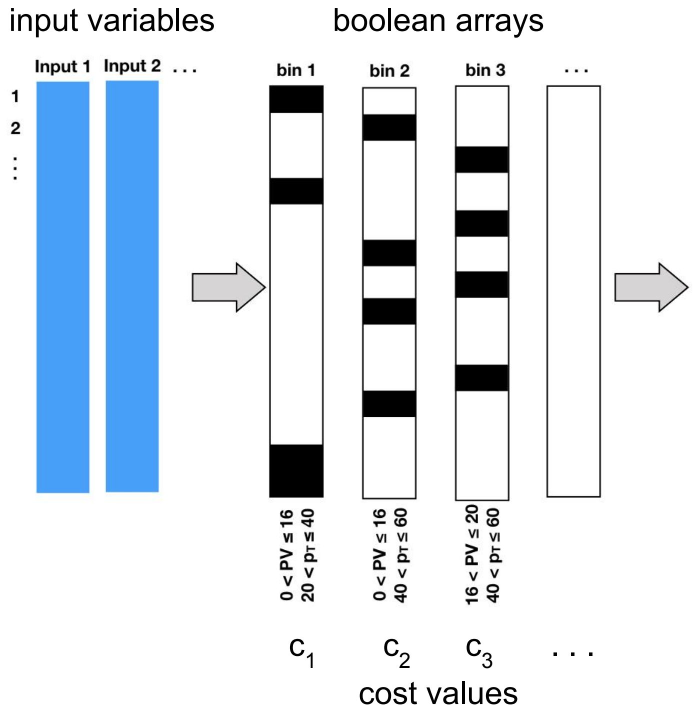
$s \in \mathbb{R}$, global scale factor

$b_i \in \{0, 1\}$

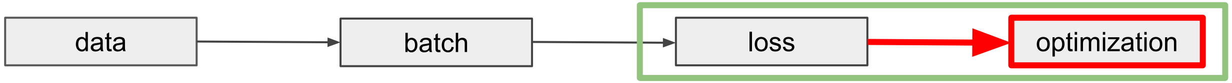


Cost values

Boolean arrays are binning the batches while ensuring the bins are all equally well populated



$$l = \sum_{i=1}^{N_{p_T} \cdot N_{PV}} c_i$$



data → batch → loss → optimization

Minimizing loss

- Each gradient step results in one loss value

$$l = \sum_{i=1}^{N_{p_T} \cdot N_{PV}} c_i$$

- Optimization of the NN:
  - Calculate gradients of the loss with respect of the NN weights
  - Apply the gradients to the weights
  - Minimize loss along gradients with optimizer algorithms
    → in this case: Adam

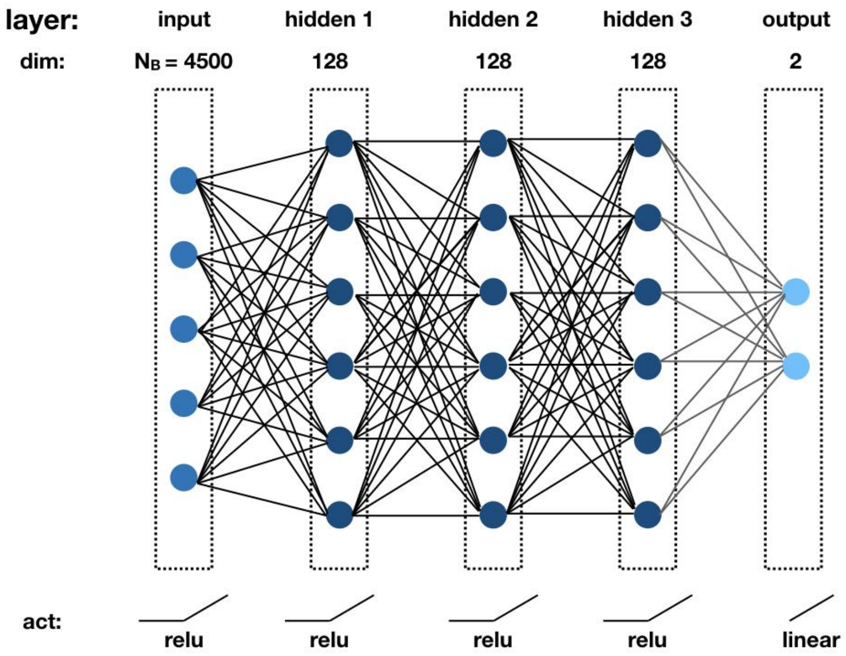
→ The NN will optimize its weights with respect to minimizing the loss

