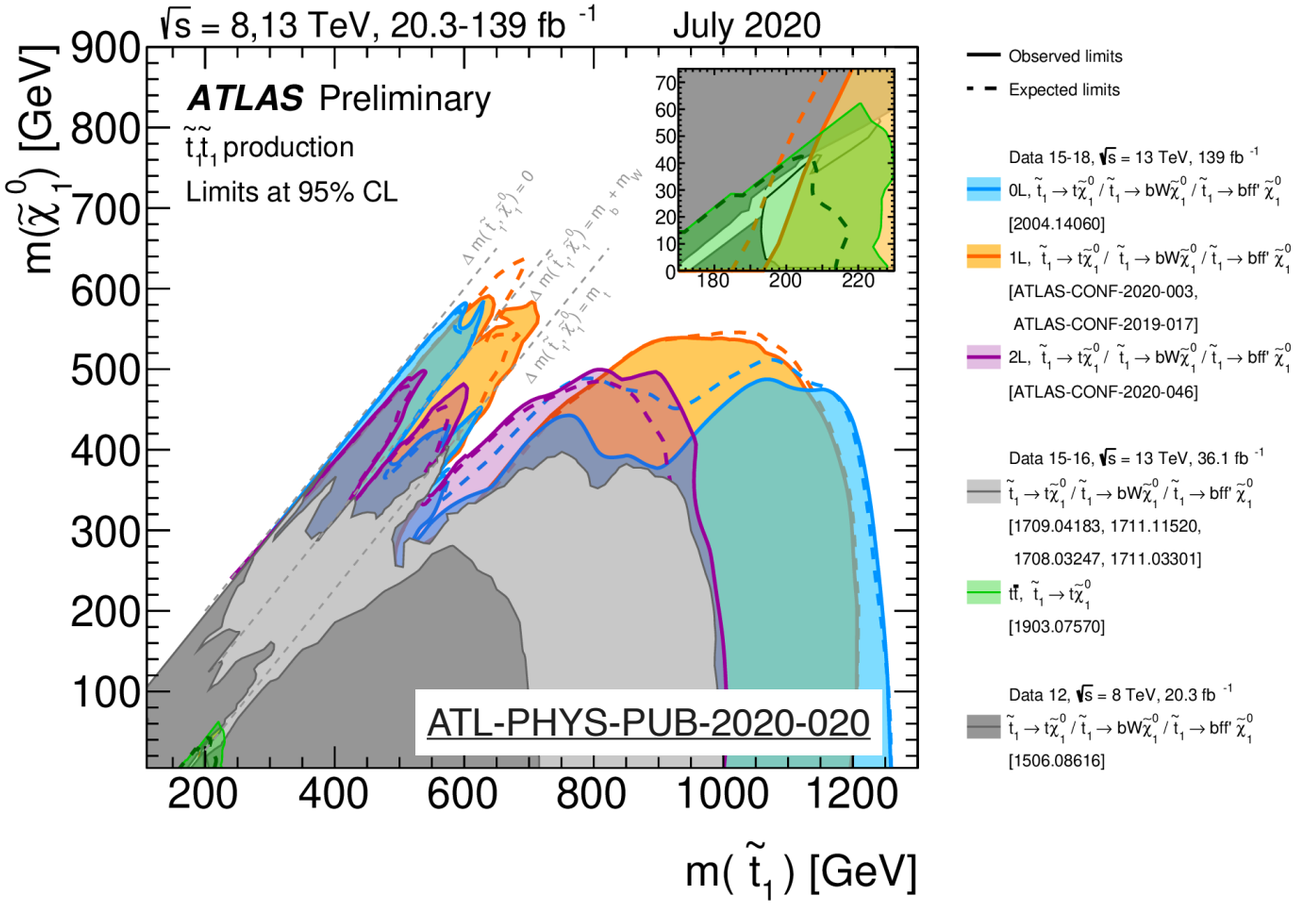
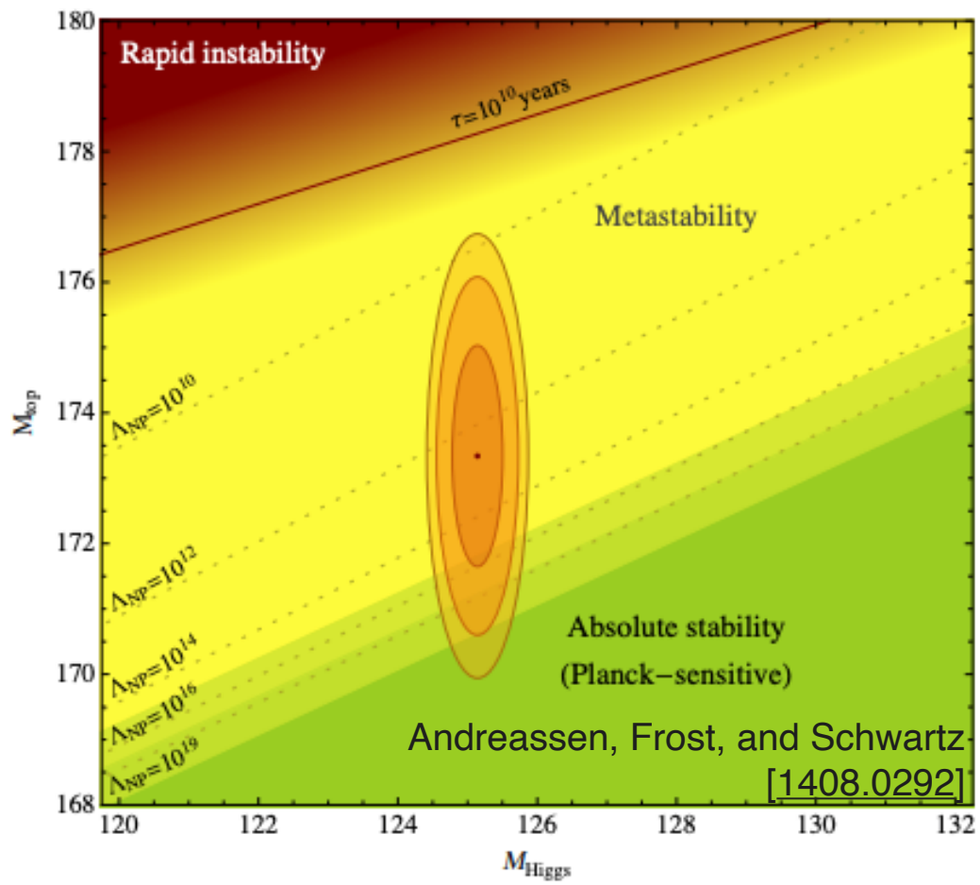
Bryan Ostdiek, TOP2020 Workshop
September 17, 2020

HARVARD UNIVERSITY

Reducing uncertainty in top quark mass with Machine Learning





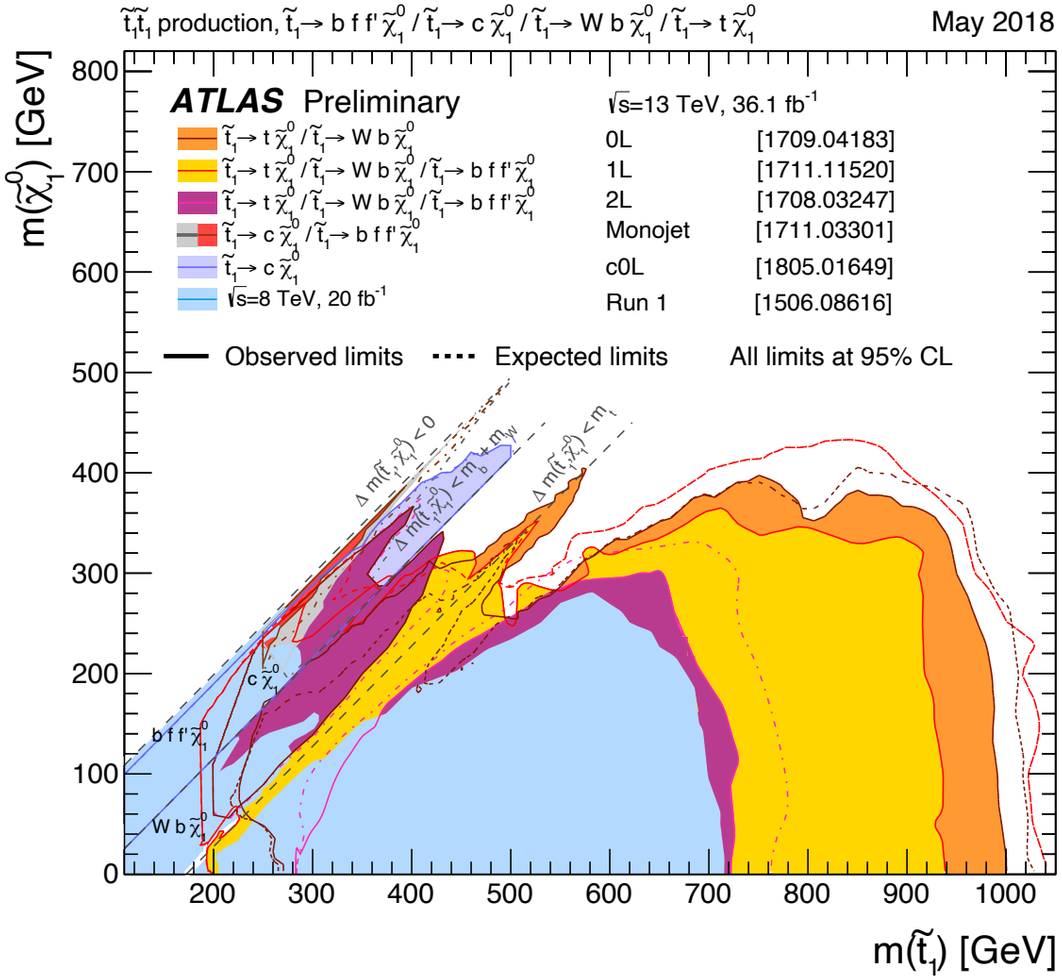
Based on:

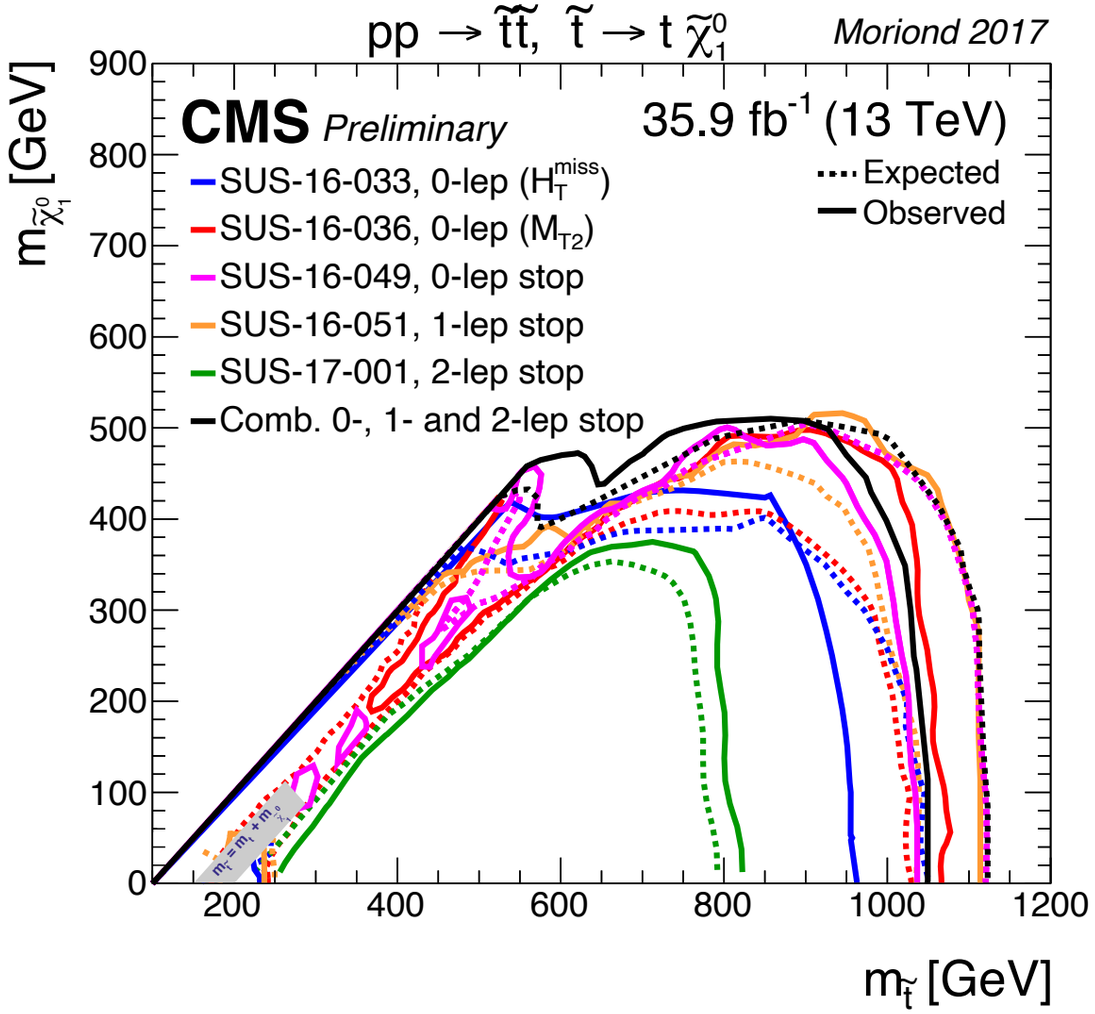
[1804.00111] T. Cohen, W. Hopkins, S. Majewski, and BO

[1909.09670] T. Cohen, S. Majewski, BO, and P. Zheng

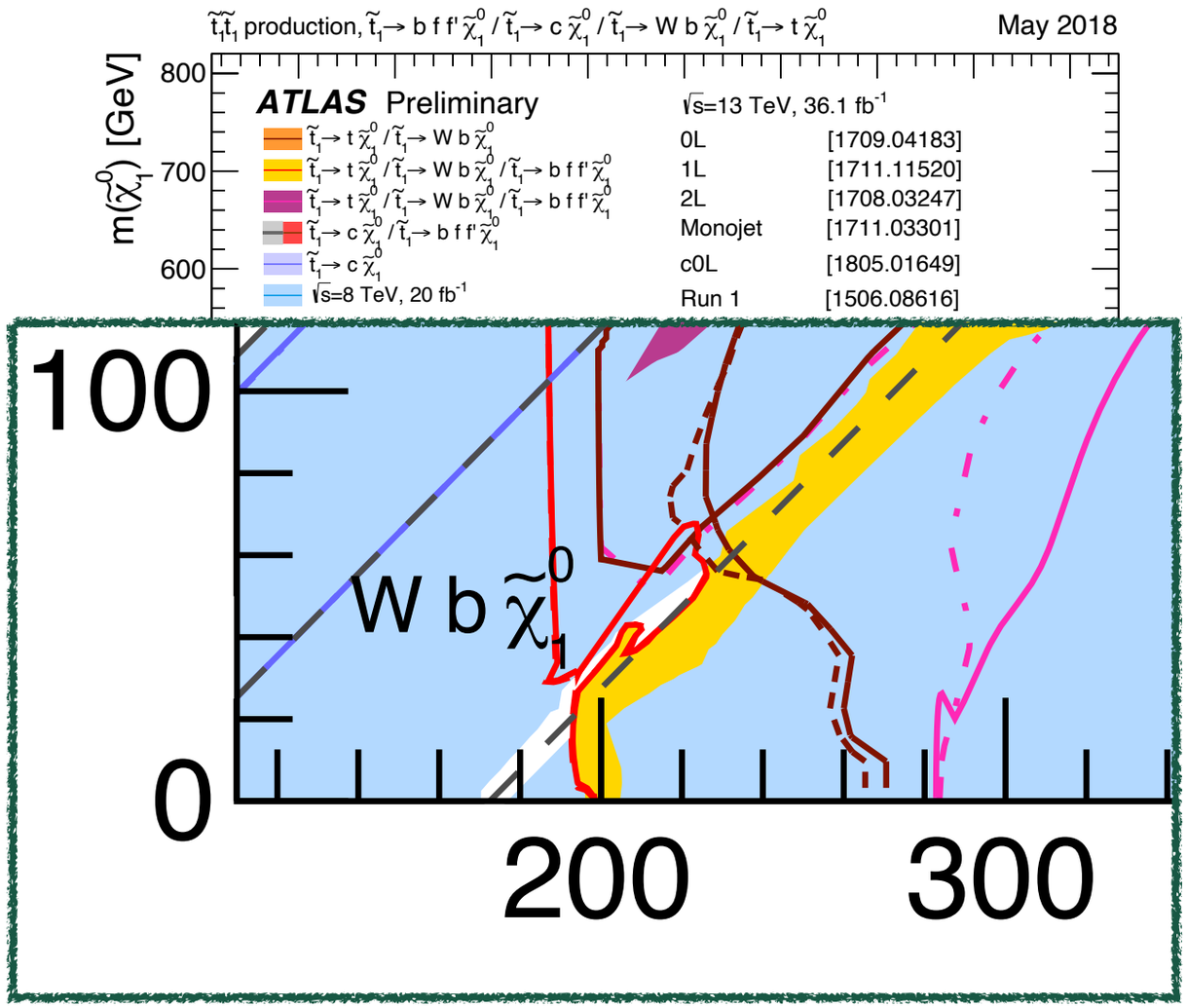
[ongoing] F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz

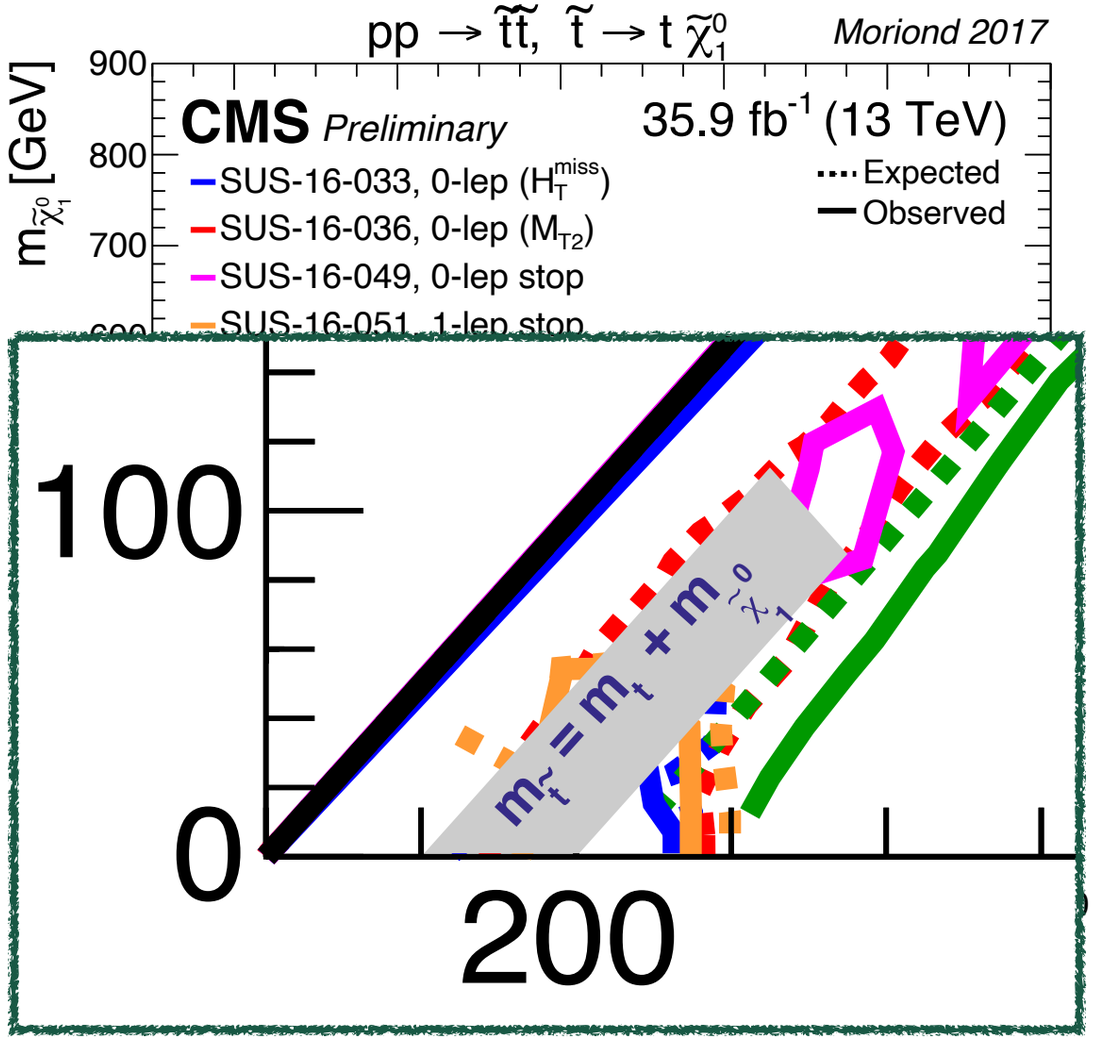
Splinters in the Stop Search





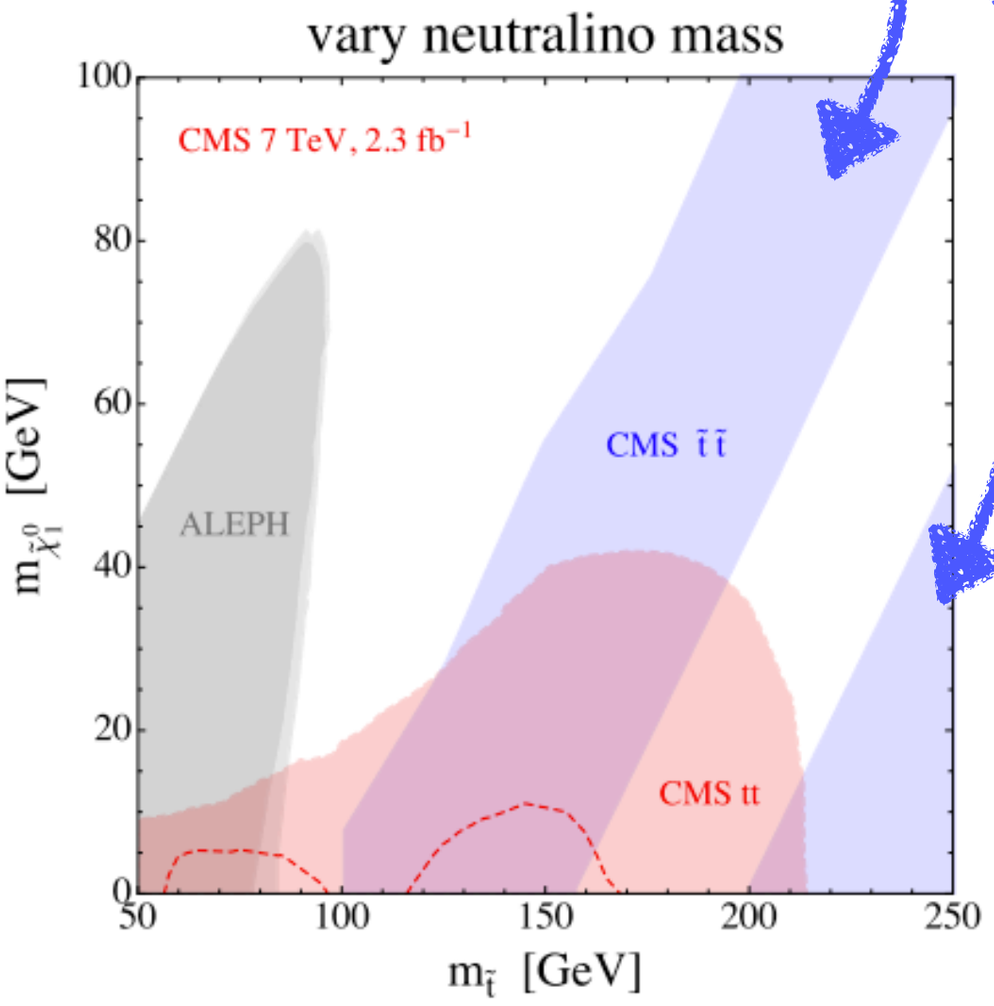
Splinters in the Stop Search





Top quark measurements provide constraints

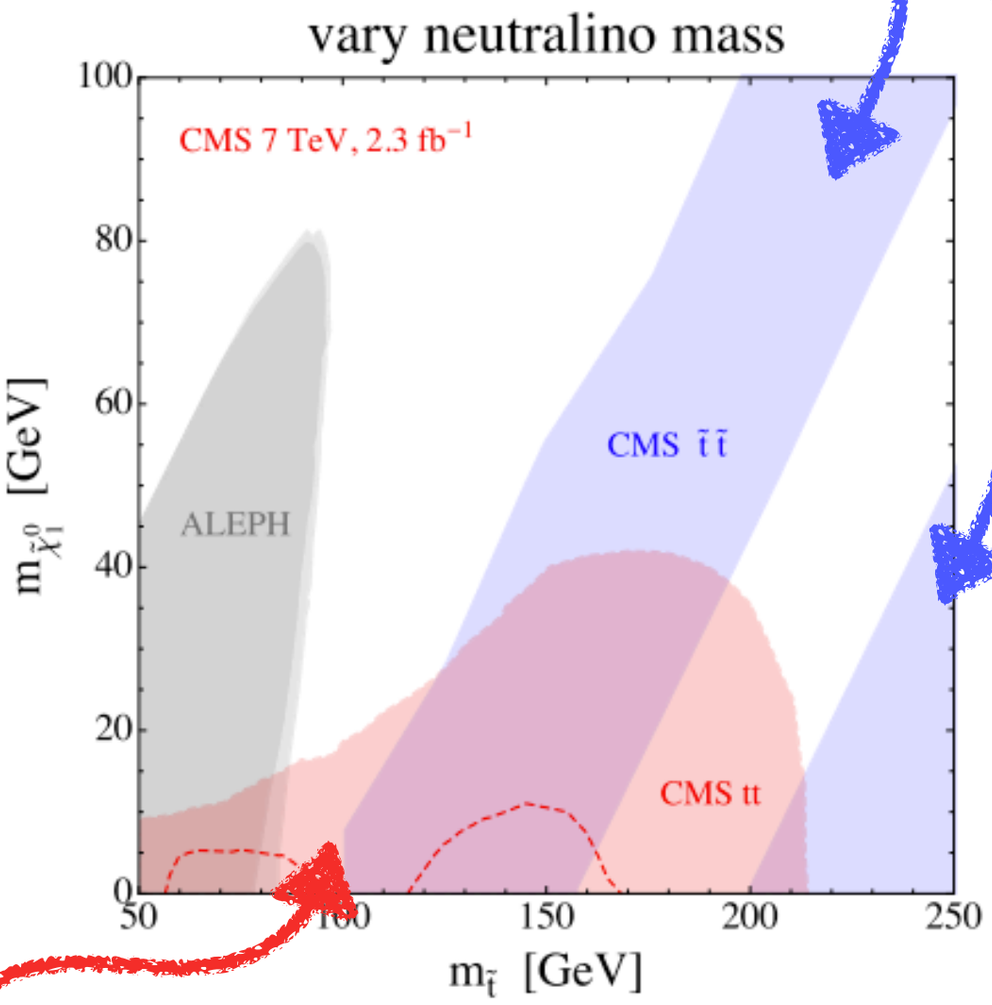
Direct stop searches exclude blue area (and leave holes)



Czakon, Mitov, Papucci, Ruderman, and Weiler [1407.1043]

Top quark measurements provide constraints

Direct stop searches exclude blue area (and leave holes)

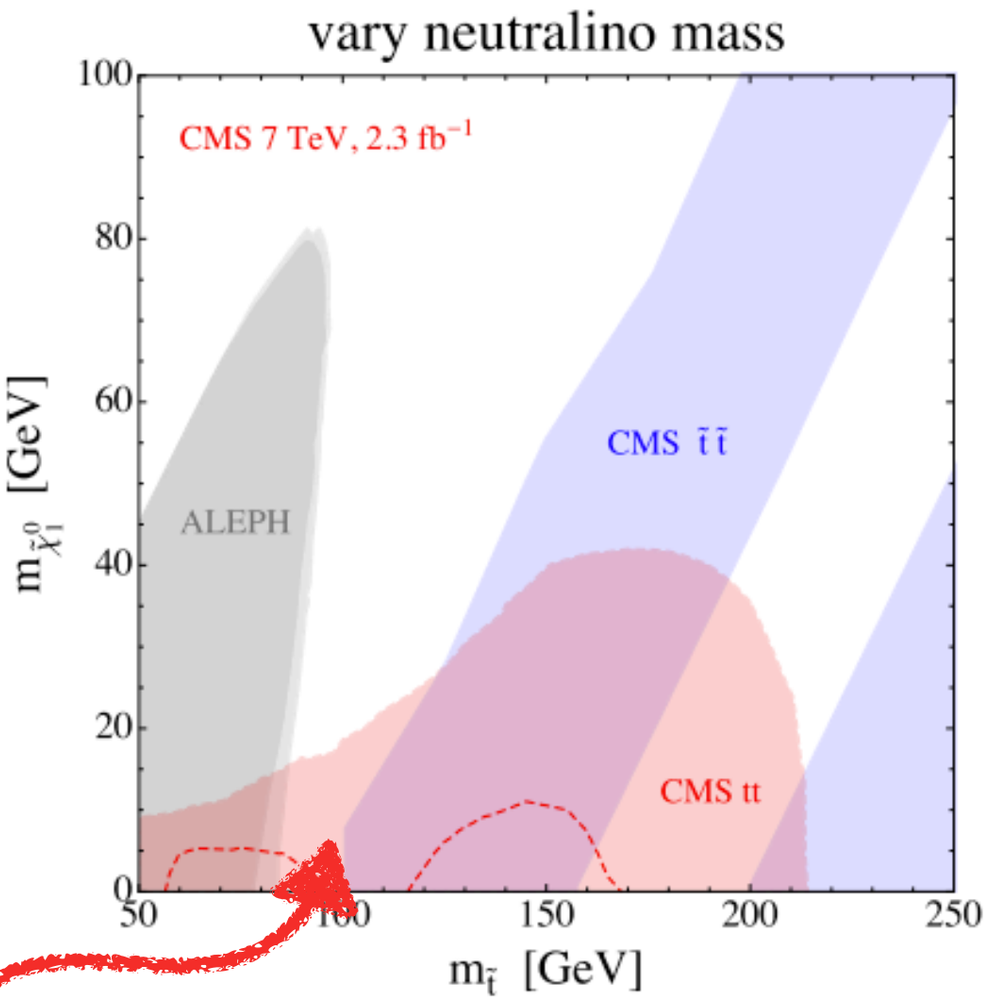


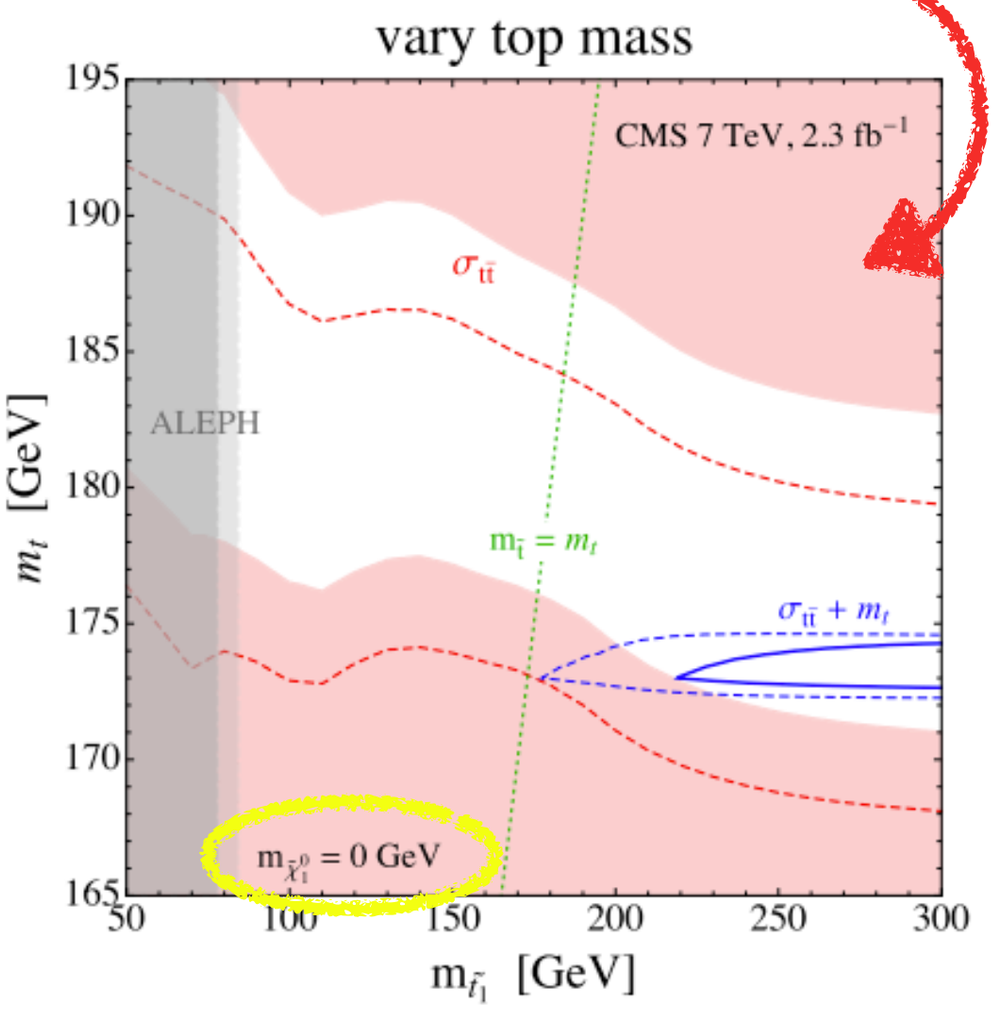
Using cross section measurement (assuming mass is known) constrains the amount of "stop contamination"

Czakon, Mitov, Papucci, Ruderman, and Weiler [1407.1043]

Top quark measurements provide constraints

Now don't assume top mass;
Cross section provides limit in stop-top plane





Using cross section measurement (assuming mass is known) constrains the amount of "stop contamination"

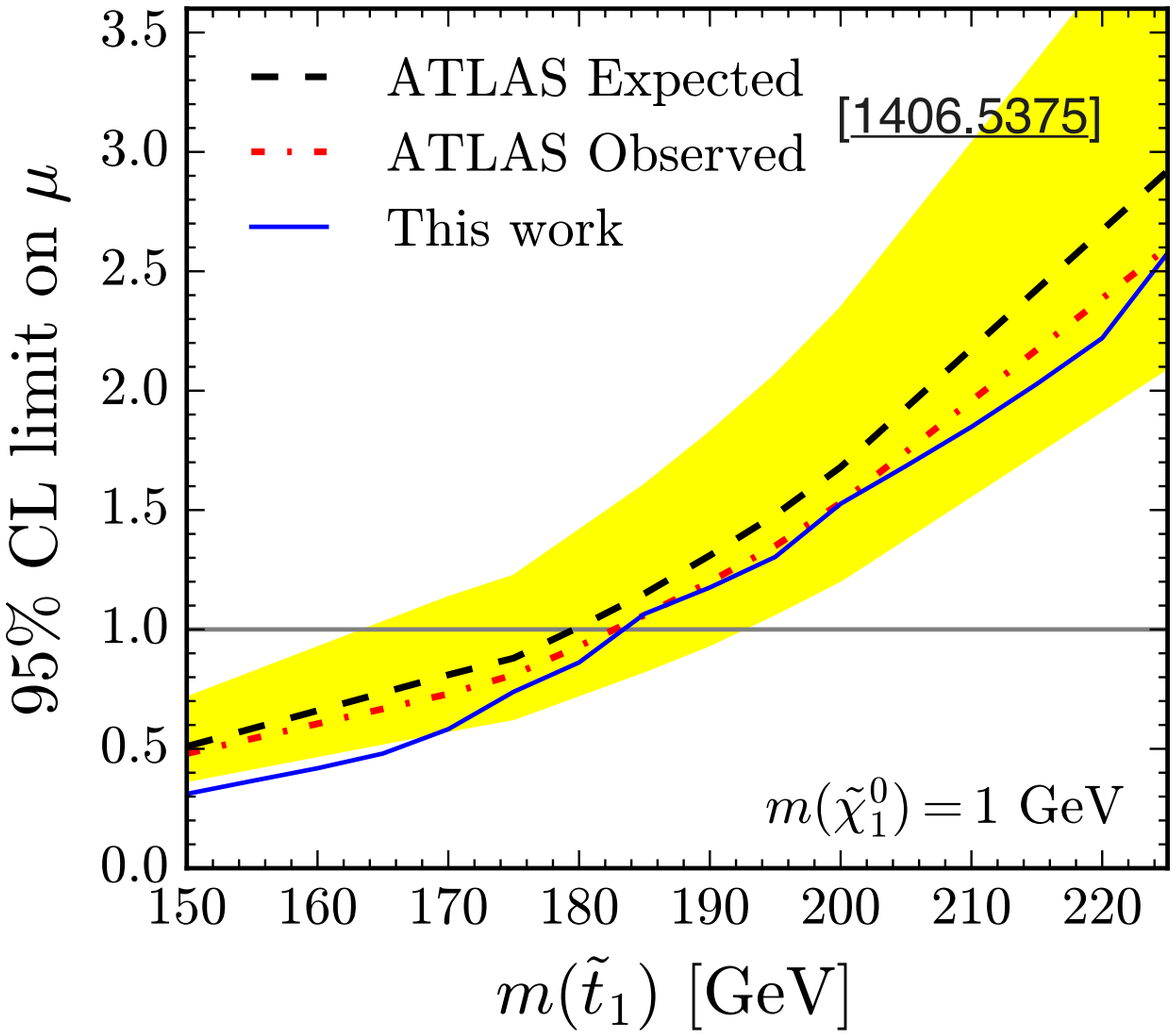
(*)Assumes stops do not affect top mass measurement

Czakon, Mitov, Papucci, Ruderman, and Weiler [1407.1043]

Top quark measurements provide constraints

ATLAS used ratio of cross section at 7 TeV and 8 TeV as stop constraint (assuming top mass known)

$$pp \to \tilde{t}_1 \tilde{t}_1 \to e^{\pm}\mu^{\mp} b\bar{b}\,\tilde{\chi}_0^1\,\tilde{\chi}_0^1\,\nu\bar{\nu}$$

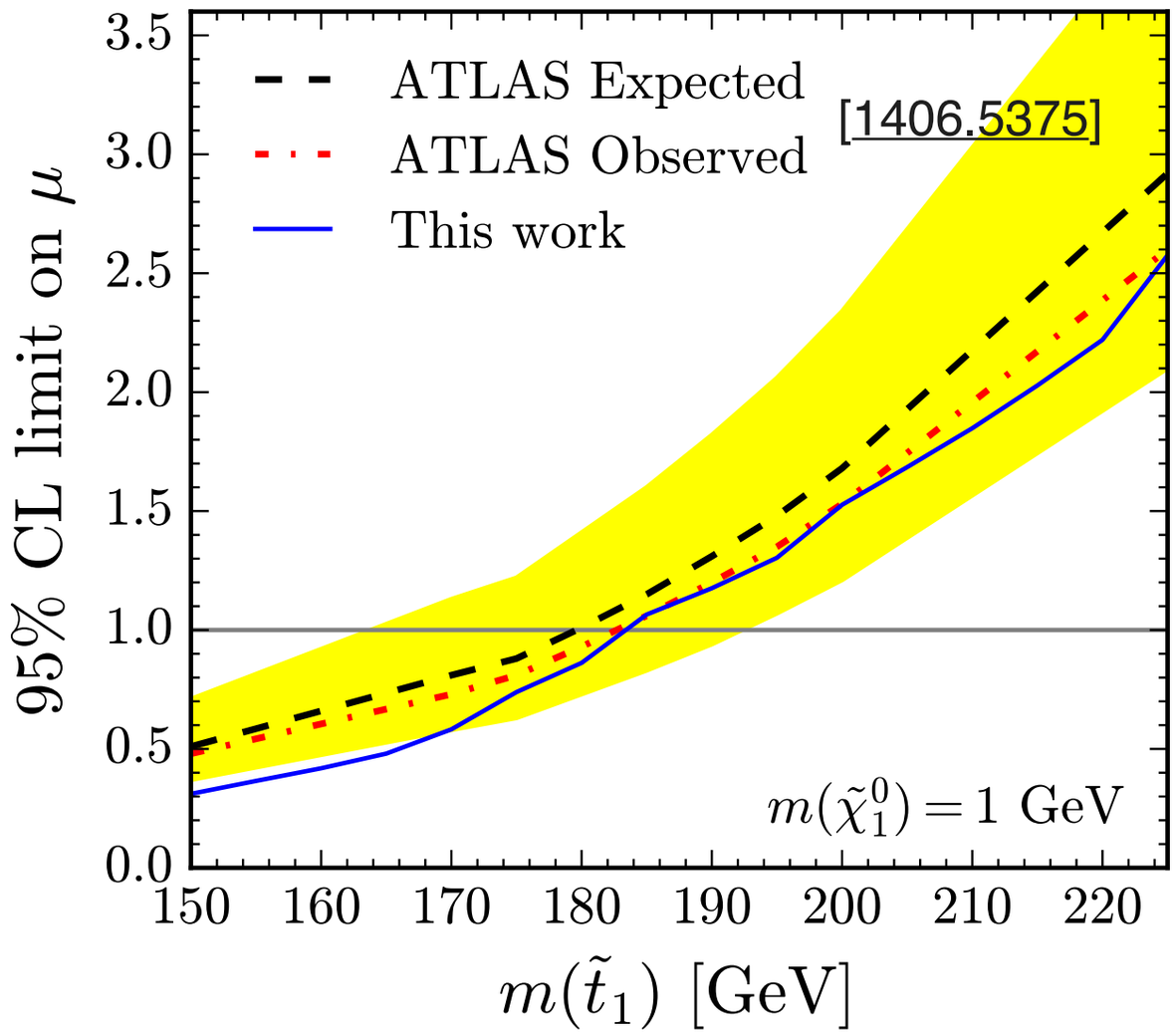


T. Cohen, W. Hopkins, S. Majewski, and BO [1804.00111]

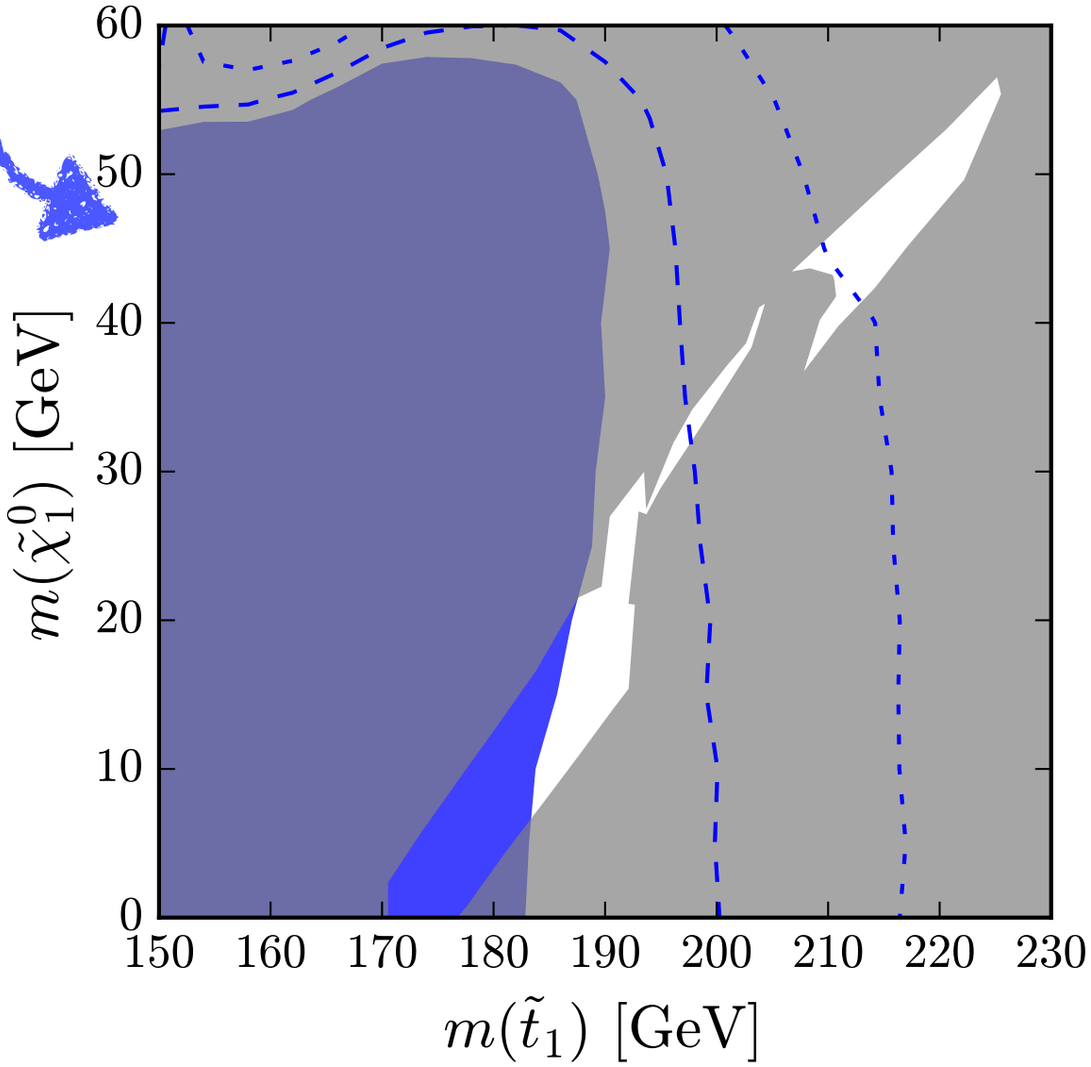
Top quark measurements provide constraints

ATLAS used ratio of cross section at 7 TeV and 8 TeV as stop constraint (assuming top mass known)

$$pp \to \tilde{t}_1 \tilde{t}_1 \to e^{\pm}\mu^{\mp} b\bar{b}\,\tilde{\chi}_0^1\,\tilde{\chi}_0^1\,\nu\bar{\nu}$$