Application

- dataset settings:
  - MC Summer 17
  - Drell-Yan Z→ μμ
  - ~2 Mio events for training and application
  - 20 GeV ≤ p_T^Z ≤ 200 GeV
  - 0 ≤ number primary vertices (#PV) ≤ 50

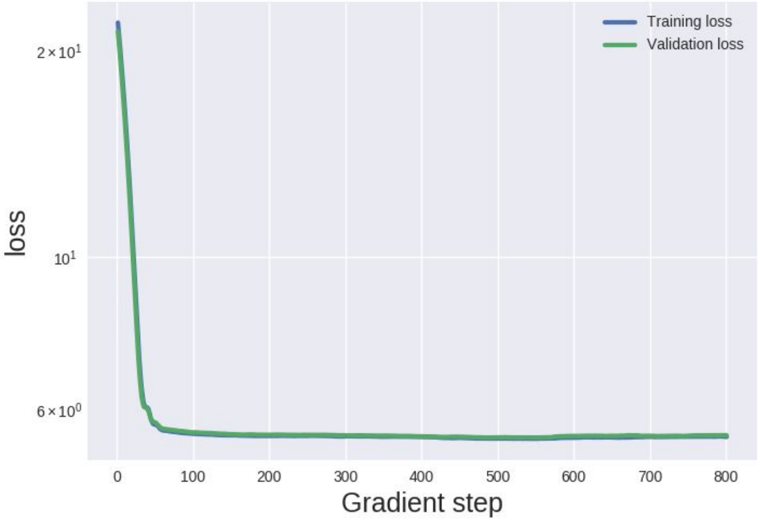
- plotting settings
  - color coding: Puppi, PF, NN