Extend to the (stop, neutralino) plane

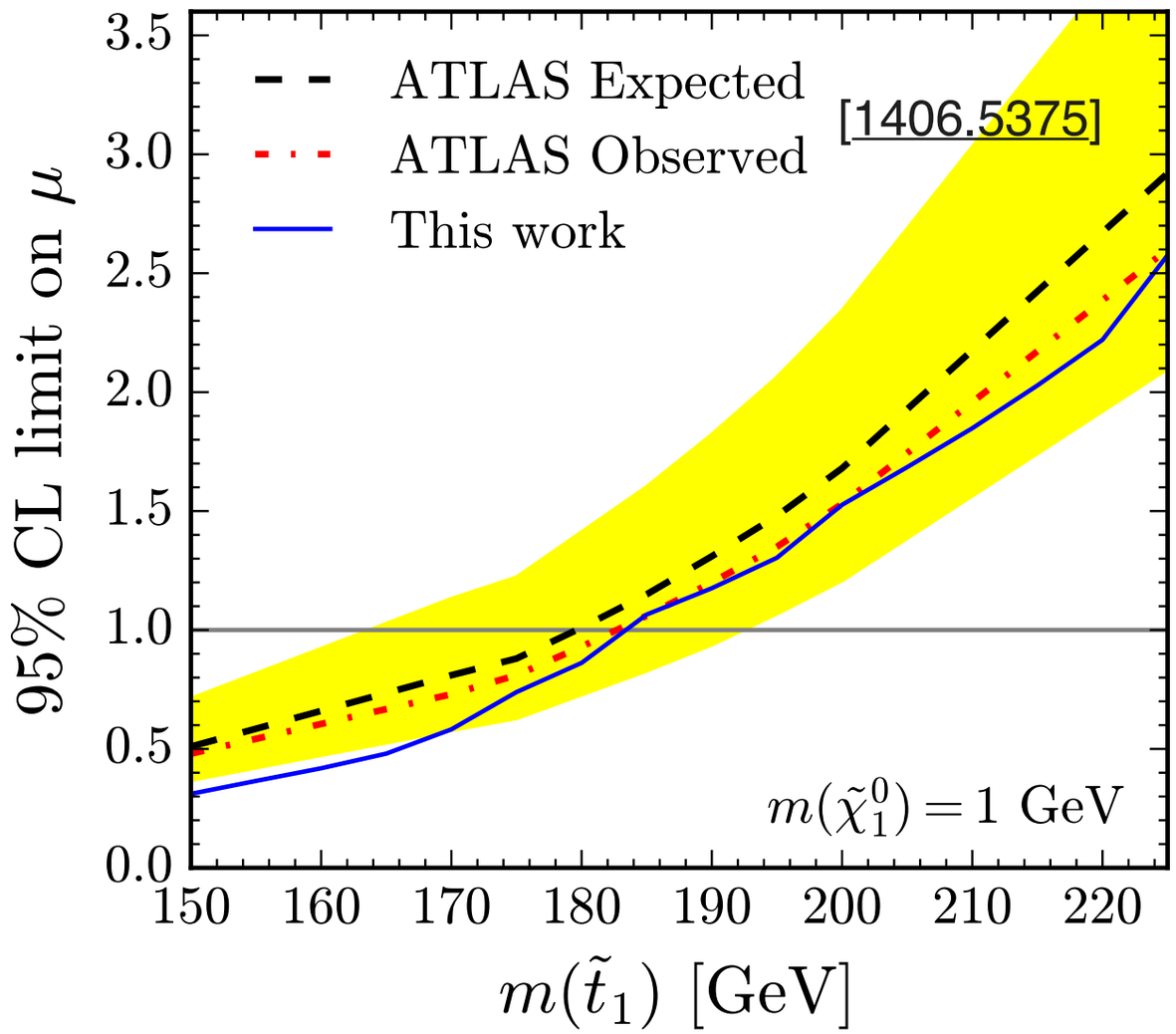


T. Cohen, W. Hopkins, S. Majewski, and BO [1804.00111]

Top quark measurements provide constraints

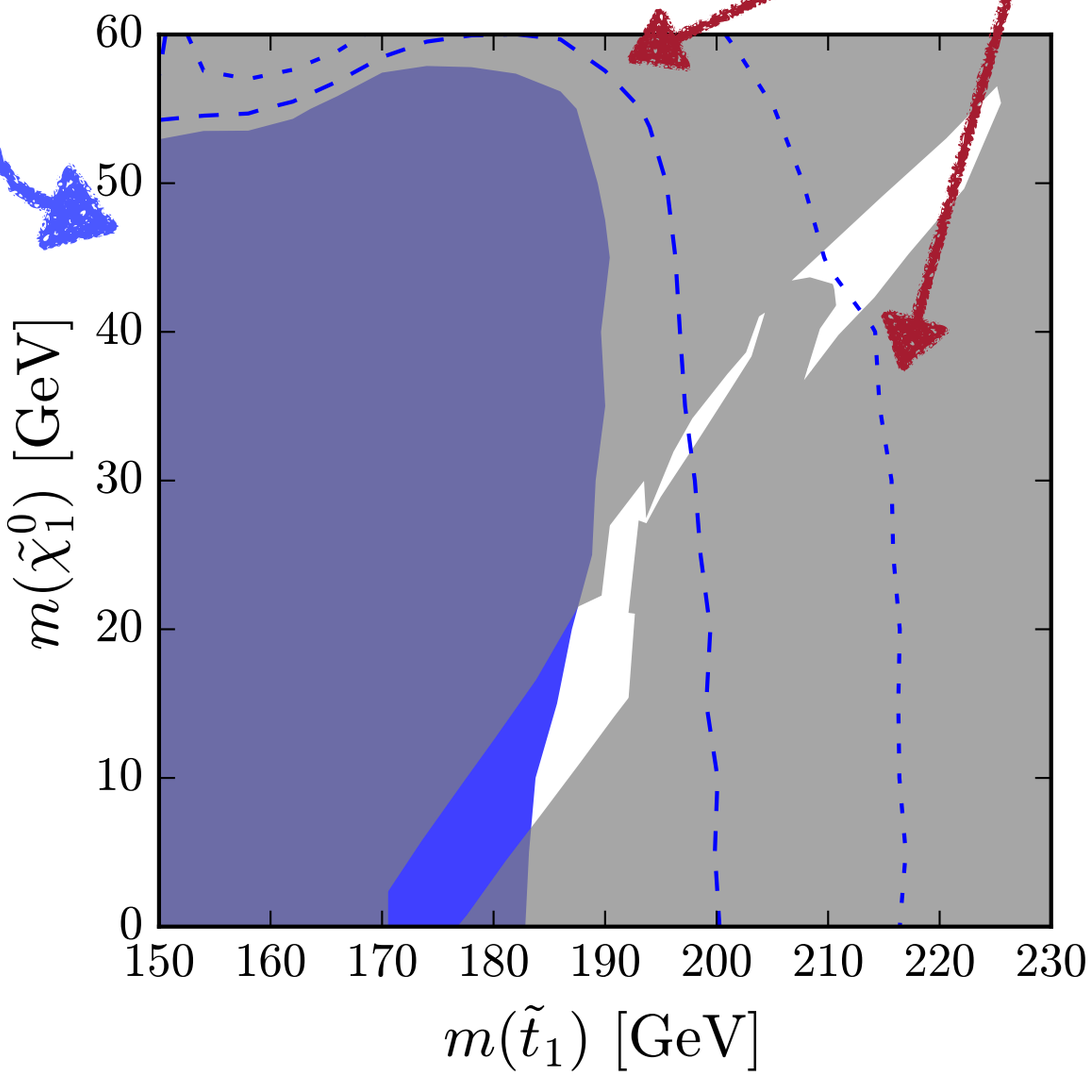
ATLAS used ratio of cross section at 7 TeV and 8 TeV as stop constraint (assuming top mass known)

$$pp \to \tilde{t}_1 \tilde{t}_1 \to e^{\pm}\mu^{\mp} b\bar{b}\,\tilde{\chi}_0^1\,\tilde{\chi}_0^1\,\nu\bar{\nu}$$



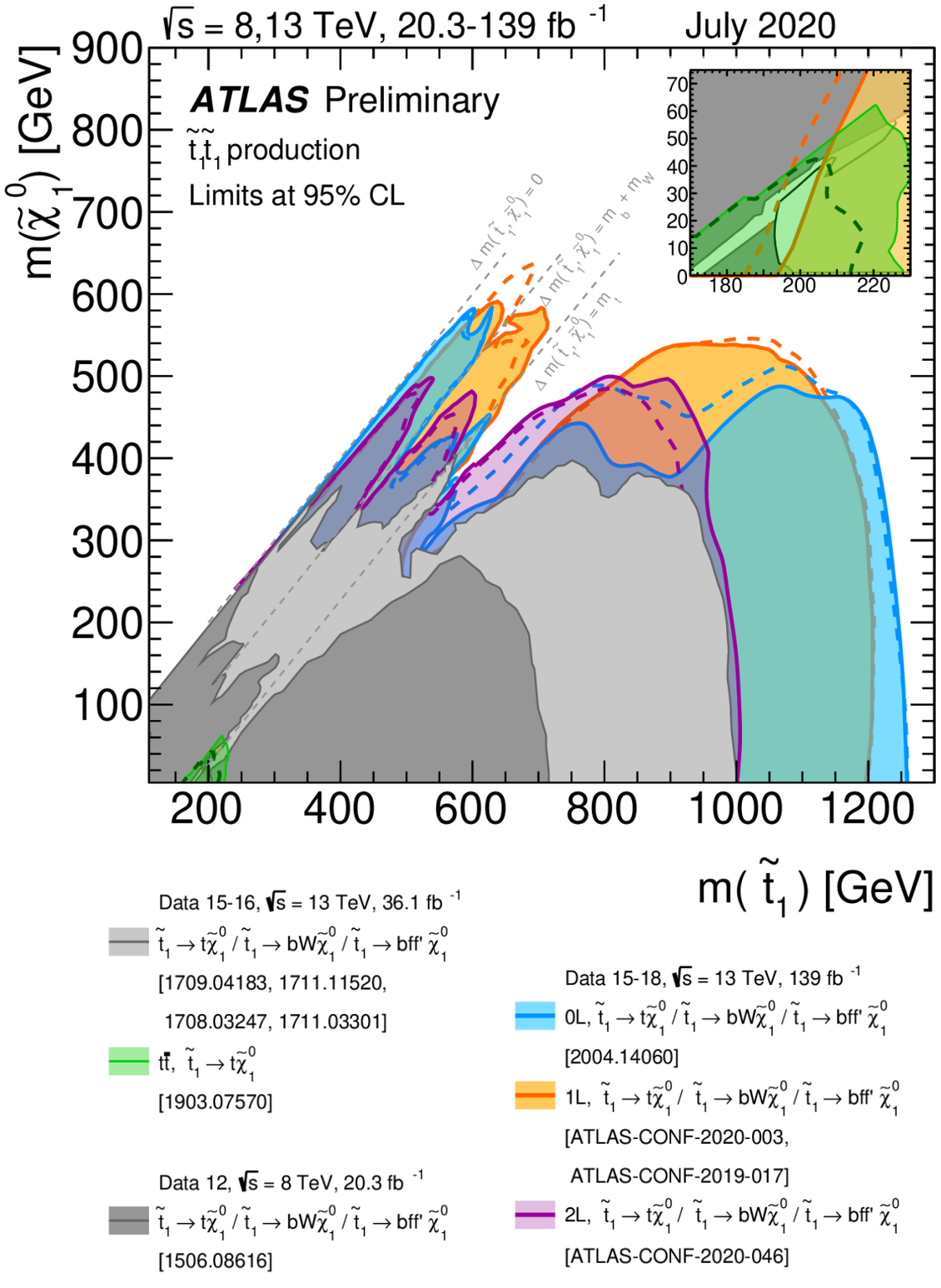
Extend to the (stop, neutralino) plane

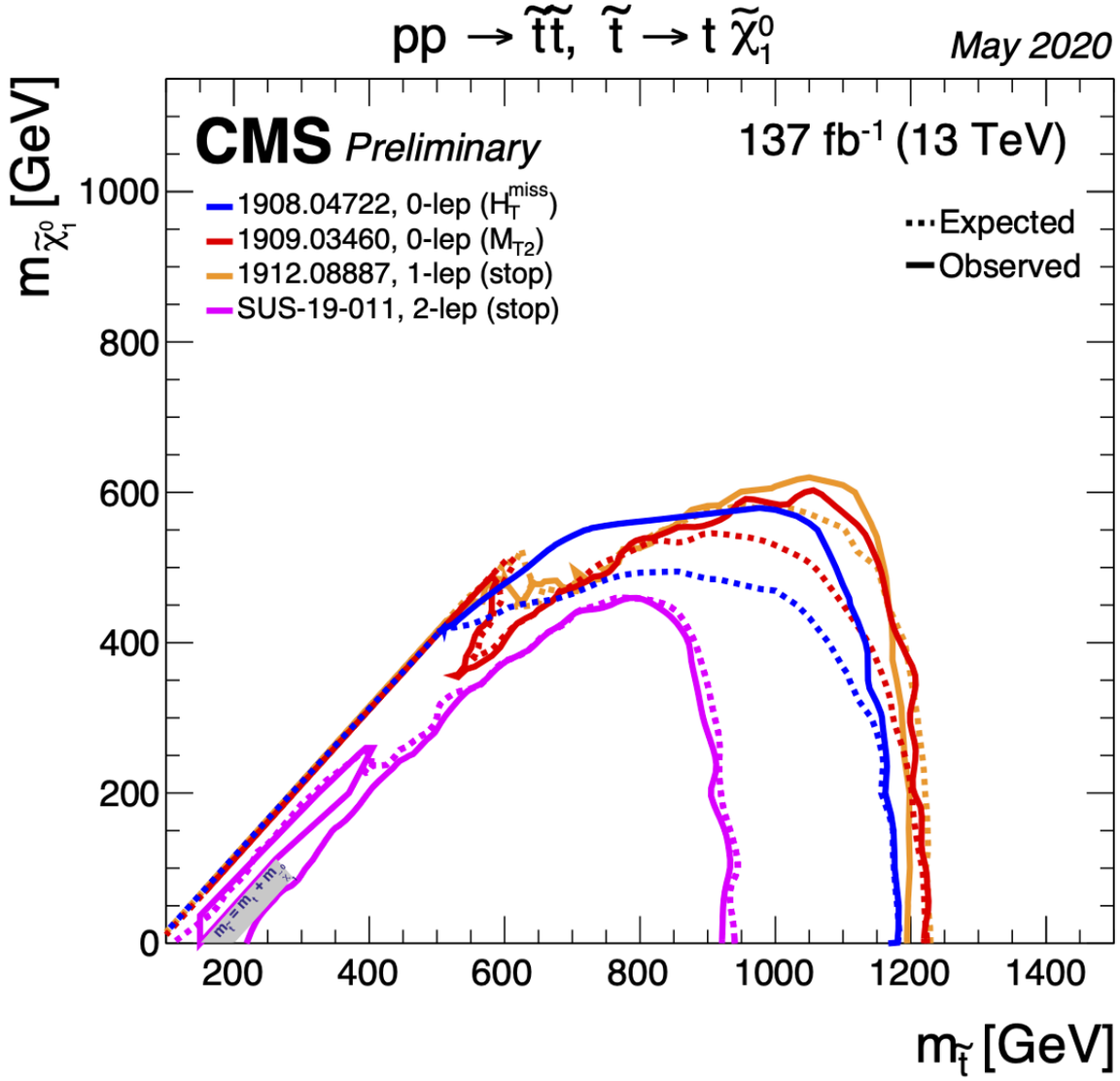
Reduce uncertainty by factor of 2 (10)



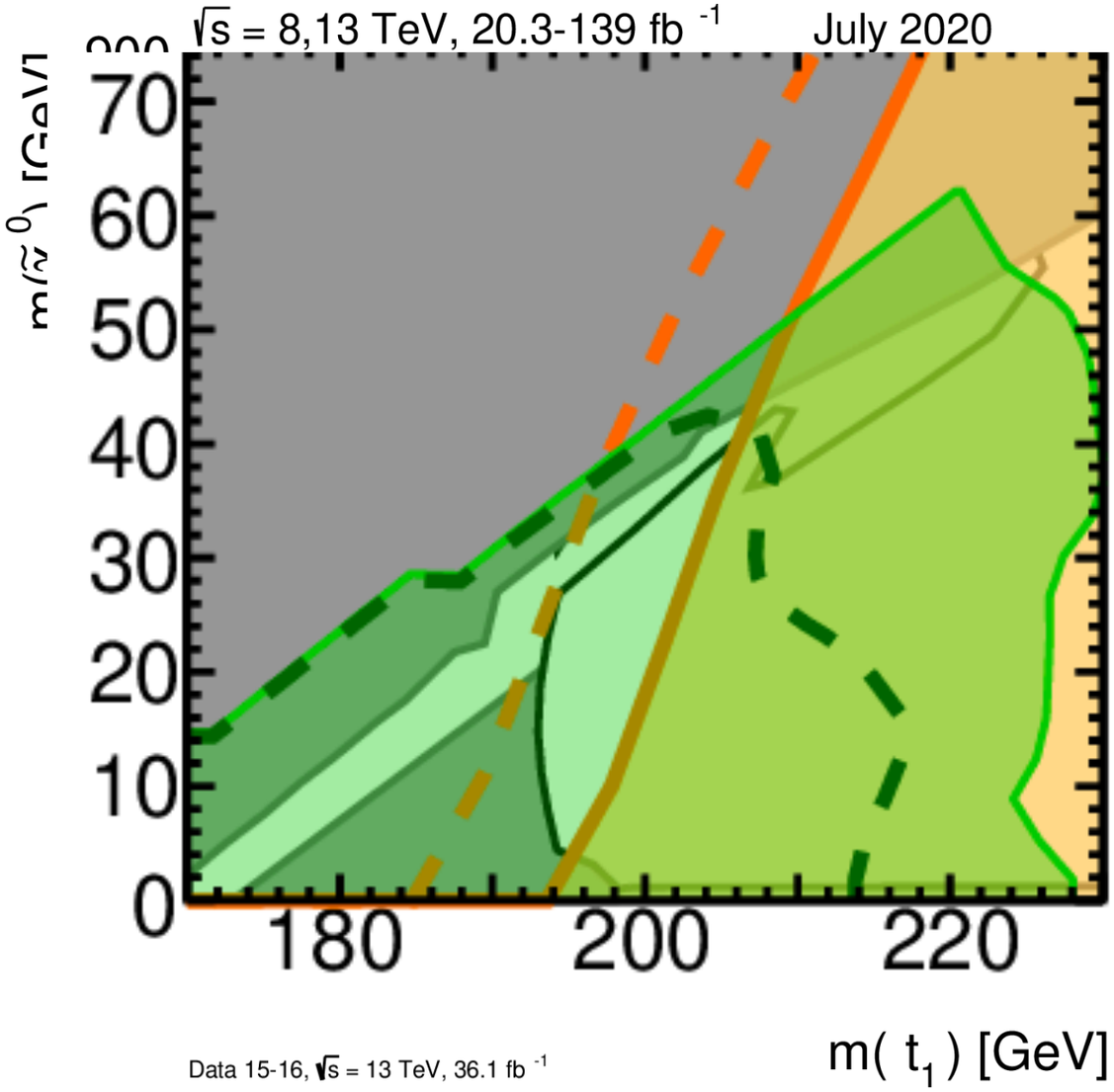
T. Cohen, W. Hopkins, S. Majewski, and BO [1804.00111]

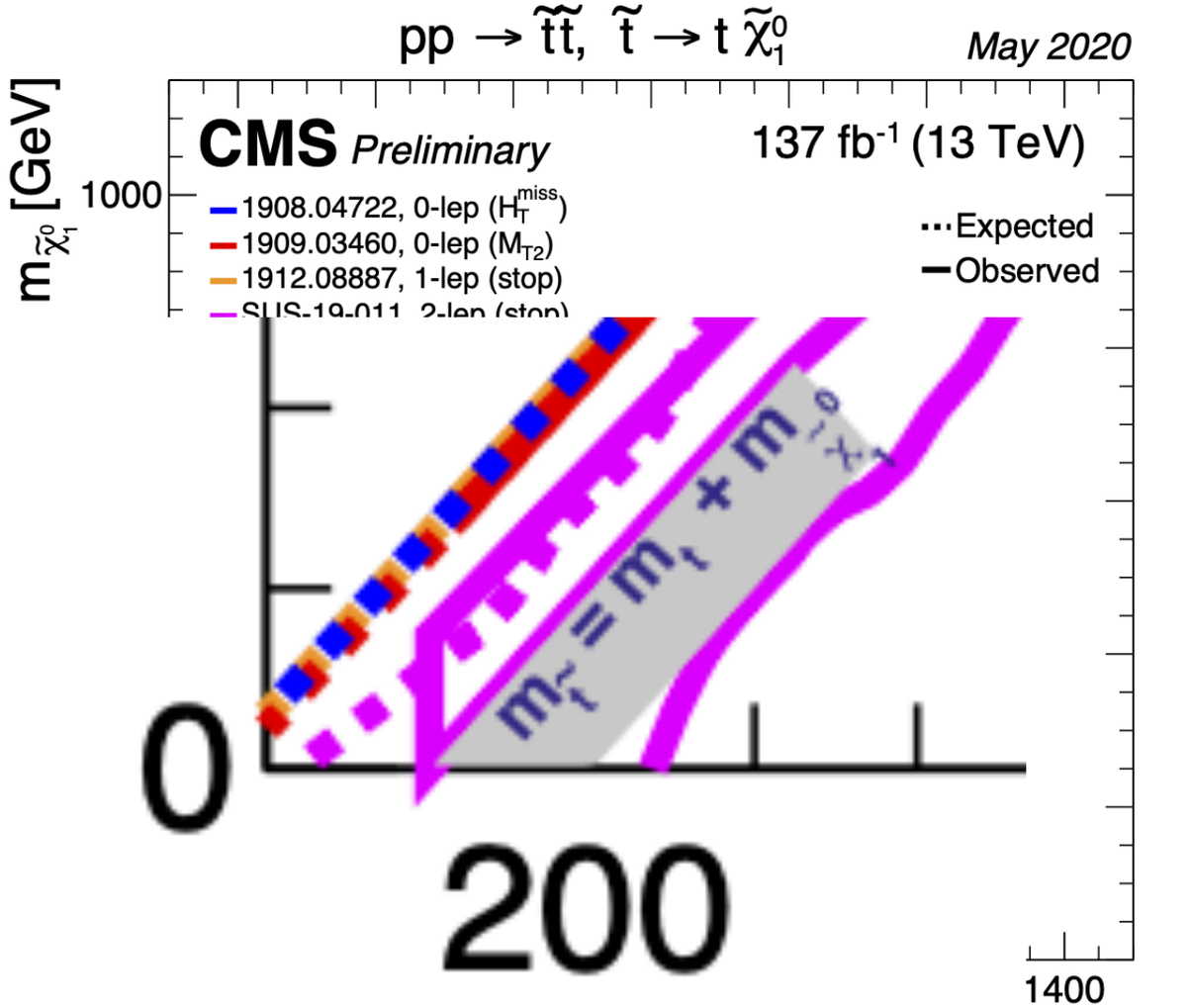
Top quark measurements provide constraints