NN topology



Loss

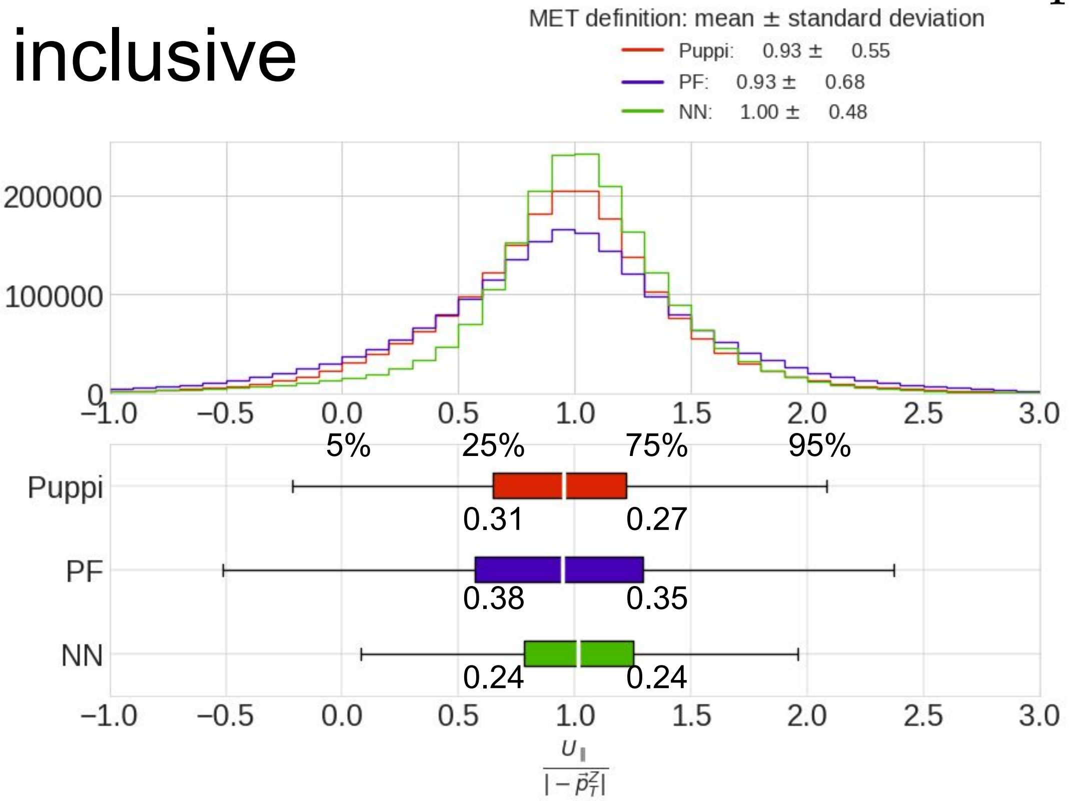
Convergence of loss function



Response

Response inclusive

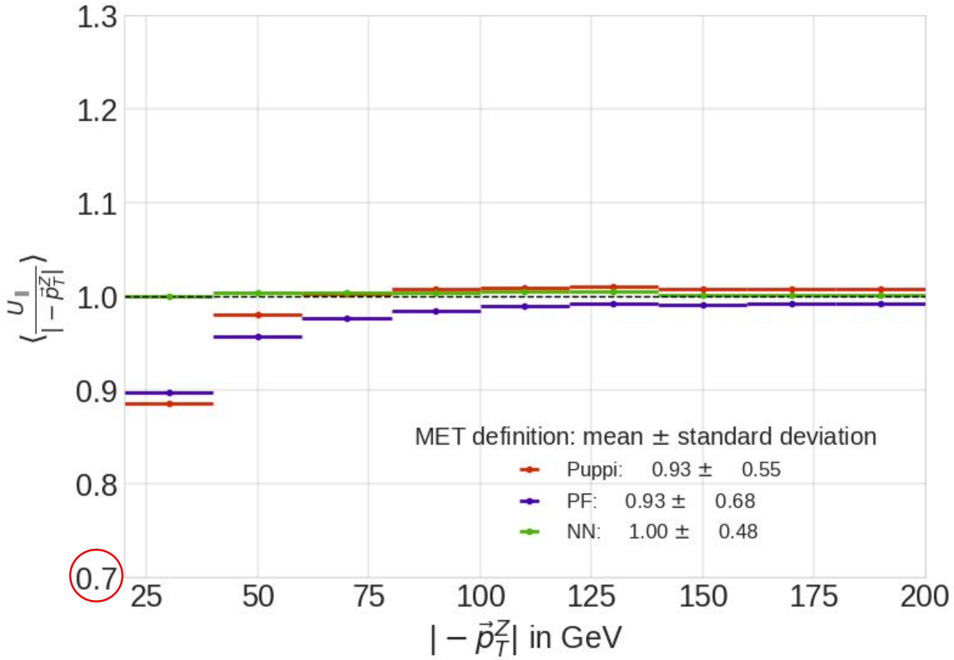
$$\text{Response} = \langle \frac{U_\parallel}{-p_T^Z} \rangle$$



→ The custom loss manages to **minimize the asymmetry of the distribution** while **optimizing the response** to be one in mean

$$\text{Response} = \langle \frac{U_{\parallel}}{-p_T^Z} \rangle$$

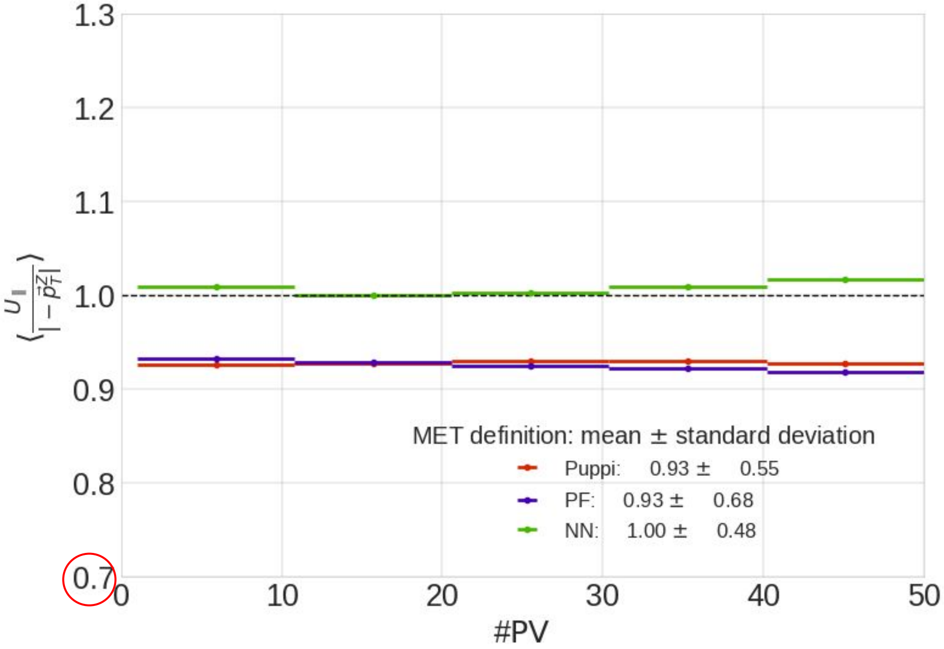
Response vs. p_T



- p_T binning in the loss results in p_T independent response
- NN is closest to 1 over the whole p_T range

$$\text{Response} = \langle \frac{U_{\|}}{-p_T^Z} \rangle$$

Response vs. #PV

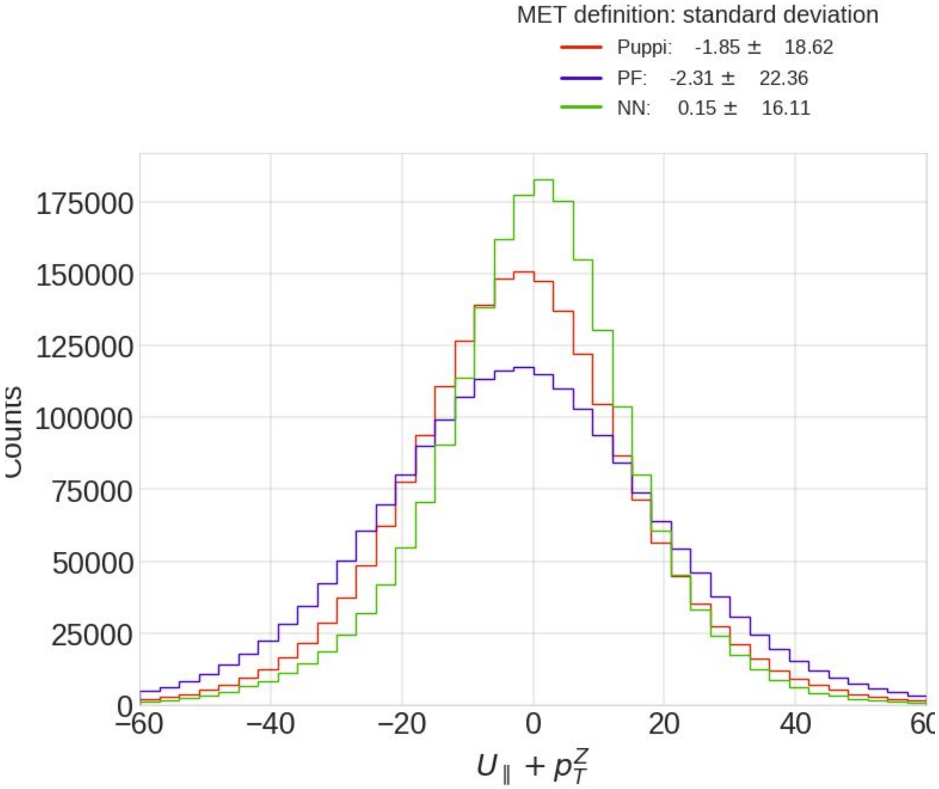


- #PV binning in loss results in minimal deviation of response over #PV range
- NN is closest to 1 over whole #PV range