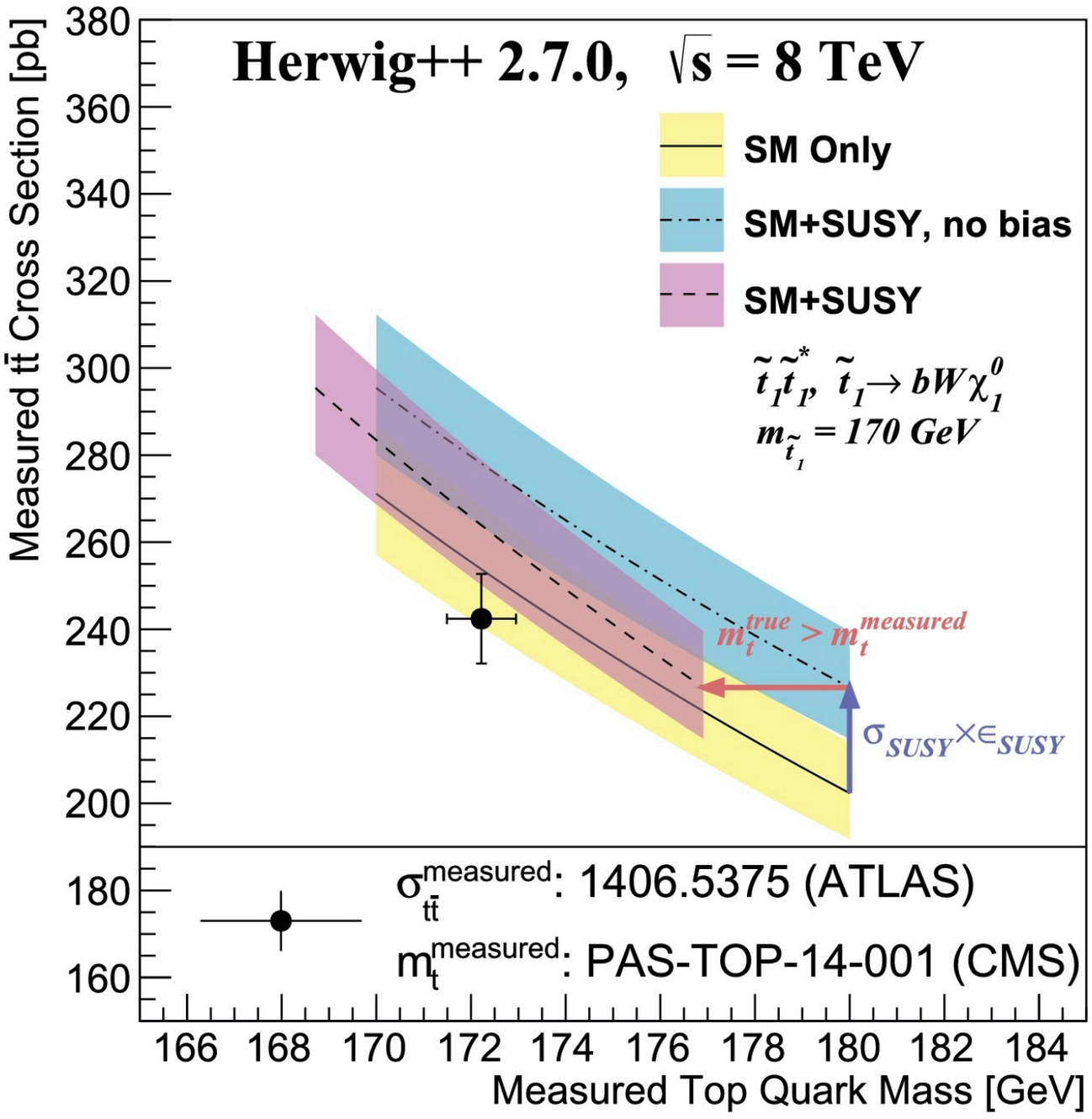
Top quark measurements provide constraints





Region is probably excluded
*Does not take into account effect of stops on measured top mass

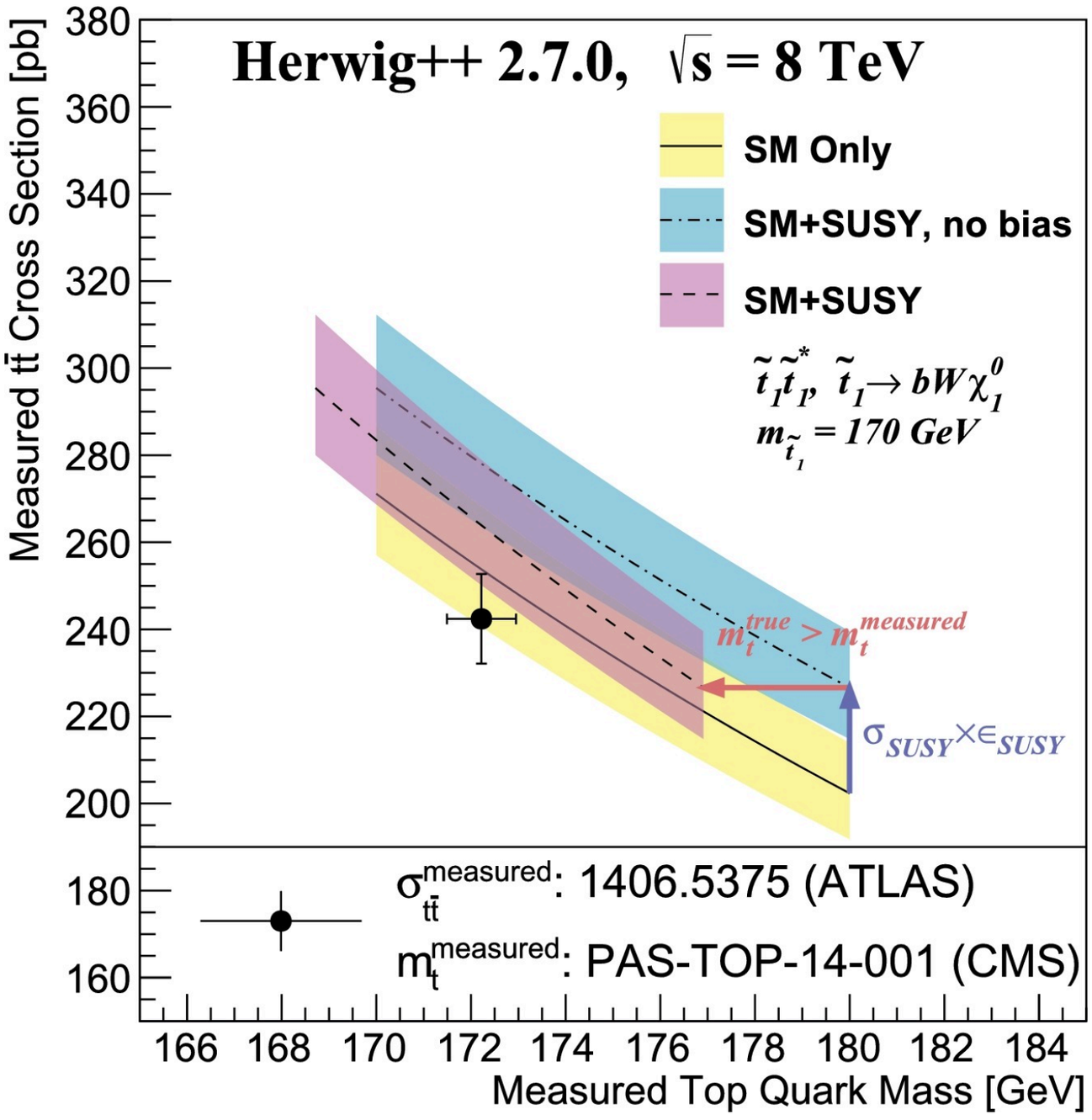
Top quark measurements provide constraints



- Previous constraints assumed top mass is known
- As stops can alter measured cross section, can also contaminate mass measurement

Eifert and Nachman [1410.7025]

Top quark measurements provide constraints



Eifert and Nachman [1410.7025]

- Previous constraints assumed top mass is known
- As stops can alter measured cross section, can also contaminate mass measurement
- Different channels have varying amounts of contamination
- Reducing uncertainty can provide independent constraints
- Can machine learning help reduce uncertainty?

How is the top mass measured?

Template Method

1) Pick an observable which is correlated with the Monte Carlo top mass

2) Get distribution of observable at multiple values of the mass

3) Fit each distribution with a parametric function

4) Interpolate the function parameters as a function of the input mass

5) Go to real data and find the mass which fits the best

How is the top mass measured?

Template Method

1) Pick an observable which is correlated with the Monte Carlo top mass

0 *l:* Ratio of 3-jet invariant mass to 2-jet invariant mass

1 *l:* Common propagator value which maximizes combined likelihood

2 *l:* Invariant mass of the lepton and b-jet

2) Get distribution of observable at multiple values of the mass

3) Fit each distribution with a parametric function

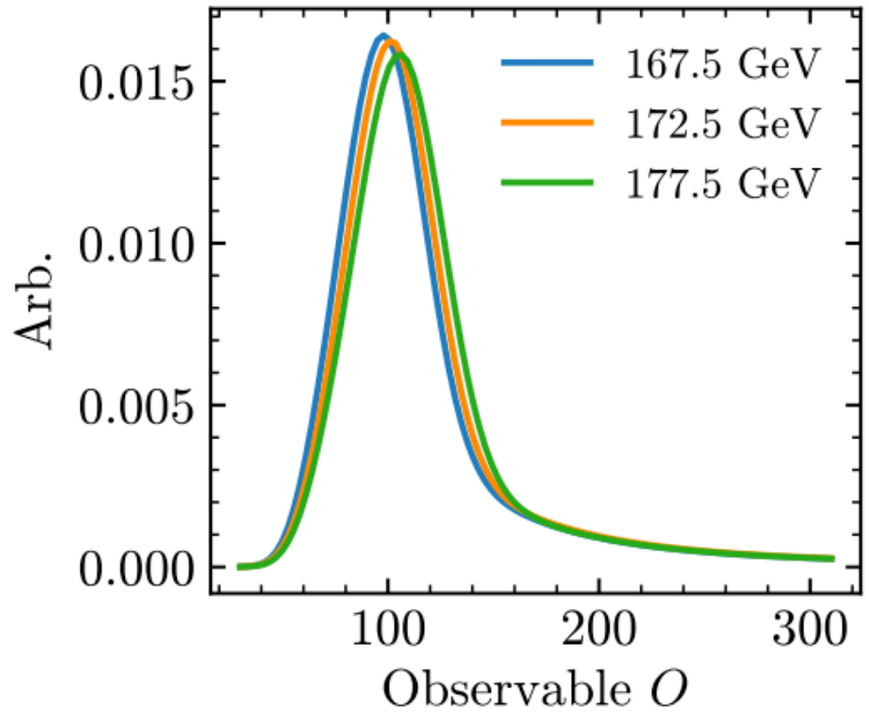
4) Interpolate the function parameters as a function of the input mass

5) Go to real data and find the mass which fits the best

How is the top mass measured?

Template Method

1) Pick an observable which is correlated with the Monte Carlo top mass
2) Get distribution of observable at multiple values of the mass

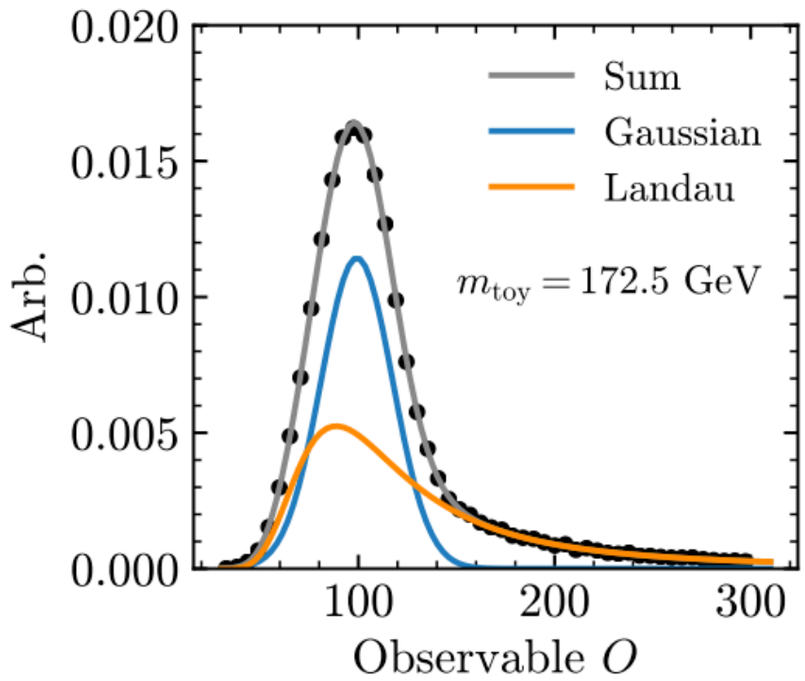


3) Fit each distribution with a parametric function
4) Interpolate the function parameters as a function of the input mass
5) Go to real data and find the mass which fits the best

How is the top mass measured?

Template Method

1) Pick an observable which is correlated with the Monte Carlo top mass
2) Get distribution of observable at multiple values of the mass
3) Fit each distribution with a parametric function

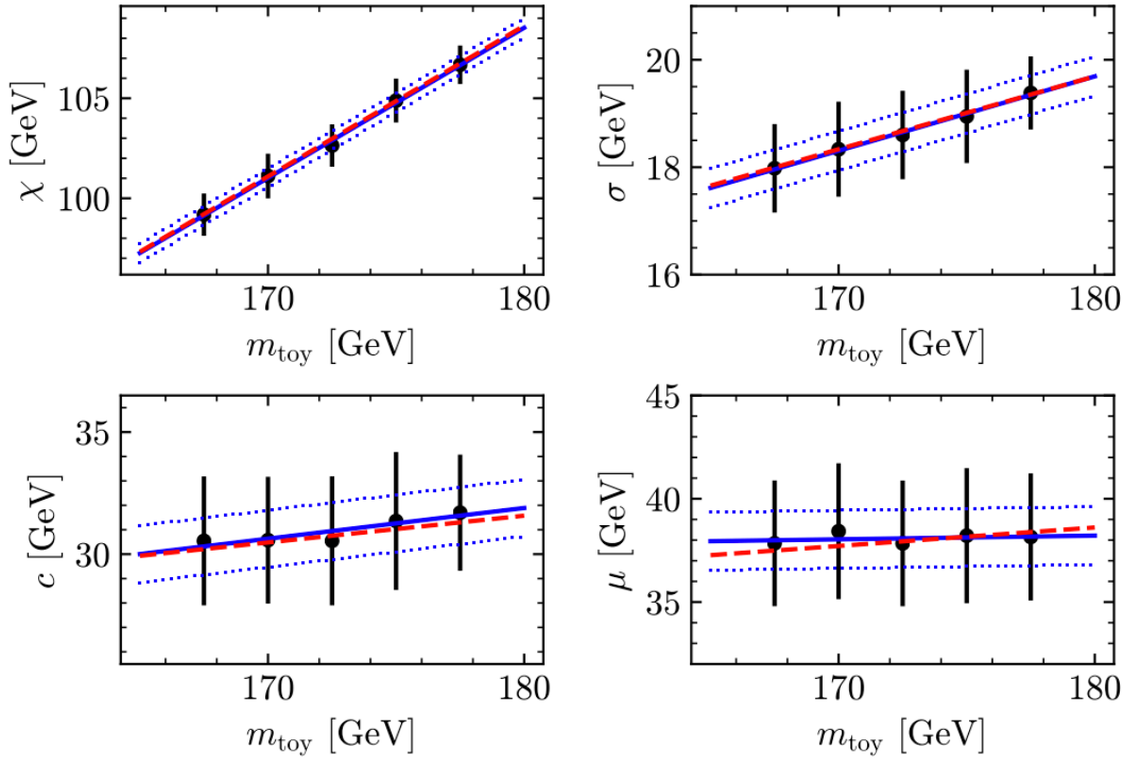


4) Interpolate the function parameters as a function of the input mass
5) Go to real data and find the mass which fits the best

How is the top mass measured?

Template Method

1) Pick an observable which is correlated with the Monte Carlo top mass
2) Get distribution of observable at multiple values of the mass
3) Fit each distribution with a parametric function
4) Interpolate the function parameters as a function of the input mass

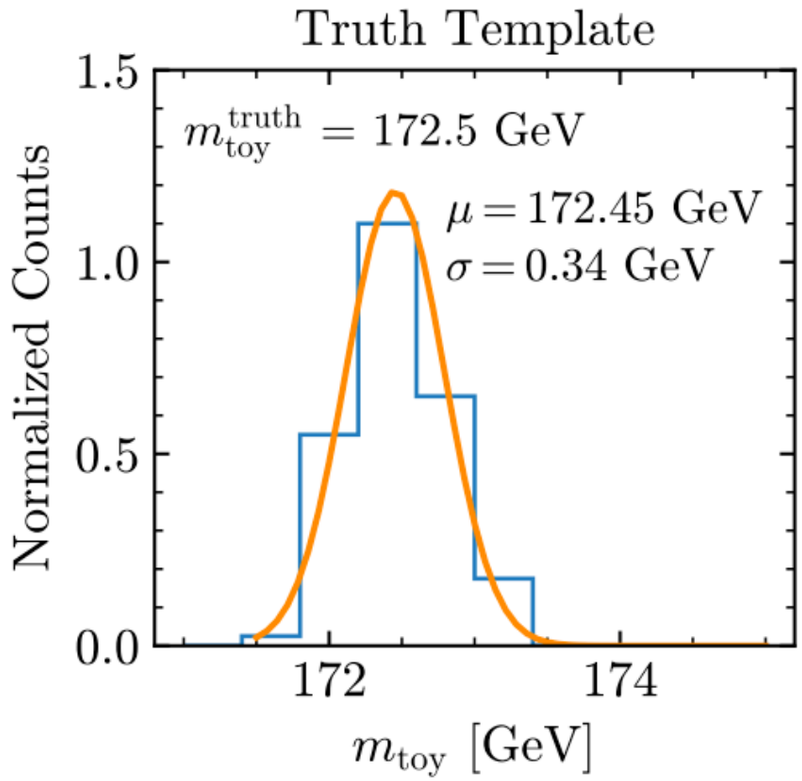


5) Go to real data and find the mass which fits the best

How is the top mass measured?

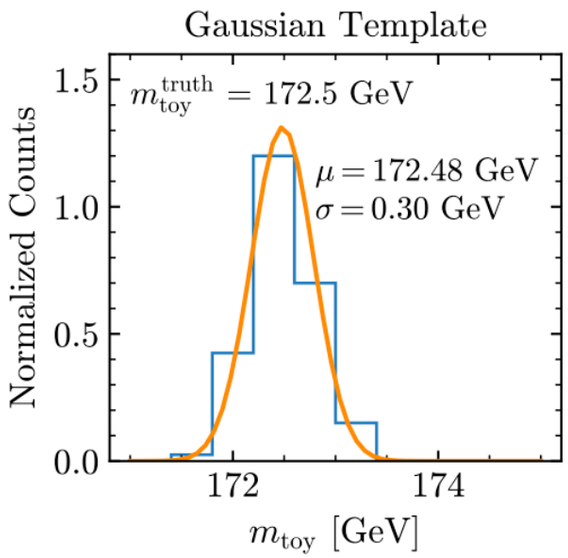
Template Method

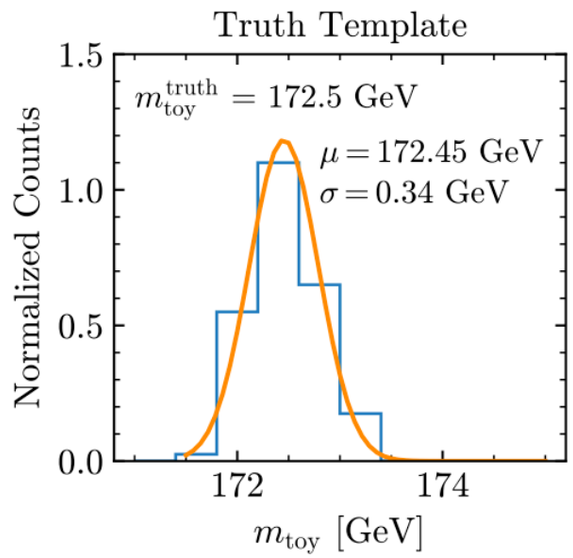
1) Pick an observable which is correlated with the Monte Carlo top mass
2) Get distribution of observable at multiple values of the mass
3) Fit each distribution with a parametric function
4) Interpolate the function parameters as a function of the input mass
5) Go to real data and find the mass which fits the best



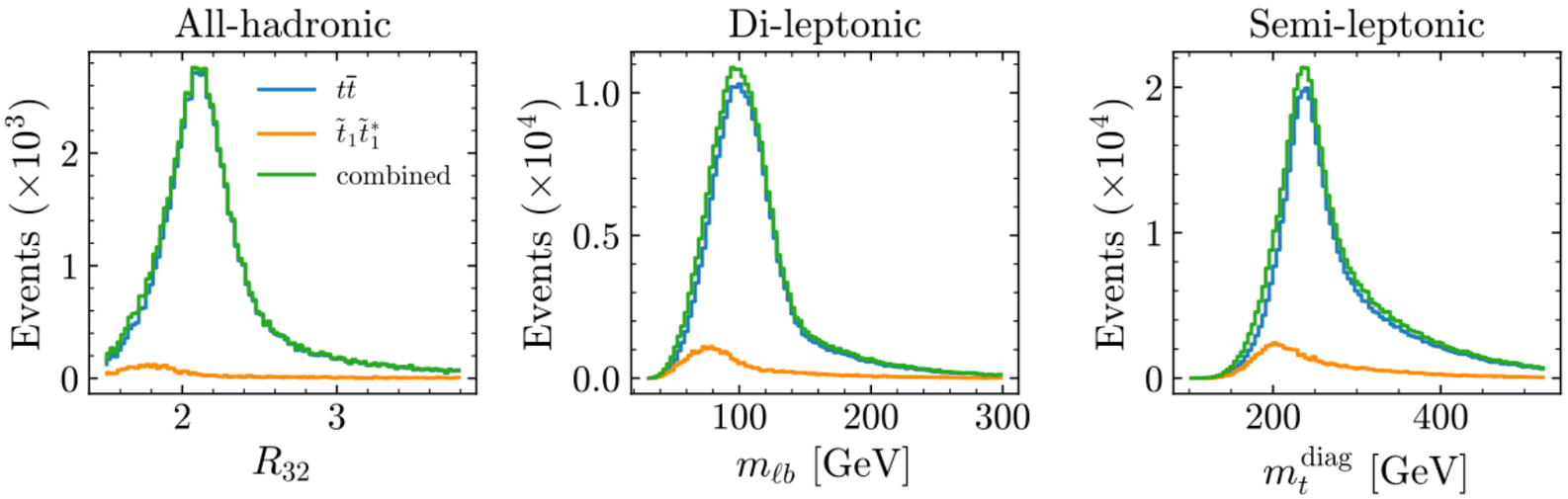
Comments

1) Unfortunately robust: the template/parametric fitting function does not need to be a good fit to the data