Resolution parallel

$$\text{Resolution}_{\parallel} = \sigma\left(U_{\parallel} + p_T^Z\right)$$

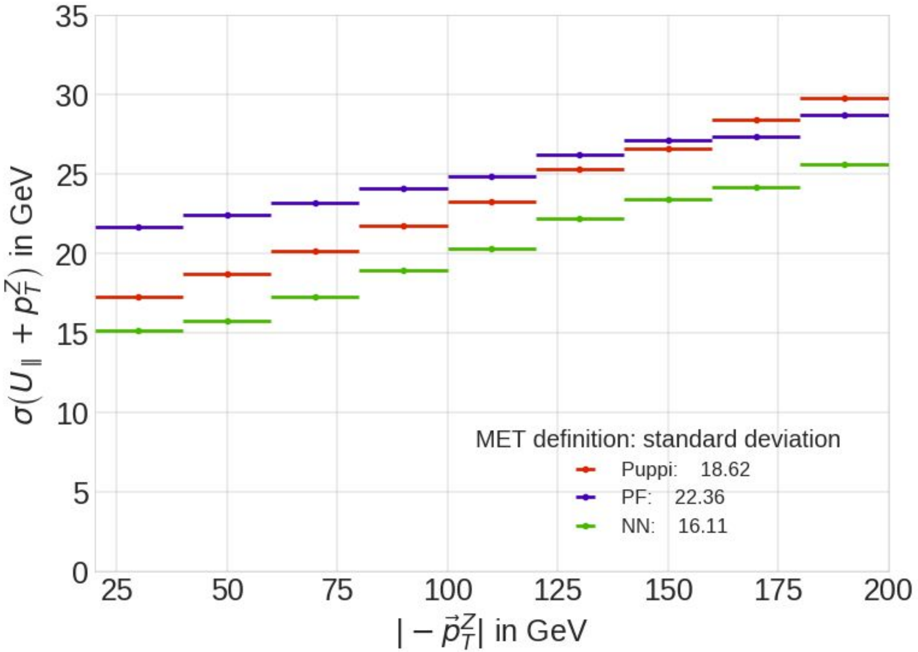
Resolution parallel: inclusive



- Distribution of resolution with small bias for NN
- NN has the best parallel resolution inclusive

$$\text{Resolution}_{||} = \sigma\left(U_{||} + p_T^Z\right)$$

resolution para vs. p_T

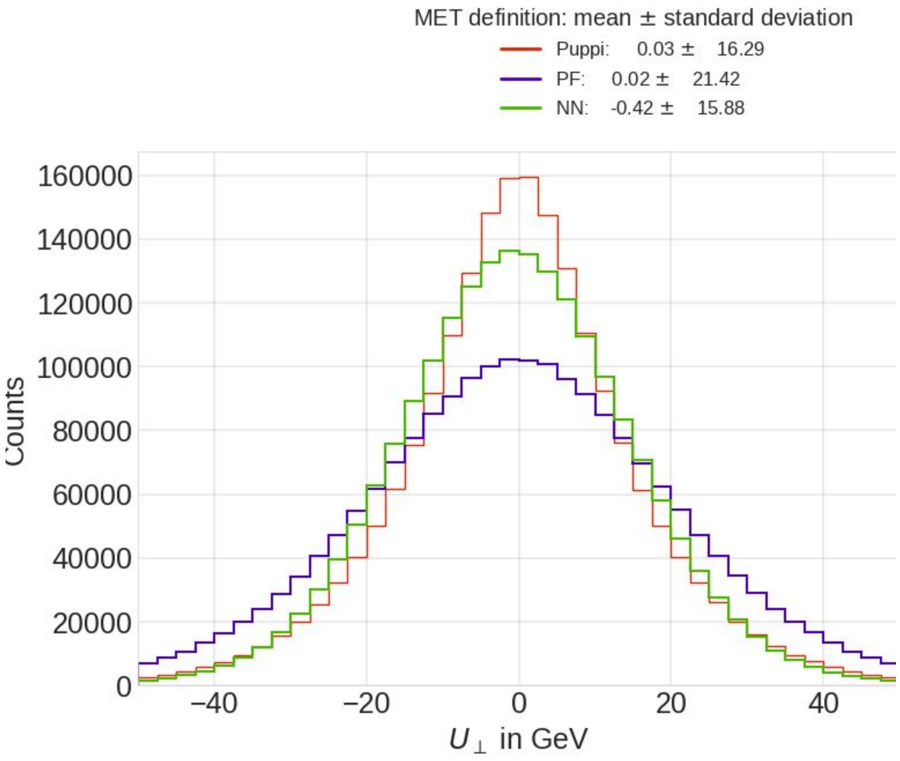


- Resolution of Puppi exceeds PF for large p_T values
- Minimum resolution for NN over p_T range

Resolution perpendicular

$$\text{Resolution}_{\perp} = \sigma\left(U_{\perp}\right)$$

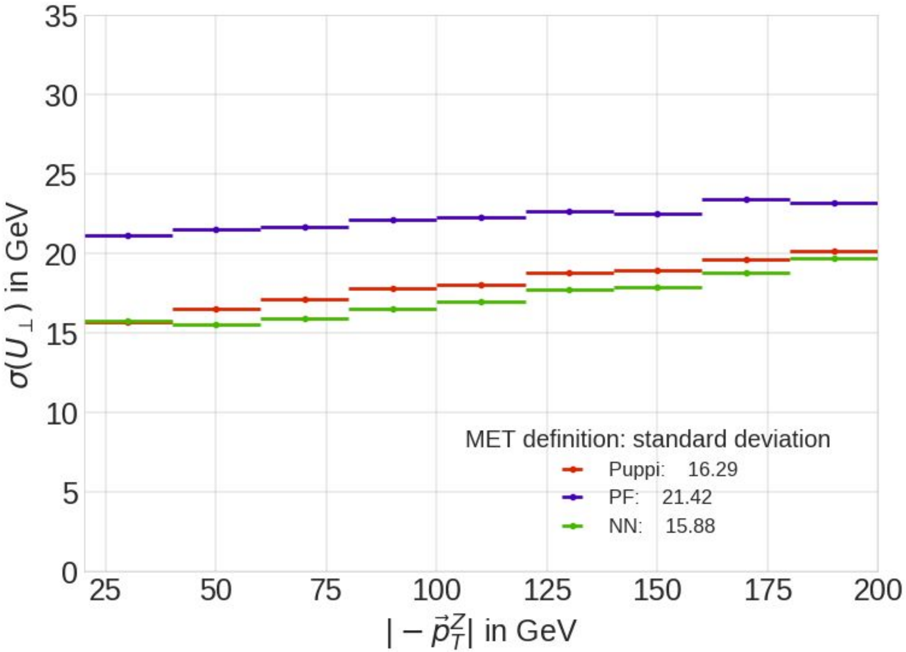
Resolution perpendicular: inclusive



- Perpendicular resolution smaller than parallel resoltution for Puppi and PF
- Not optimized in loss of NN
- Minimum resolution for NN inclusive

$$\text{Resolution}_{\perp} = \sigma\left(U_{\perp}\right)$$

Resolution perp vs. p_T



- In smallest p_T bin the resolutions of NN and Puppi are the same
  - NN response next to 1 in this bin
  - Puppi response < 1 in this bin
- NN smallest resolution for higher p_Ts

Conclusion

Performance inclusive overview

		Response	Resolution parallel	Resolution perpendicular
	Puppi	0.93±0.55	±18.62	±16.29
	PF	0.93±0.68	±22.36	±21.42
	NN	1.00±0.48	±16.11	±15.88
Response corrected[1]	Puppi	0.93±0.55	±20.02	±17.52
	PF	0.93±0.68	±24.04	±23.03
	NN	1.00±0.48	±16.11	±15.88