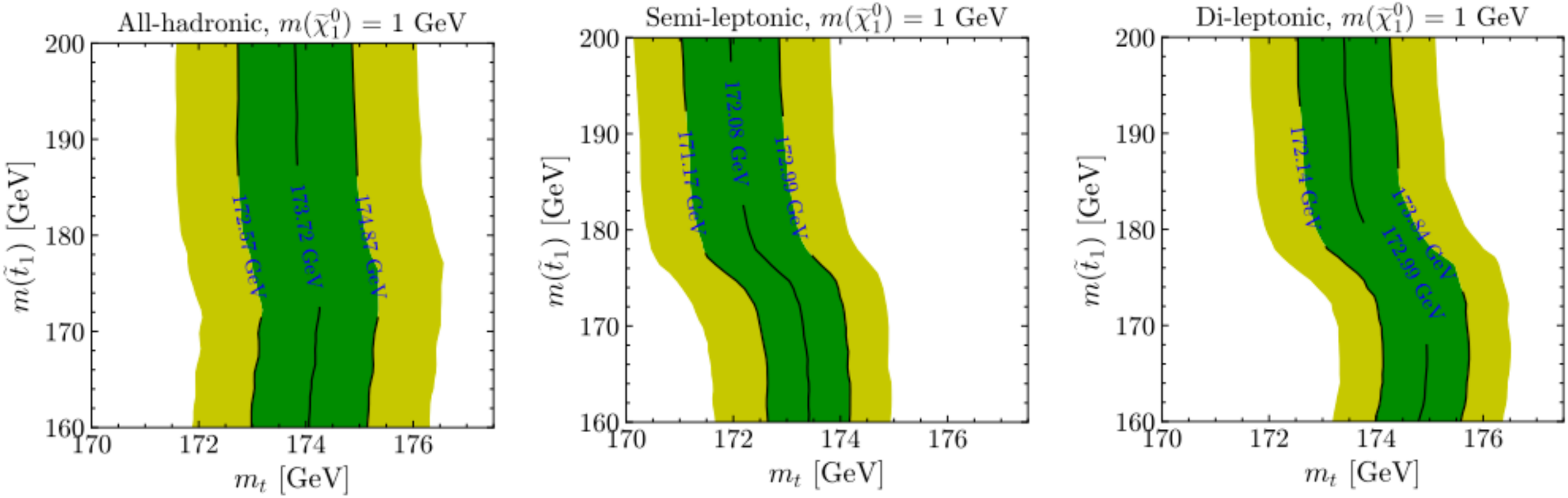




2) Without worrying about overall fit, become susceptible to contamination



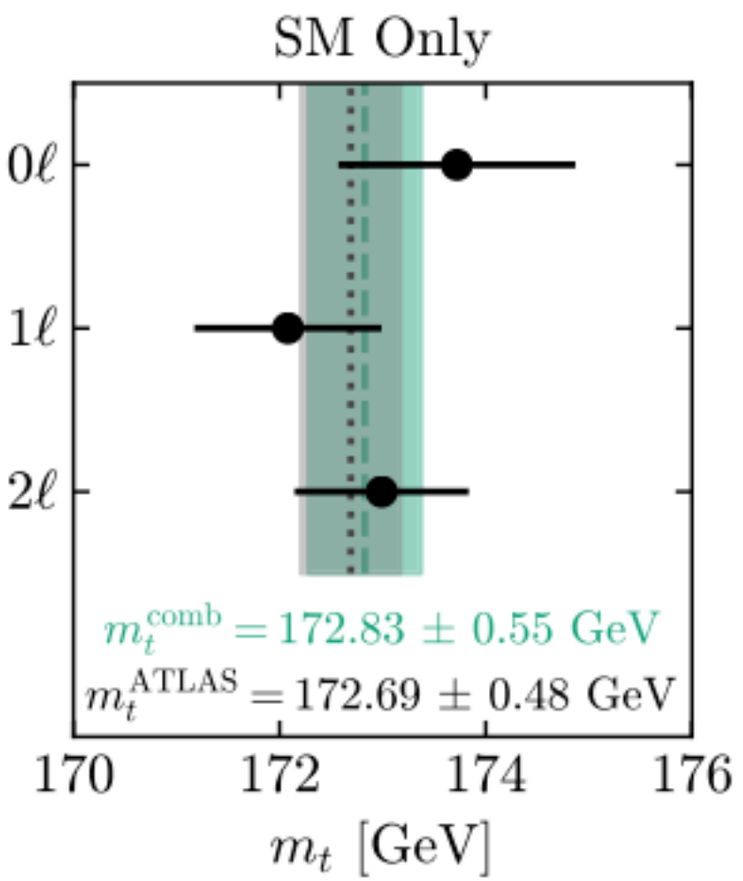
Contamination Results

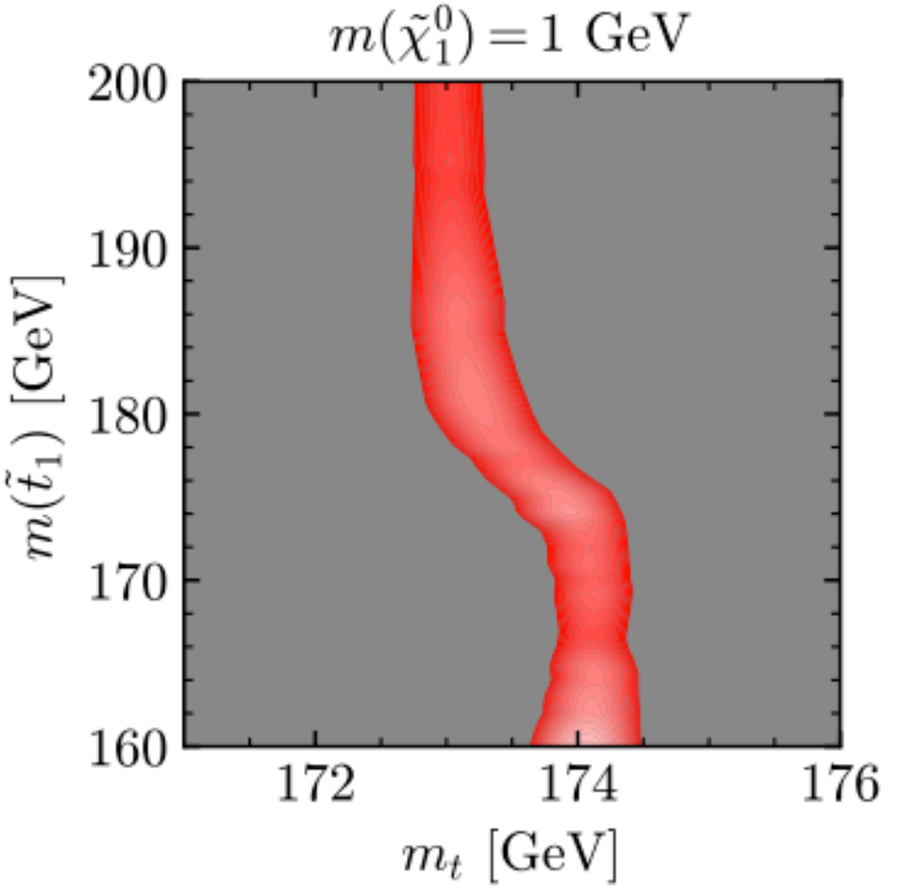


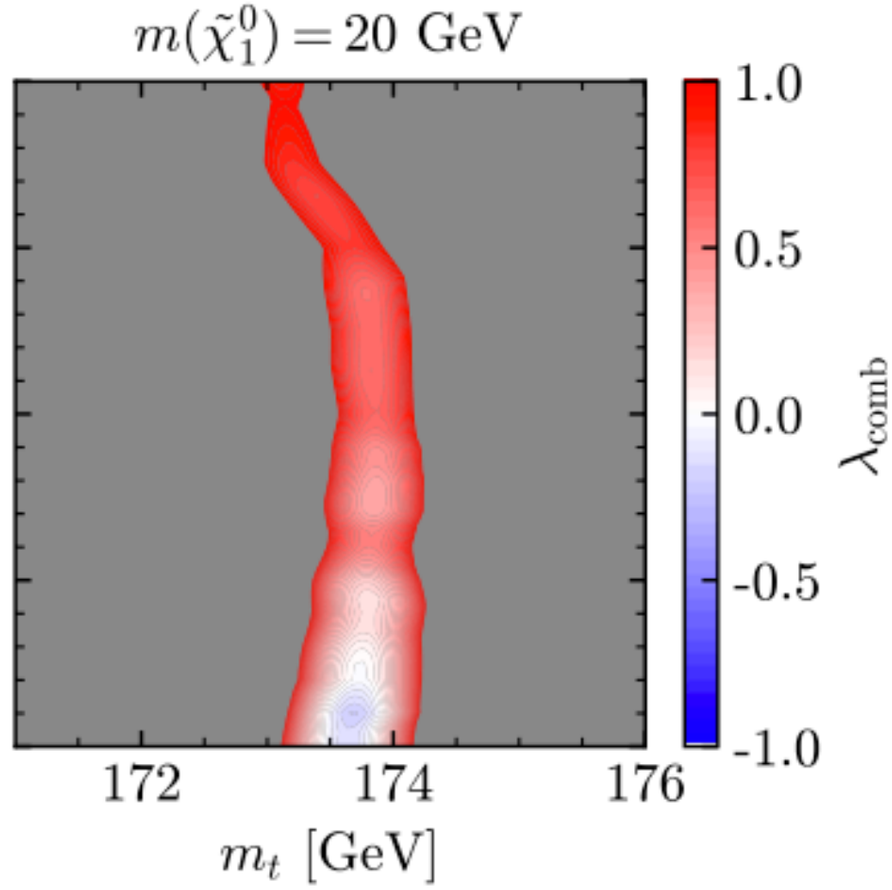
Cohen, Majewski, BO, and Zheng [1909.09670]

- The measured mass in each channel is different
- Channels with missing energy are affected more
- Largest effect comes when stops decay through off-shell top

Contamination Results







Cohen, Majewski, BO, and Zheng [1909.09670]

Stop contamination affects the three channels differently

Reducing uncertainty (and central values converging) will make the stop hypothesis inconsistent with the data

Reducing uncertainty with ML

F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

Reducing uncertainty with ML

F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

- Template method finds parameterized model which best fits a distribution of a single observable

Reducing uncertainty with ML

F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

- Template method finds parameterized model which best fits a distribution of a single observable

- Uncertainty due to ISR/FSR estimated by altering Monte Carlo parameters determining the change in extracted mass

Reducing uncertainty with ML

F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

- Template method finds parameterized model which best fits a distribution of a single observable

- Uncertainty due to ISR/FSR estimated by altering Monte Carlo parameters determining the change in extracted mass

- Uncertainty can be reduced by jet calibration and removing extra radiation through grooming

Reducing uncertainty with ML

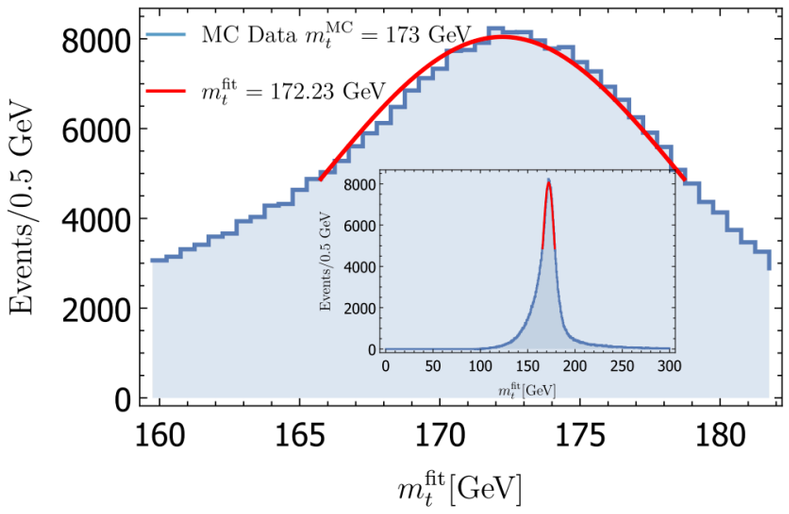
F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

- Template method finds parameterized model which best fits a distribution of a single observable
- Uncertainty due to ISR/FSR estimated by altering Monte Carlo parameters determining the change in extracted mass
- Uncertainty can be reduced by jet calibration and removing extra radiation through grooming
- Use NN as universal function; makes it easy to use multiple observables and may be able to use information which grooming throws out

Reducing uncertainty with ML

- Examine semi-leptonic $t\bar{t}$ production at $\sqrt{s} = 13\ \mathrm{TeV}$
- Use leptonic side to tag event, obtain mass measurement in hadronic side
- Anti-kt jets with R=0.5, $p_T^j > 30\ \mathrm{GeV}$, $|\eta| < 2.4$
- $150\ \mathrm{GeV} < \mathrm{m_{3j}} < 200\ \mathrm{GeV}$
- Use PYTHIA 8 with the A14 tunes:
  - 4 PDF sets
  - 5 families of tune variations
- Compute $\Delta m_t^{\mathrm{MC}} \equiv \Delta m_t^{\mathrm{fit}} \frac{m_t^{\mathrm{MC}}}{m_t^{\mathrm{fit}}}$, where $\Delta m_t^{\mathrm{fit}}$ is half of spread in fit mass using different tunes

Reducing uncertainty with ML



Andreassen and Schwartz [1705.07135]

Fitting the 3-jet invariant mass peak

tunes 19-22

tunes 21, 23, 24

tunes 21, 25, 26

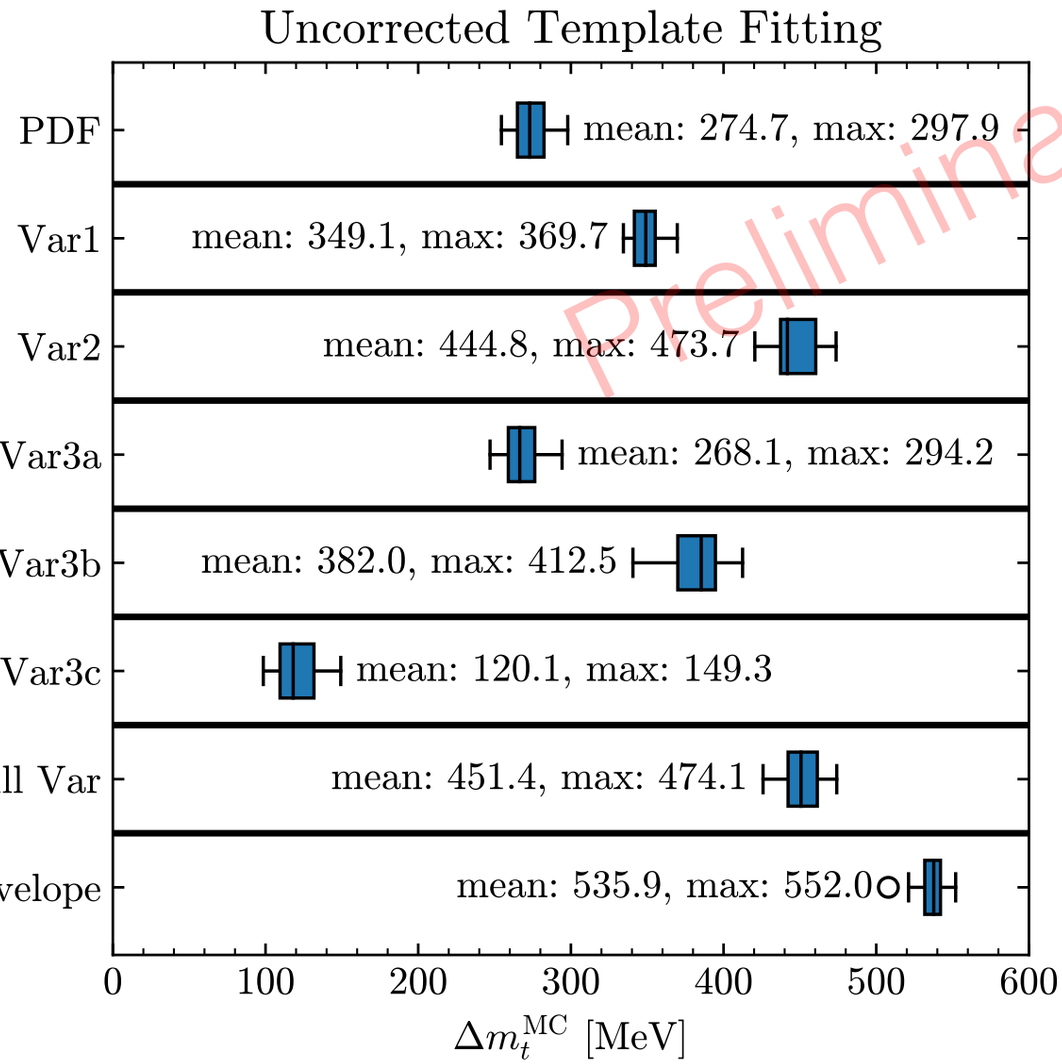
tunes 21, 27, 28

tunes 21, 29, 30

tunes 21, 31, 32

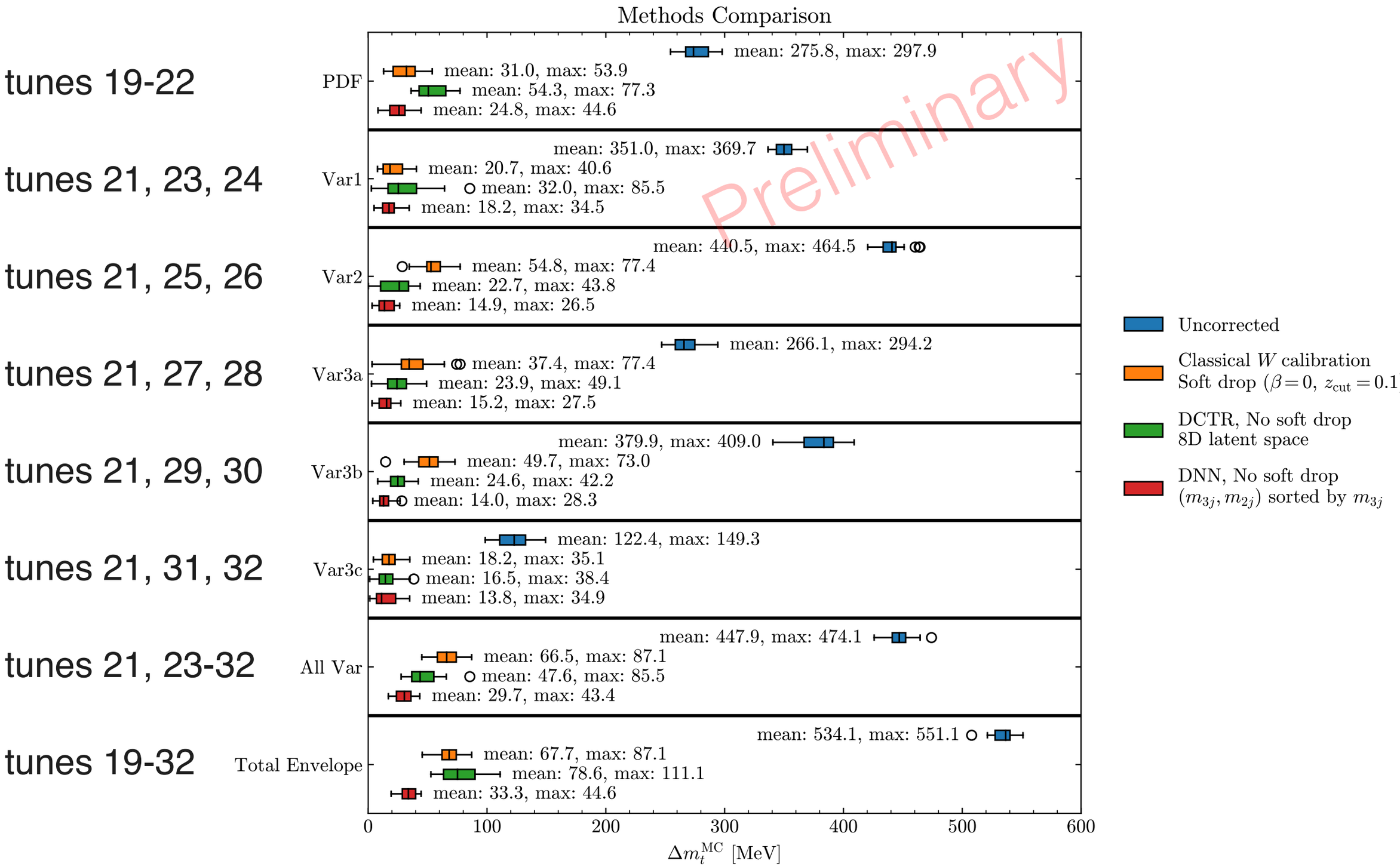
tunes 21, 23-32

tunes 19-32



F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Schwartz [in progress]

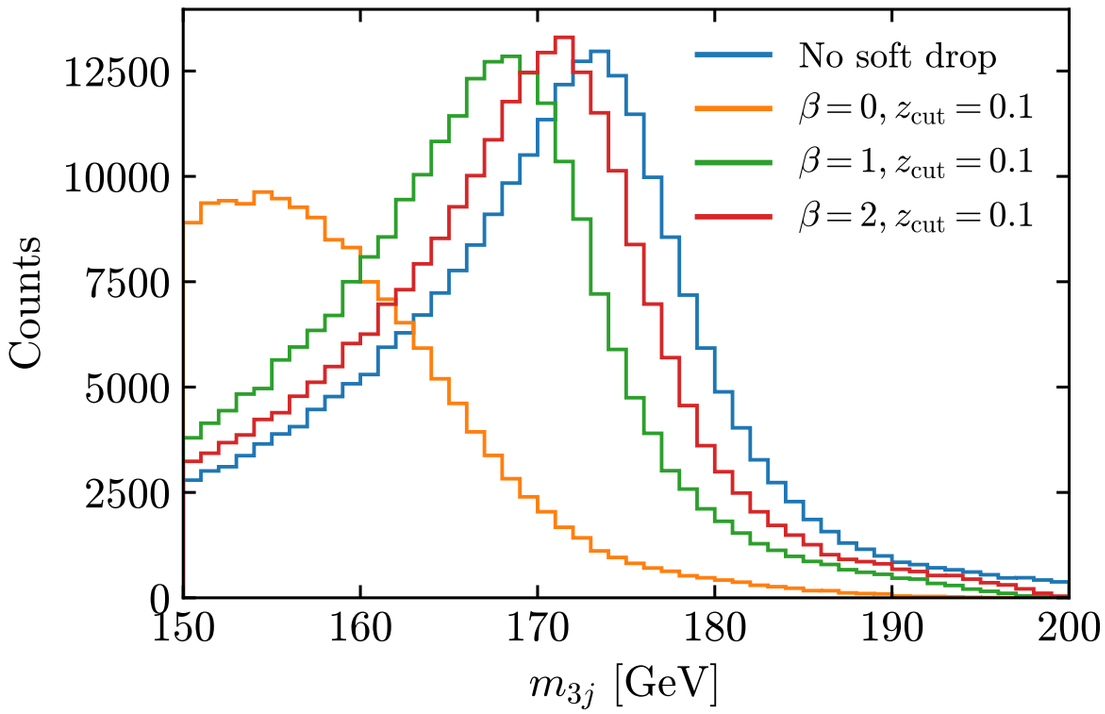
Reducing uncertainty with ML



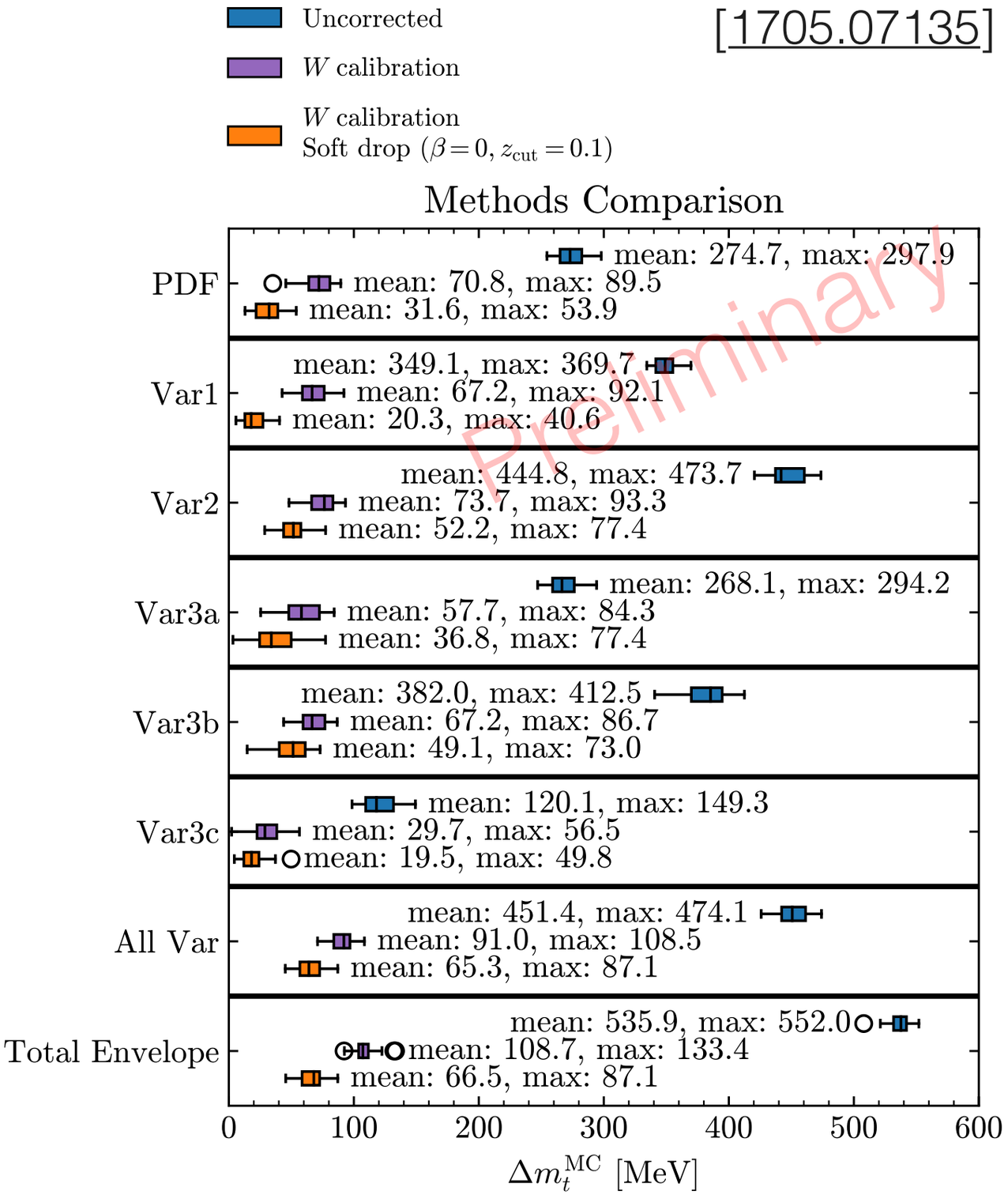
F. Flesher, K. Fraser, C. Hutchinson, BO, and M. Shwartz [in progress]