[1] Response corrected: resolution divided by response

Conclusion

- PF has less tails than Puppi
- Puppi has better resolution than PF

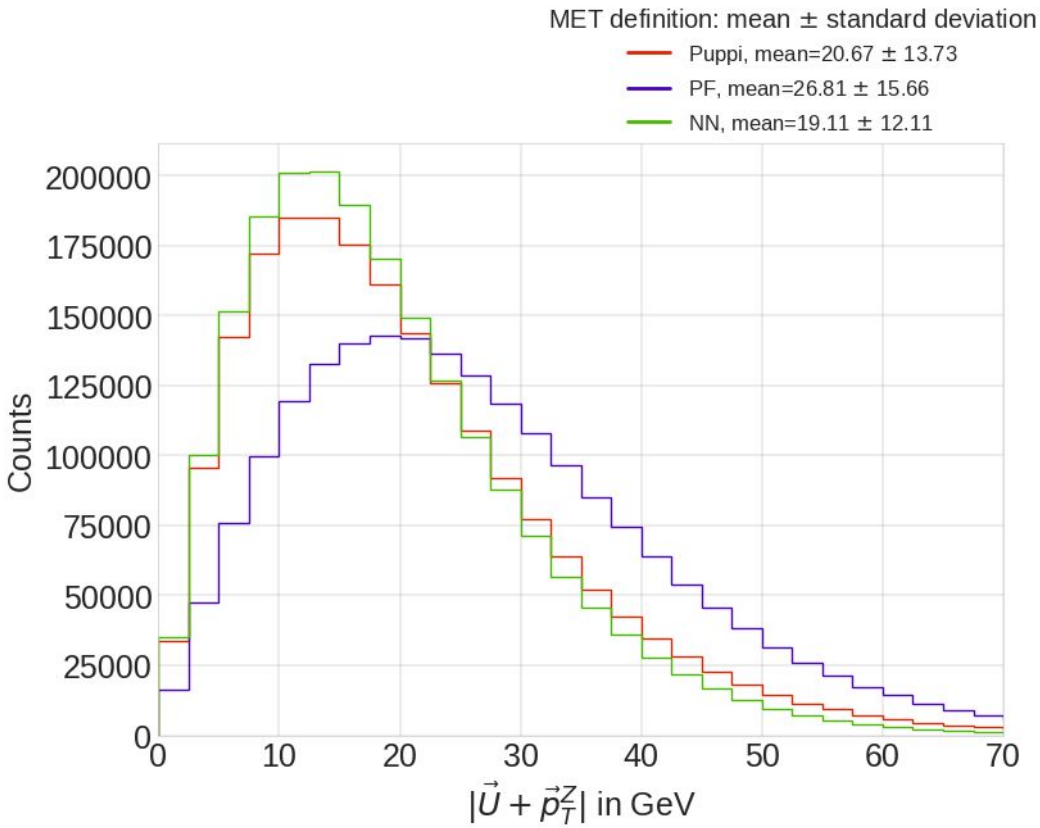
→ NN is able to combine these two advantages for a overall promising result

Outlook

- Physics benchmark with reweight W-mass-reconstruction
- Combined contribution for MVA W-mass-reconstruction with
  - **Pedro Vieira De Castro Ferreira Da Silva,** CMS
  - **Paolo Gunnellini,** Uni Hamburg (CMS)
- GitHub repository for everyone to use with support

Appendix

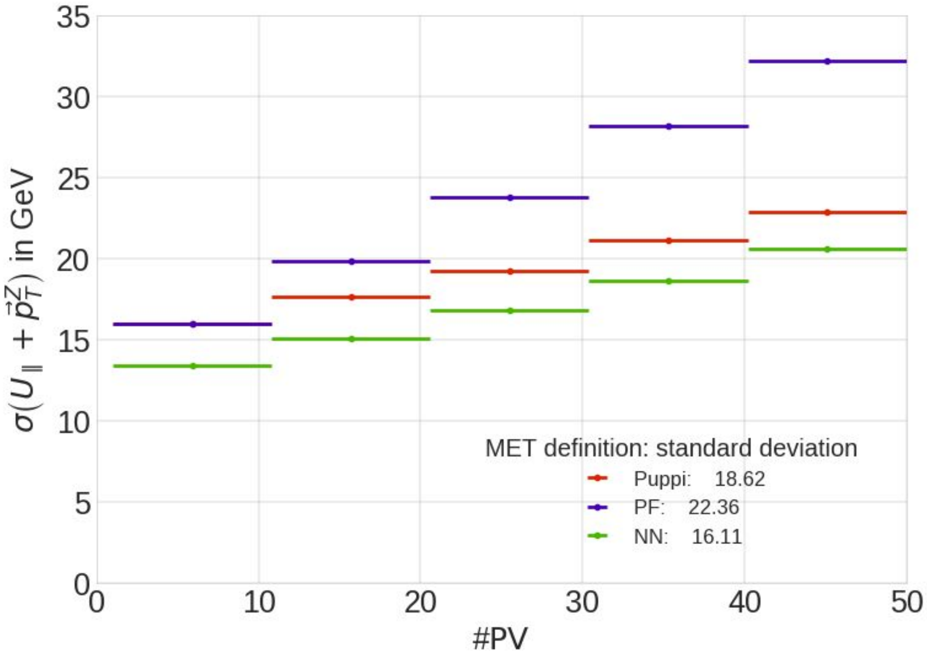
Histogram absolute MET



- Mean of inclusive MET highest for PF
- Mean of inclusive MET for NN under Puppi
- Less tails for NN

$$\text{Resolution}_{||} = \sigma\left(U_{||} + p_T^Z\right)$$

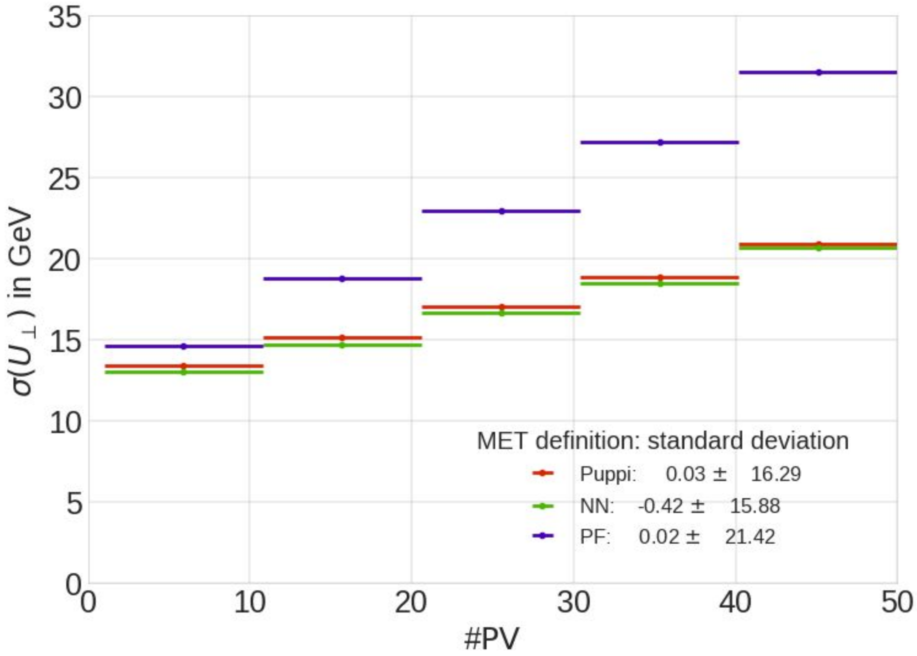
resolution para vs. #PV



- Puppi and NN are less dependent on #PV than PF
- Advantage of NN in comparison to Puppi stable over #PV

$$\text{Resolution}_{\perp} = \sigma\left(U_{\perp}\right)$$

Resolution perp vs. #PV



- Puppi and NN are less dependent on #PV than PF
- Perpendicular resolution of NN and Puppi comparable over #PV