W calibration and soft drop

$$m_t^{\text{fit}} \to m_t^{\text{fit}} \frac{m_W^{\text{MC}}}{m_W^{\text{fit}}}$$

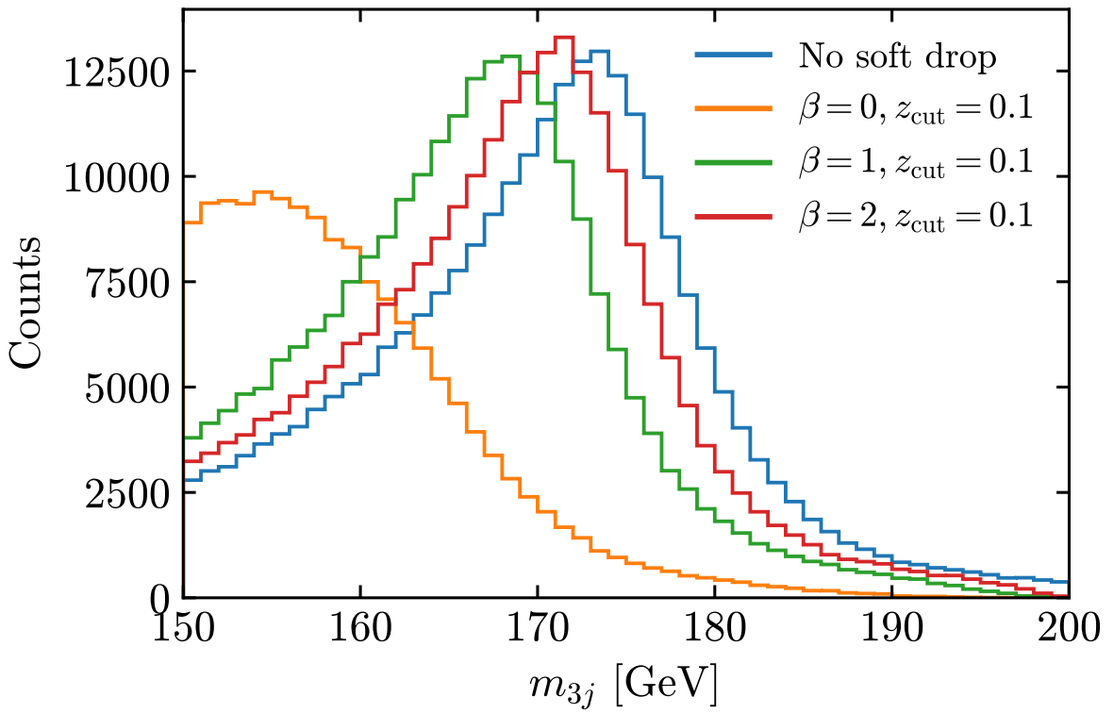


Andreassen and Schwartz [1705.07135]



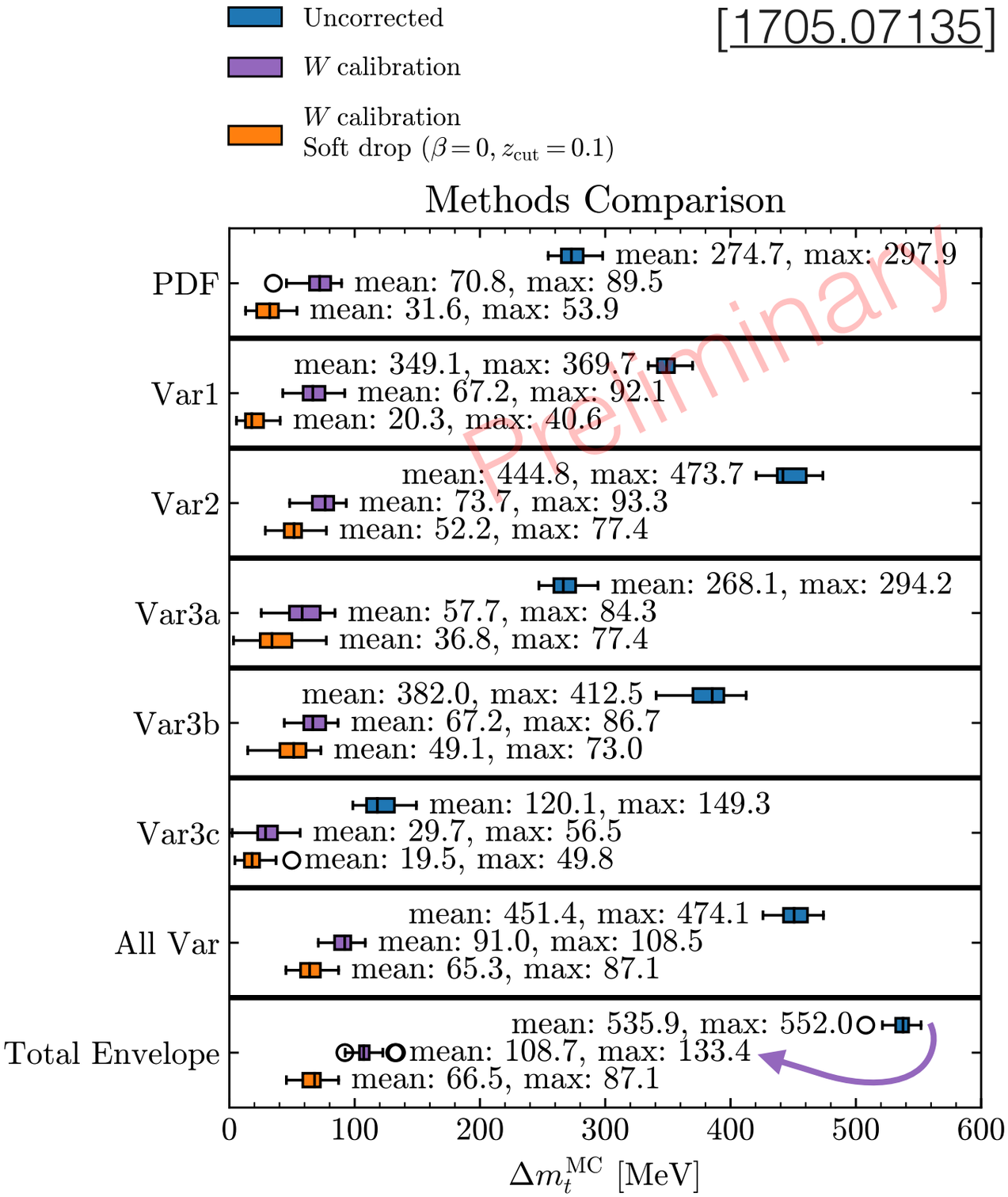
W calibration and soft drop

$$m_t^{\text{fit}} \rightarrow m_t^{\text{fit}} \frac{m_W^{\text{MC}}}{m_W^{\text{fit}}}$$



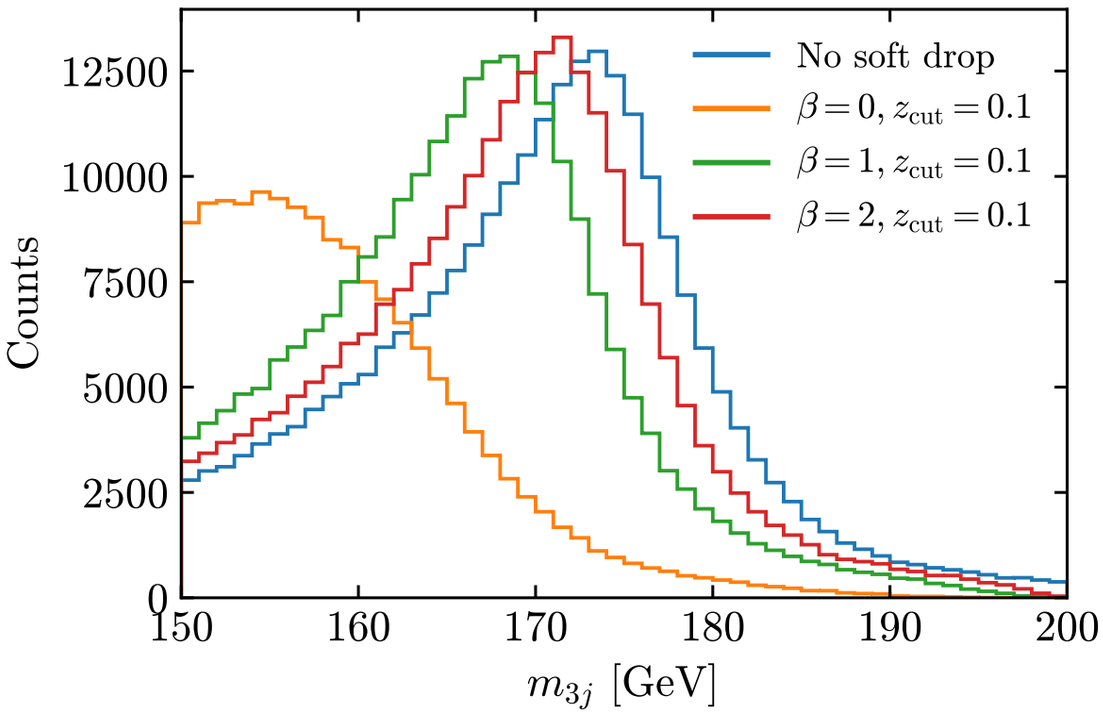
W calibration reduces uncertainty by 76%

Andreassen and Schwartz [1705.07135]



W calibration and soft drop

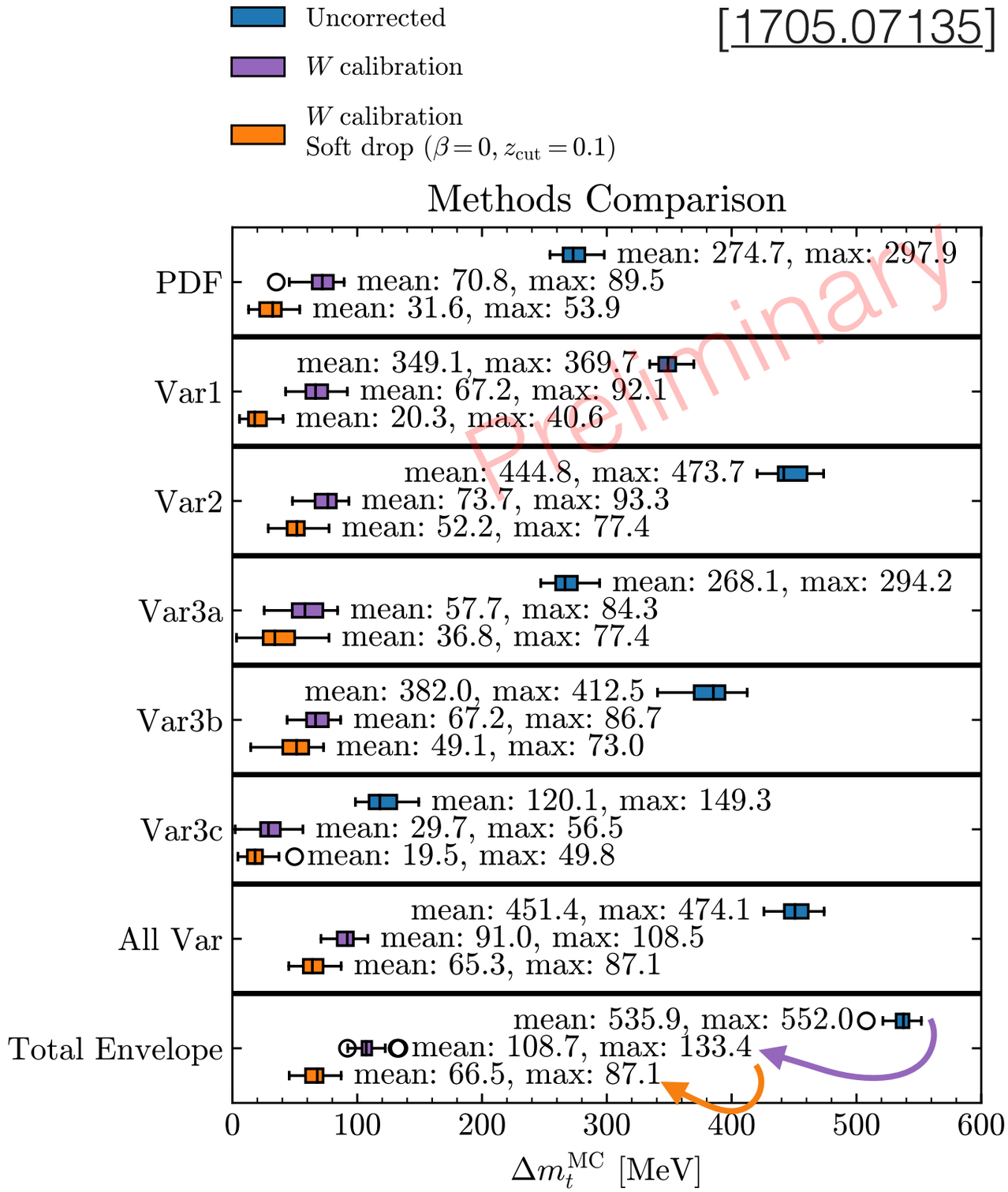
$$m_t^{\text{fit}} \rightarrow m_t^{\text{fit}} \frac{m_W^{\text{MC}}}{m_W^{\text{fit}}}$$



W calibration reduces uncertainty by 76%

Additional soft drop reduces uncertainty by another 35%

Andreassen and Schwartz [1705.07135]



DCTR

***D**eep neural networks using **C**lassification for **T**uning and **R**eweighting* [Andreassen and Nachman [*1907.08209*]]

Use parameterized neural network classifier to learn likelihood ratio [use binary cross entropy loss function]

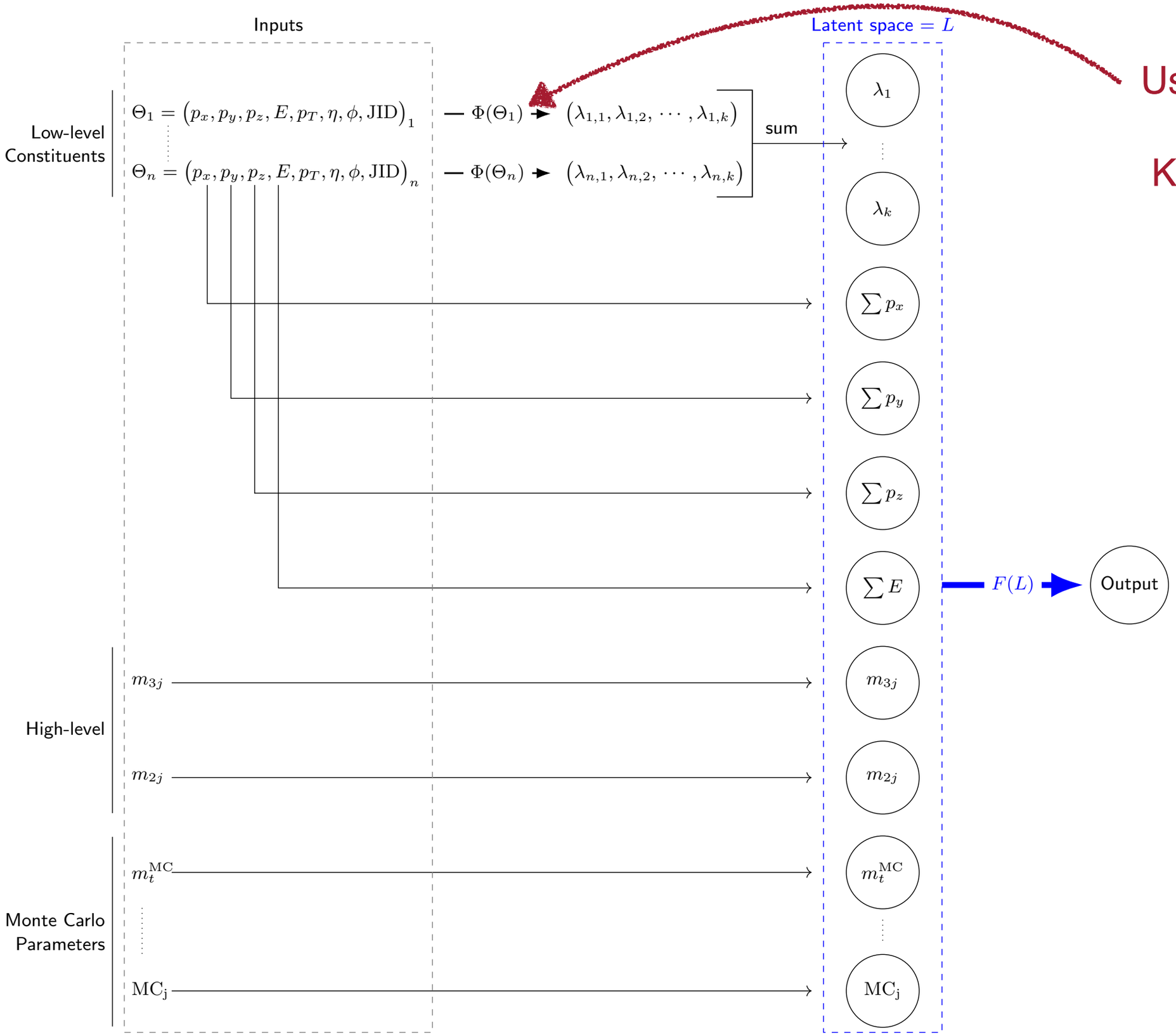
$$L(y, y^p) = \frac{-1}{n} \sum_{i=1}^{n} \left(y_i \log(y_i^p) + (1 - y_i) \log\left(1 - y_i^p\right) \right)$$

Classify as reference sample or any other mass/tune

Infer mass by computing the loss on unknown sample and reference sample

Use combination of high-level and low-level variables

DCTR



Use ParticleFlowNetwork on low-level inputs, Komiske, Metodiev, and Thaler [1810.05165]

Also include high-level information

Parameterize with tune parameters

DCTR

Reference Set

$$m_t^{\mathrm{MC}} = 175 \text{ Gev}$$

$$\text{Color reconnection range} = 1.71$$

$$\alpha_s^{\mathrm{MPI}} = 0.127$$

$$\text{Space shower } \alpha_s = 0.127$$

$$\text{Space shower } p_{t0}^{\mathrm{Ref}} = 1.51$$

$$\text{Space shower } p_{t,\mathrm{damp}}^{\mathrm{Fudge}} = 1.04$$

$$\text{Space shower } p_{t,\mathrm{max}}^{\mathrm{Fudge}} = 0.88$$

$$\text{Time shower } \alpha_s = 0.124$$

$$10^6 \text{ events}$$

Sampling Set

$$m_t^{\mathrm{MC}} \in (170 \text{ GeV}, 176 \text{ GeV})$$

$$\text{Color reconnection range} \in (1.67, 1.75)$$

$$\alpha_s^{\mathrm{MPI}} \in (0.116, 0.136)$$

$$\text{Space shower } \alpha_s \in (0.1025, 0.1525)$$

$$\text{Space shower } p_{t0}^{\mathrm{Ref}} \in (1.415, 1.755)$$

$$\text{Space shower } p_{t,\mathrm{damp}}^{\mathrm{Fudge}} \in (0.715, 1.575)$$

$$\text{Space shower } p_{t,\mathrm{max}}^{\mathrm{Fudge}} \in (0.75, 1.085)$$

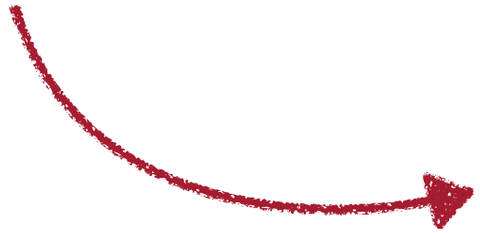
$$\text{Time shower } \alpha_s \in (0.097, 0.153)$$

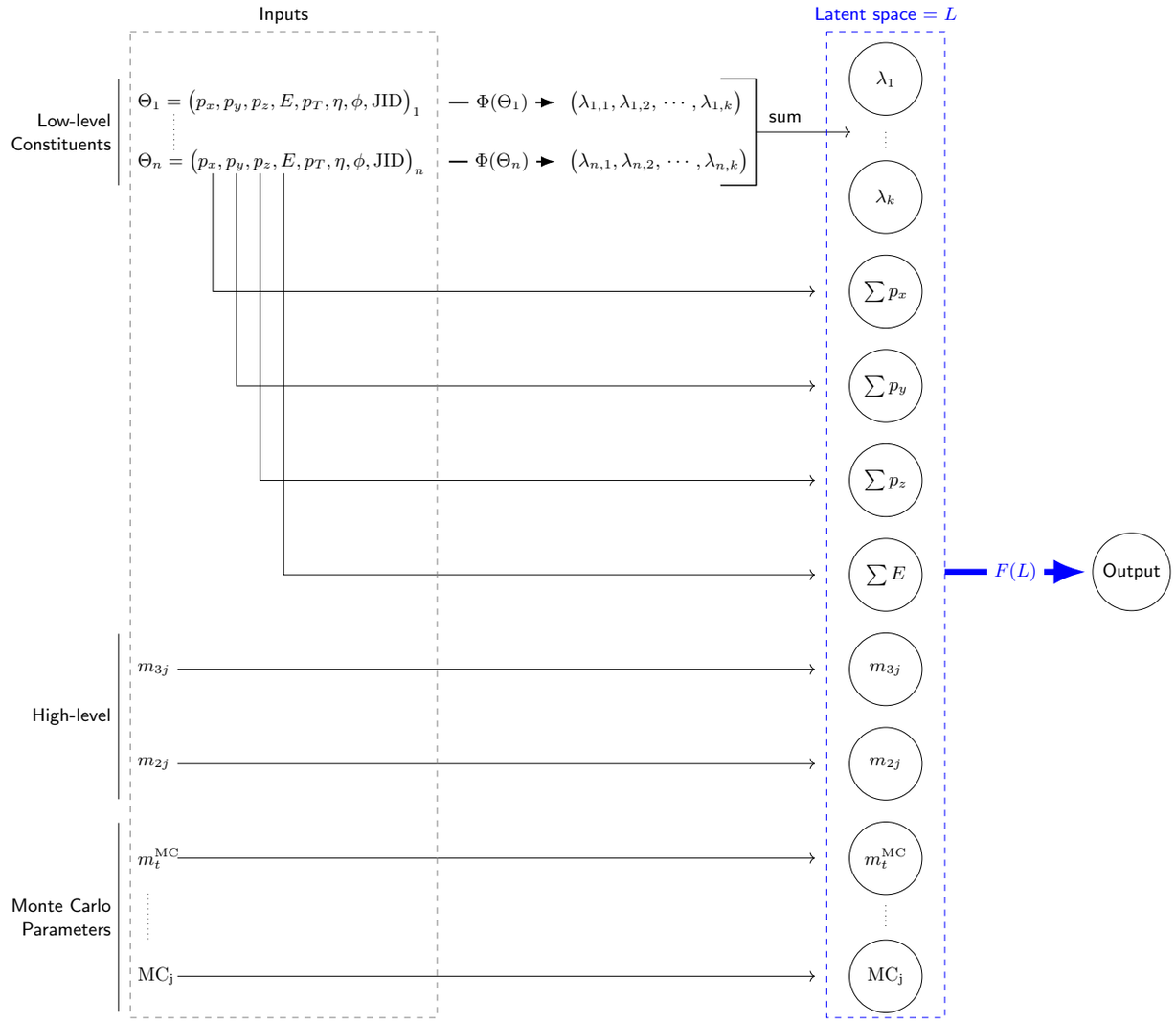
$$10^6 \text{ events}$$

ParticleFlowNetwork on jet constituents helps network adapt to different tune parameters

Evaluate (and minimize) loss on new dataset compared with reference set

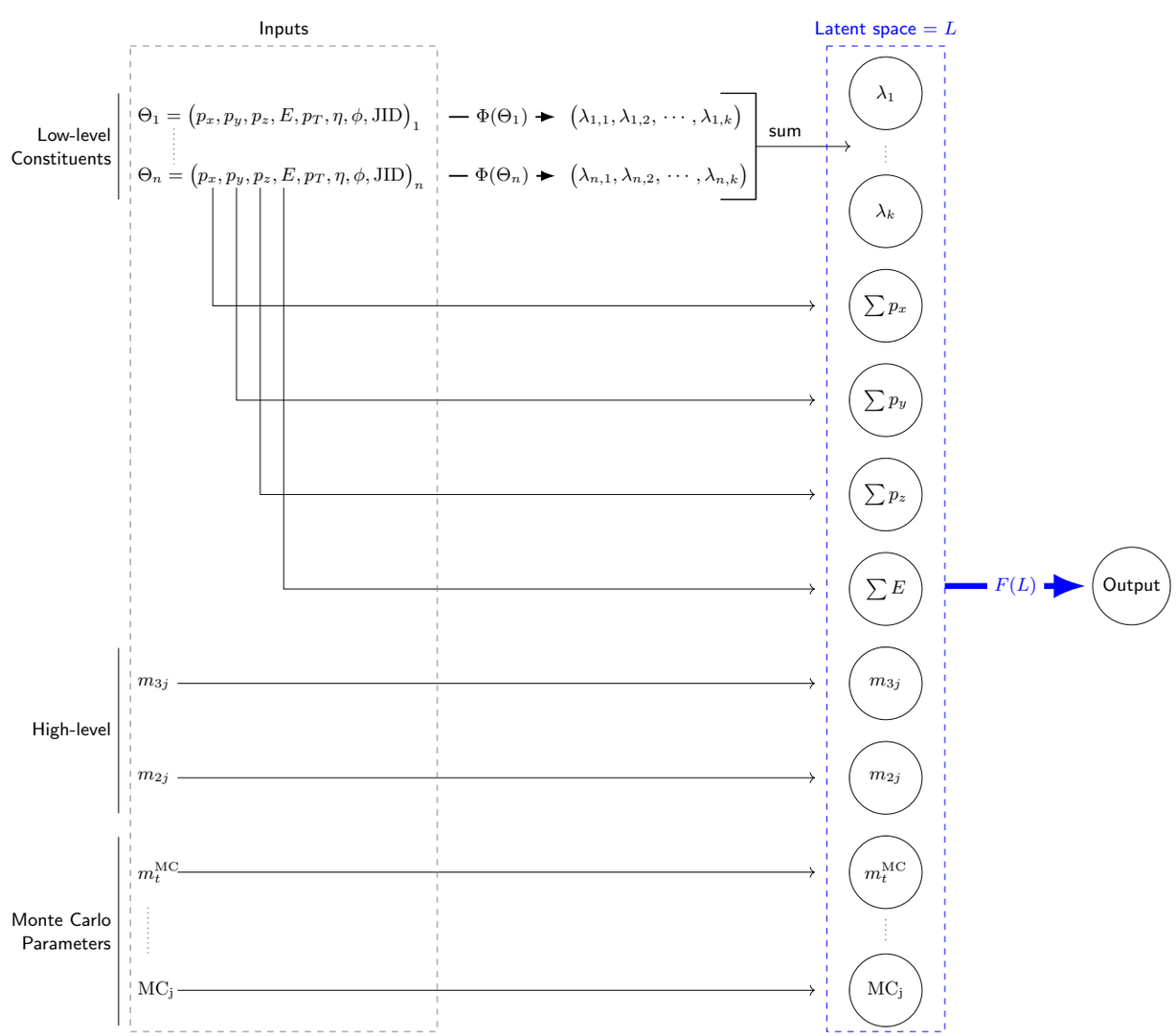
Feed in same events and only change the parameterized input

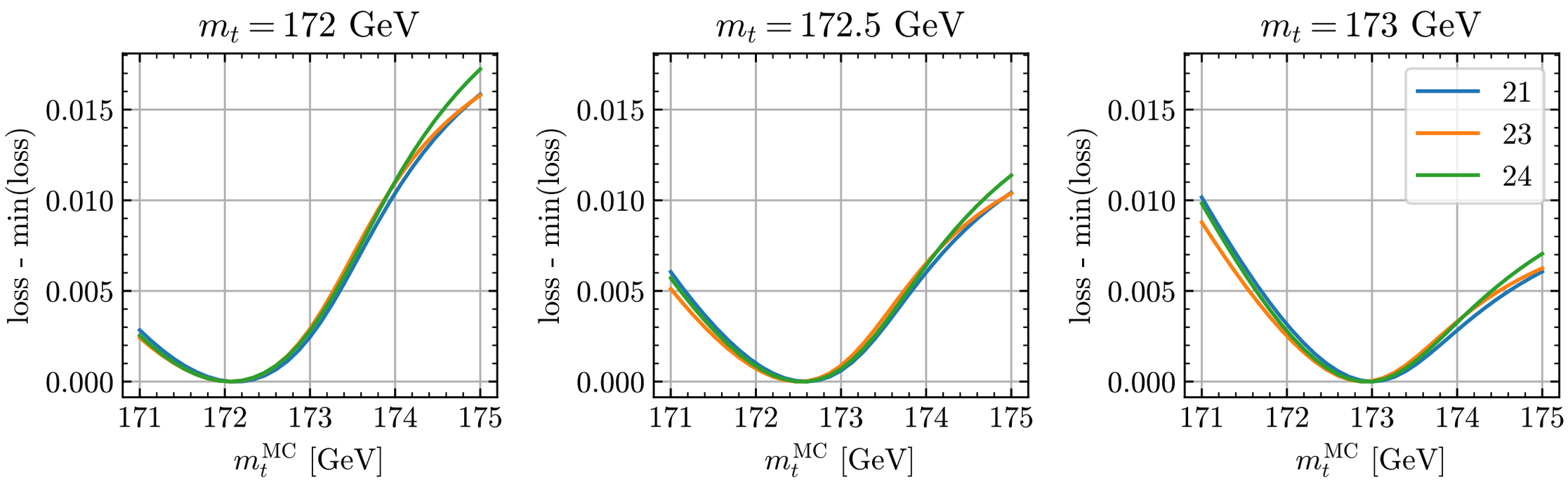




Evaluate (and minimize) loss on new dataset compared with reference set

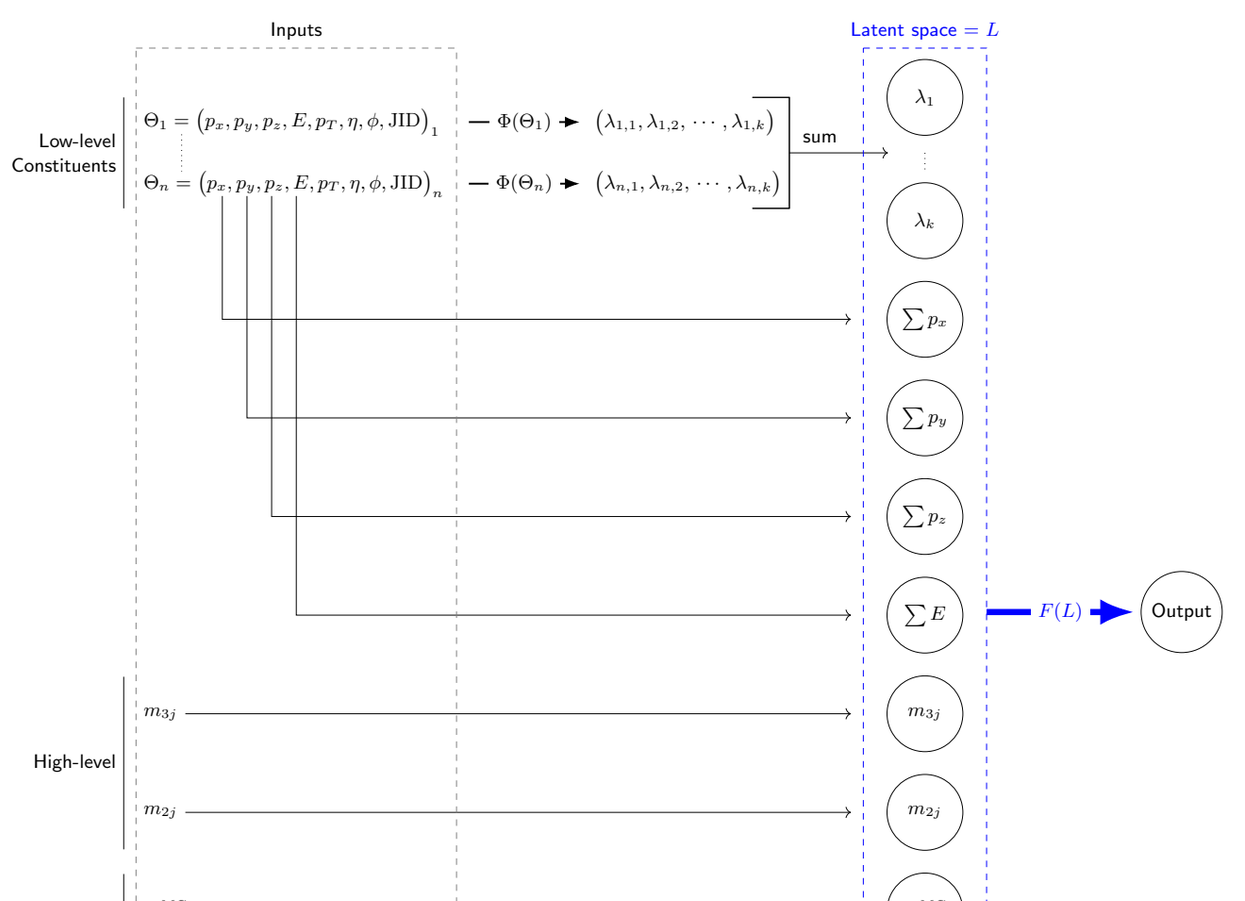
Feed in same events and only change the parameterized input

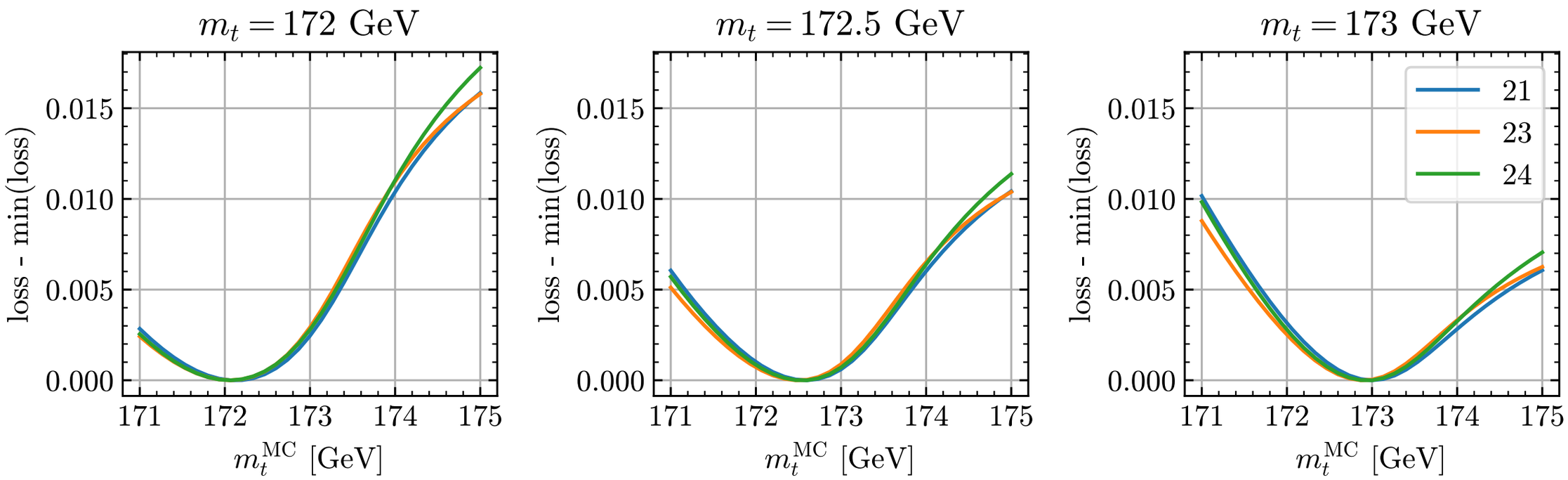




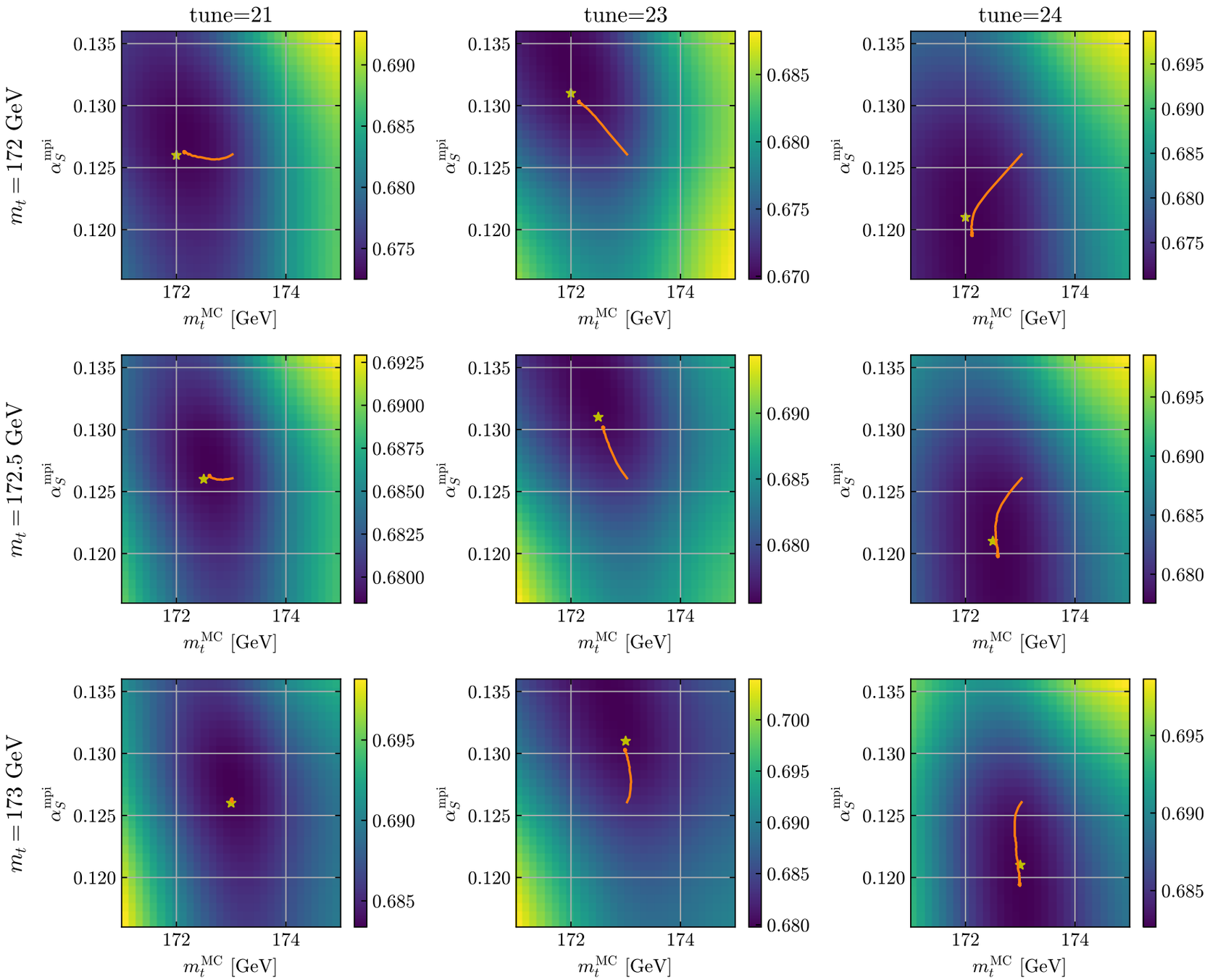
Evaluate (and minimize) loss on new dataset compared with reference set

Feed in same events and only change the parameterized input





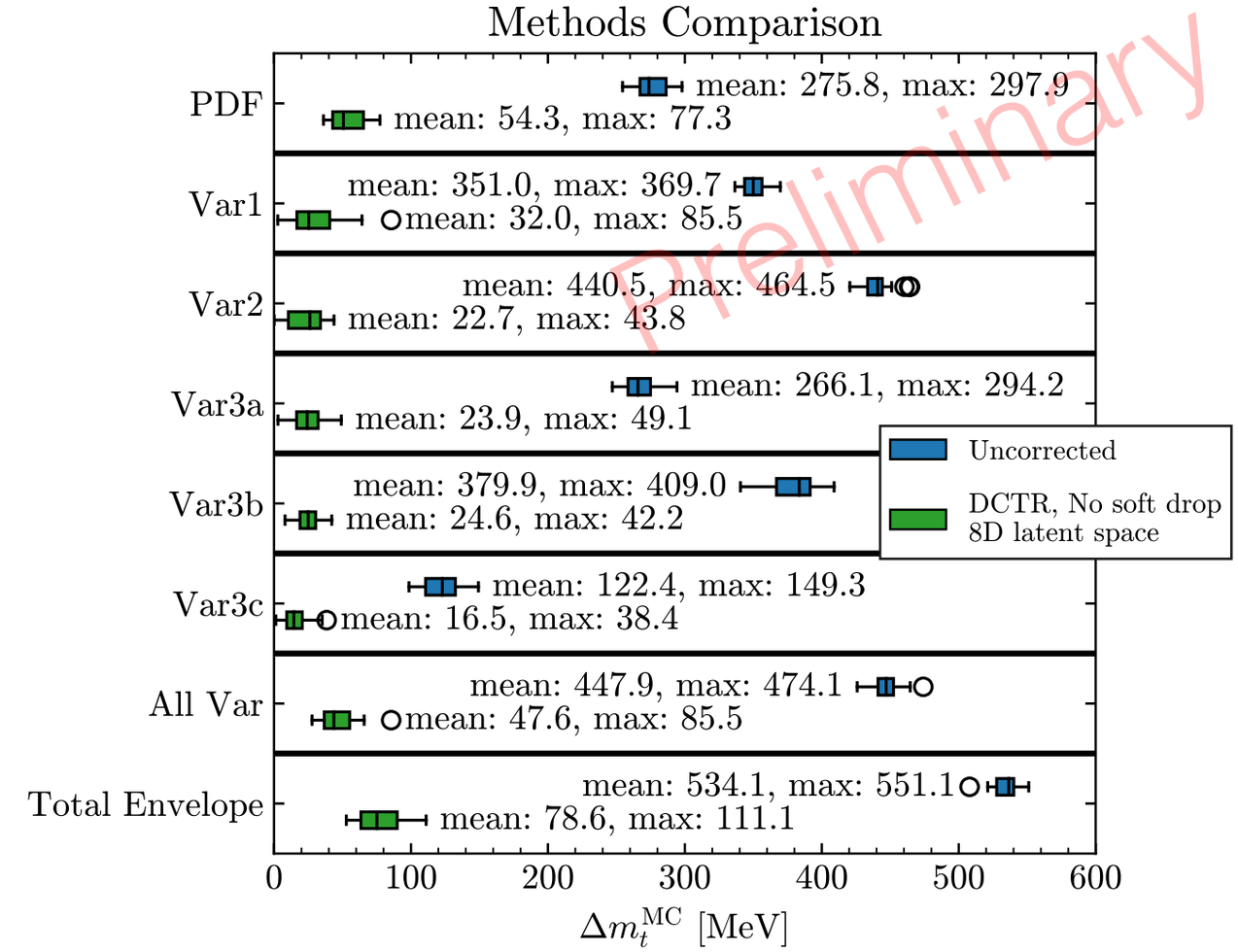
DCTR



Mass and tune parameters extracted by gradient descent minimization

DCTR

Not trained for PDF variations:
how to parameterize in network?

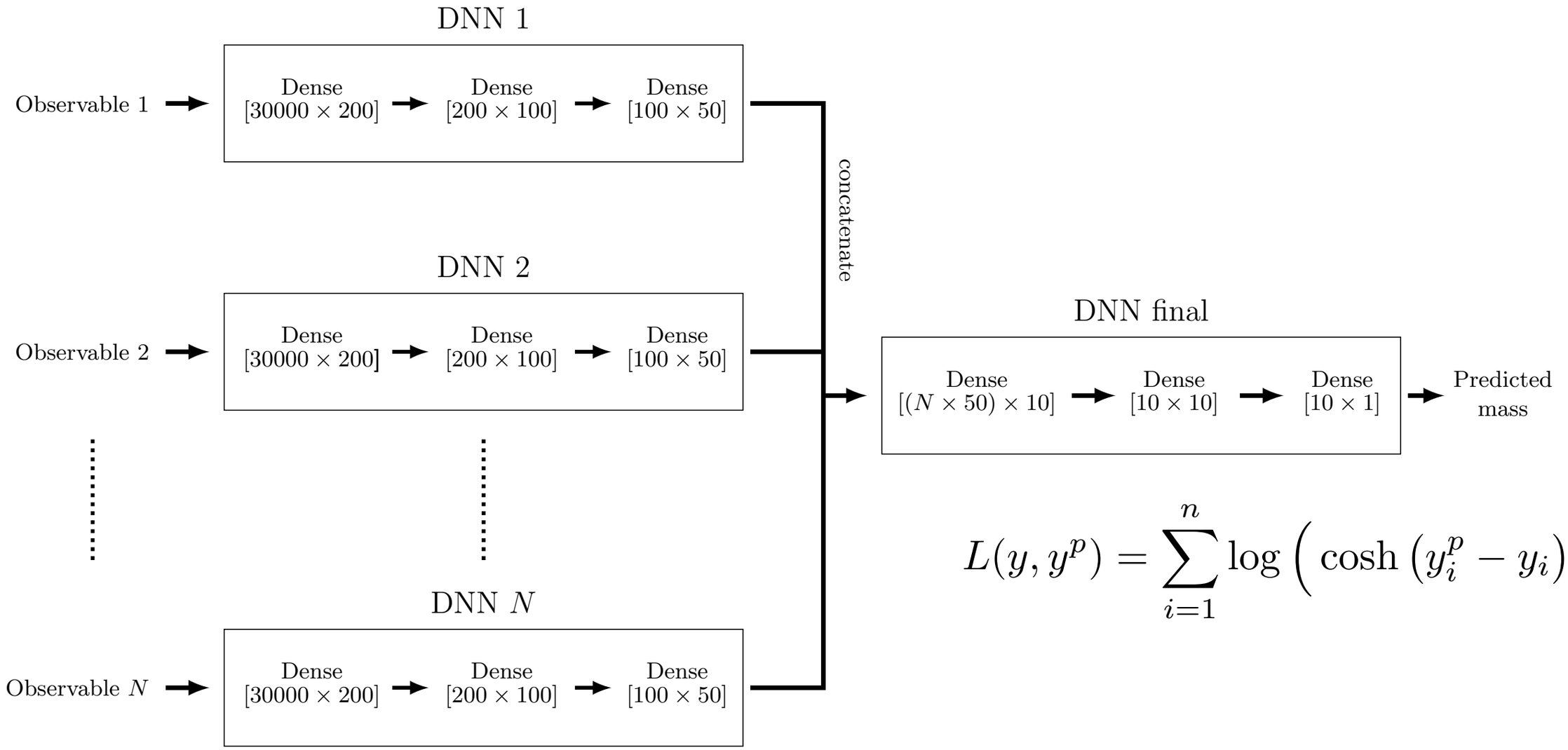


(89%, 82%) reduction in (mean, max) uncertainty

(85%, 80%) reduction in (mean, max) uncertainty

- Soft drop makes inference more challenging
- Similar to profiling over nuisance parameters
- Training is finicky (1/5 random initializations do not learn)
- Empirically find that only works for a small mass window

Deep Neural Network



$$L(y, y^p) = \sum_{i=1}^{n} \log\left(\cosh\left(y_i^p - y_i\right)\right)$$

Training data: 100k events for each

$m_t \in (165\ \text{GeV}, 180\ \text{GeV})$ steps of $0.2\ \text{GeV}$

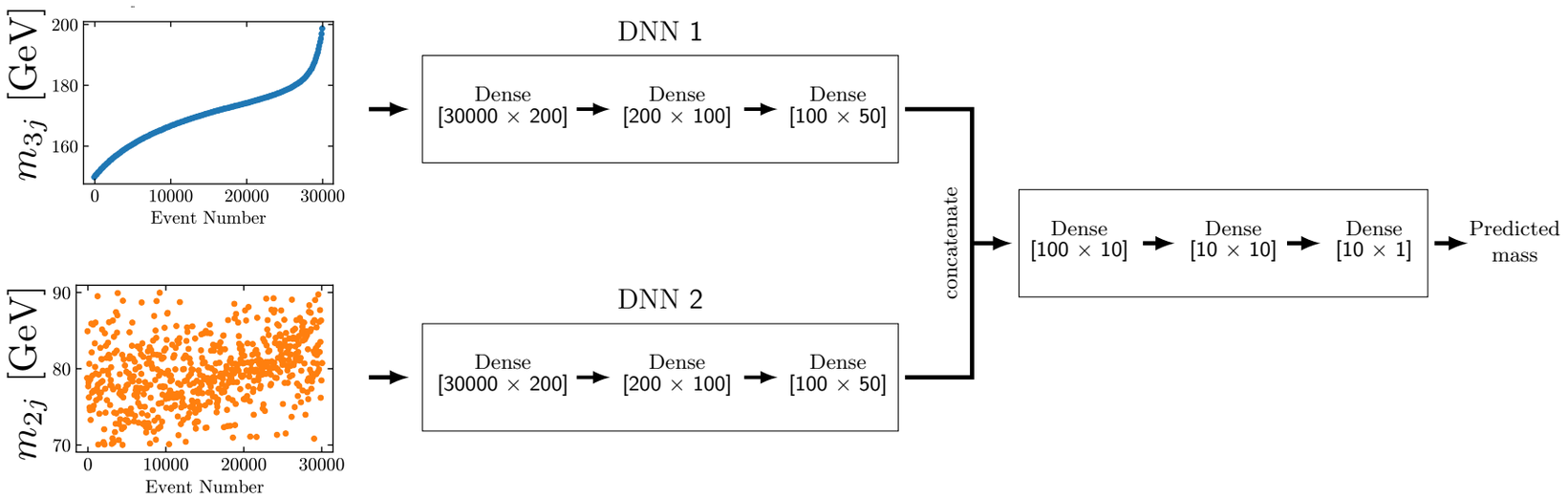
tunes 19-32

Randomly select 30k events and train on the whole distribution

Deep Neural Network

Test on $m_t = (172, 172.5, 173, 173.5, 174)$ GeV

Test set size: 400k (5 separate test sets per mass)
- Randomly select 30k -> get mass
- Repeat many times: mean is prediction



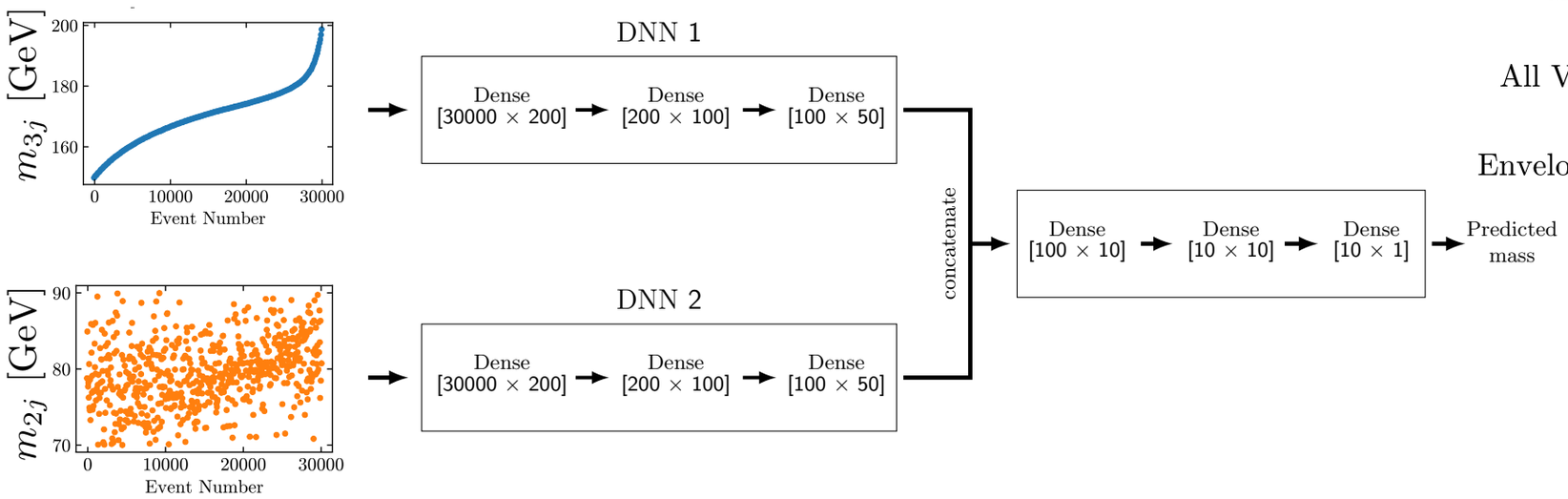
Deep Neural Network

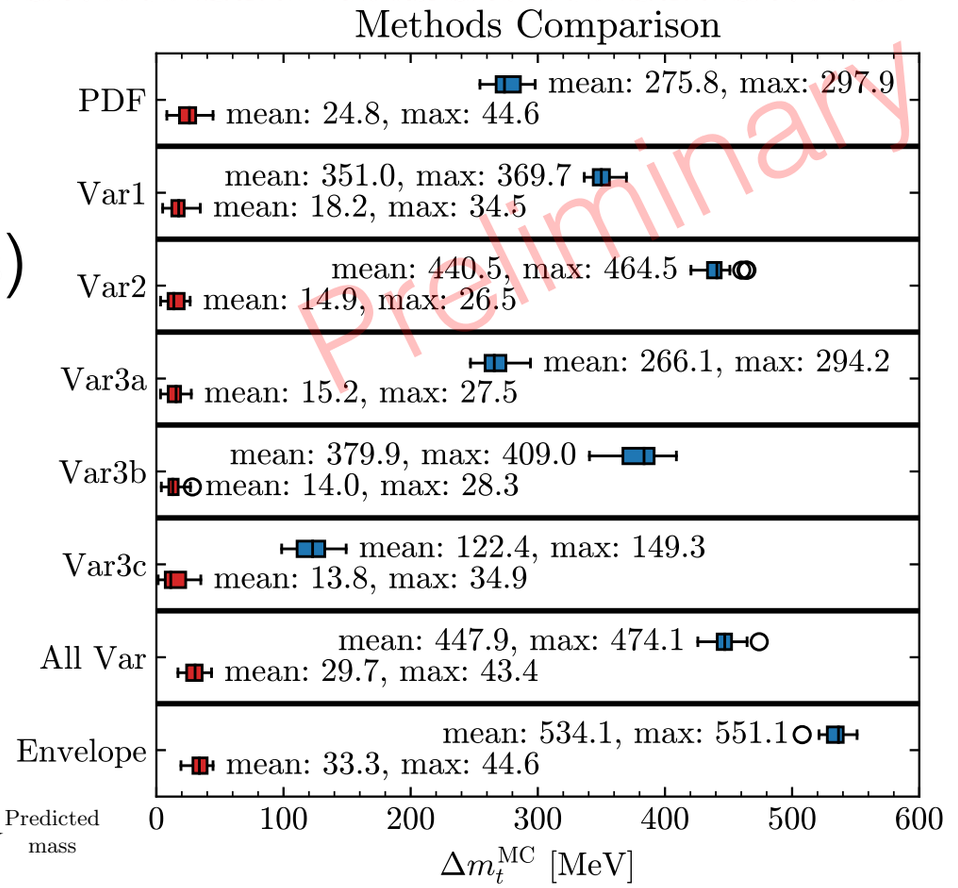
Test on $m_t = (172, 172.5, 173, 173.5, 174)$ GeV

Test set size: 400k (5 separate test sets per mass)

Randomly select 30k -> get mass

Repeat many times: mean is prediction





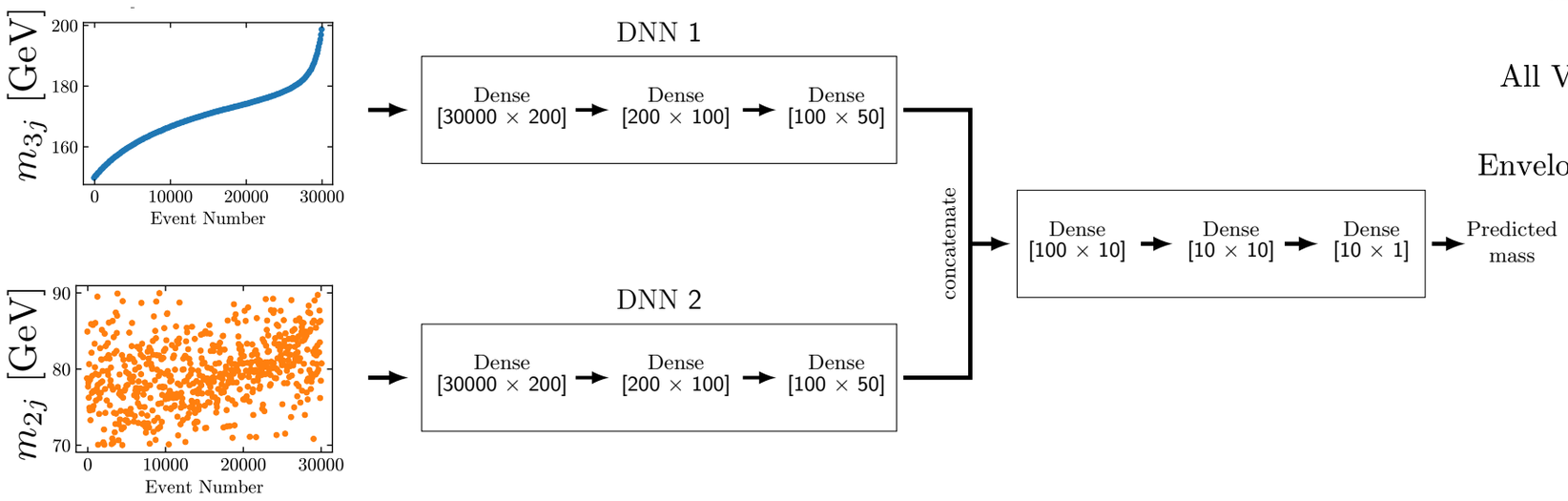
Deep Neural Network

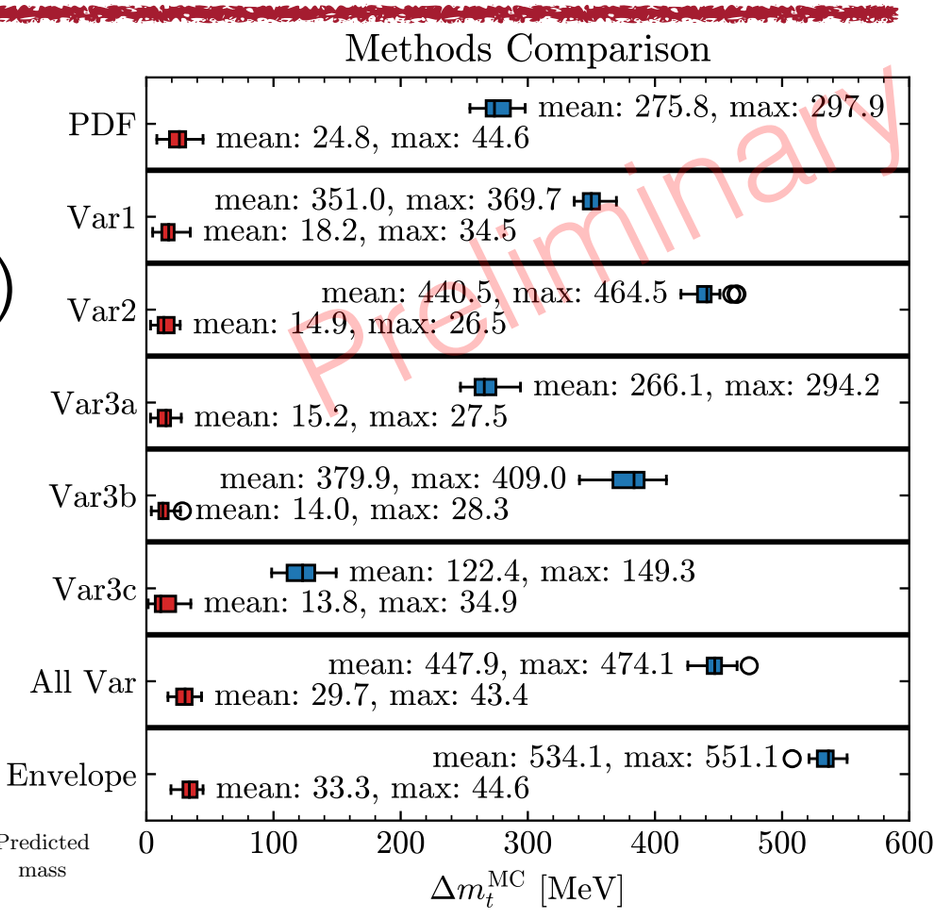
Test on $m_t = (172, 172.5, 173, 173.5, 174)$ GeV

Test set size: 400k (5 separate test sets per mass)

Randomly select 30k -> get mass

Repeat many times: mean is prediction





- Network is able to interpolate between masses
- Network learns distributions for the PDFS and tunes
- Grooming does not help
- Other observables did not help (2-jet mass useful for network to calibrate)

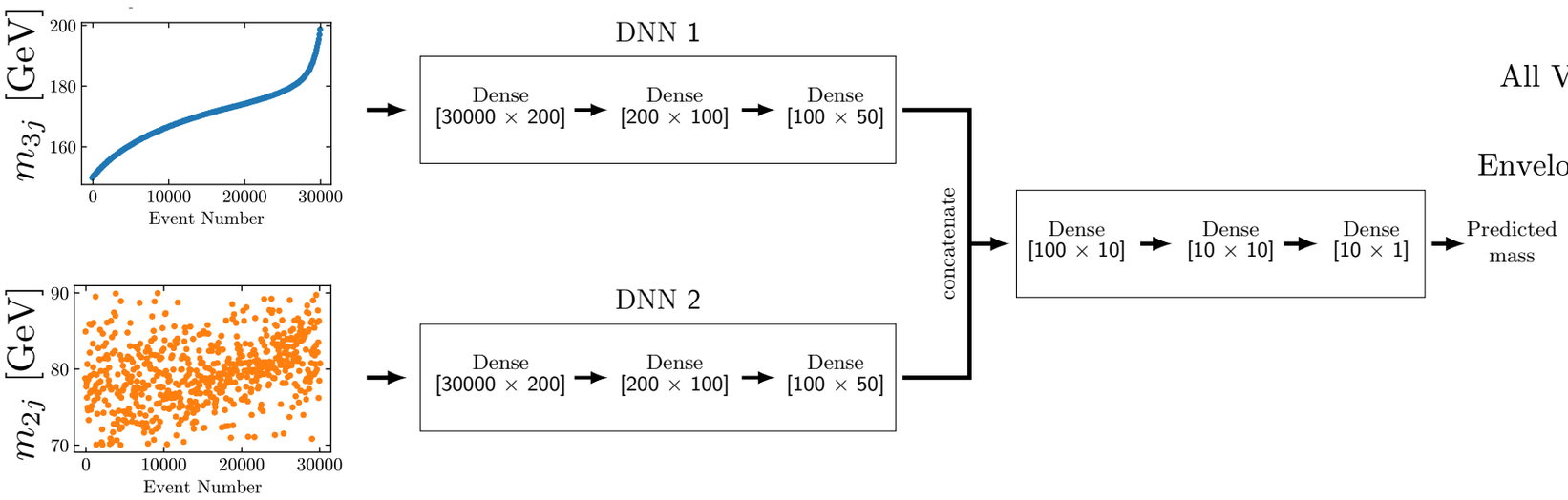
Deep Neural Network

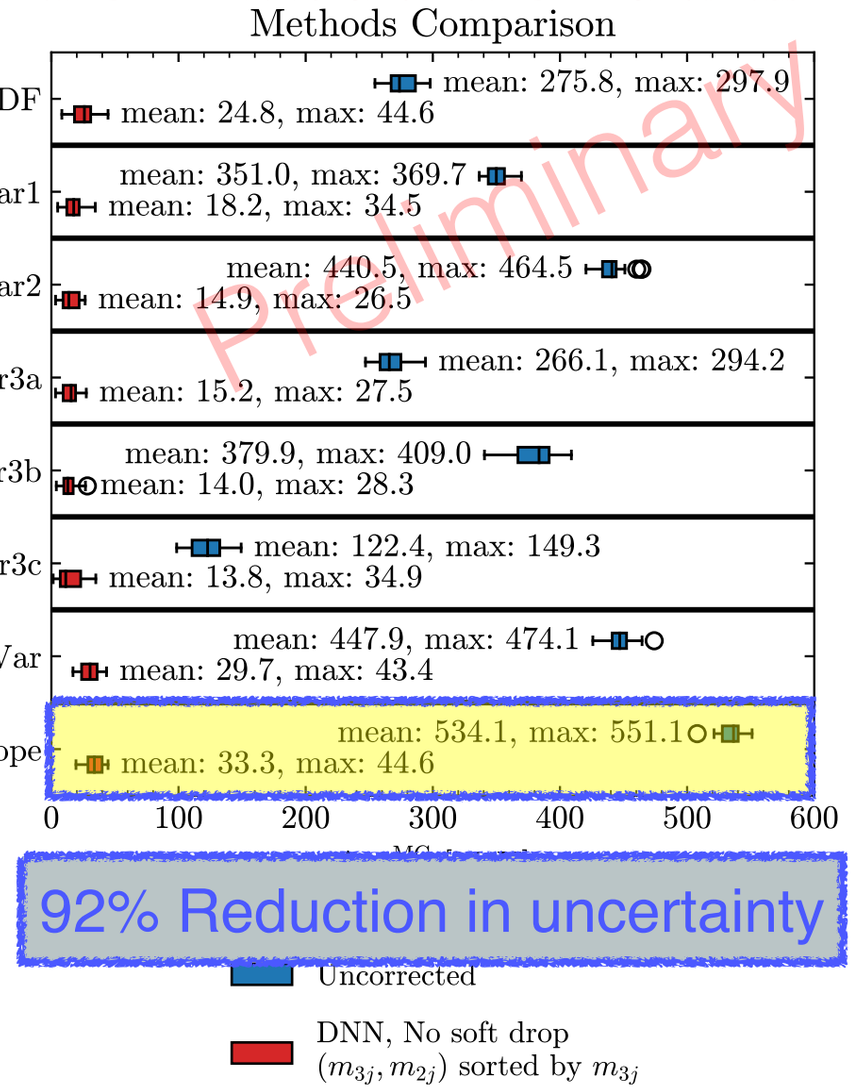
Test on $m_t = (172, 172.5, 173, 173.5, 174)$ GeV

Test set size: 400k (5 separate test sets per mass)

Randomly select 30k -> get mass

Repeat many times: mean is prediction

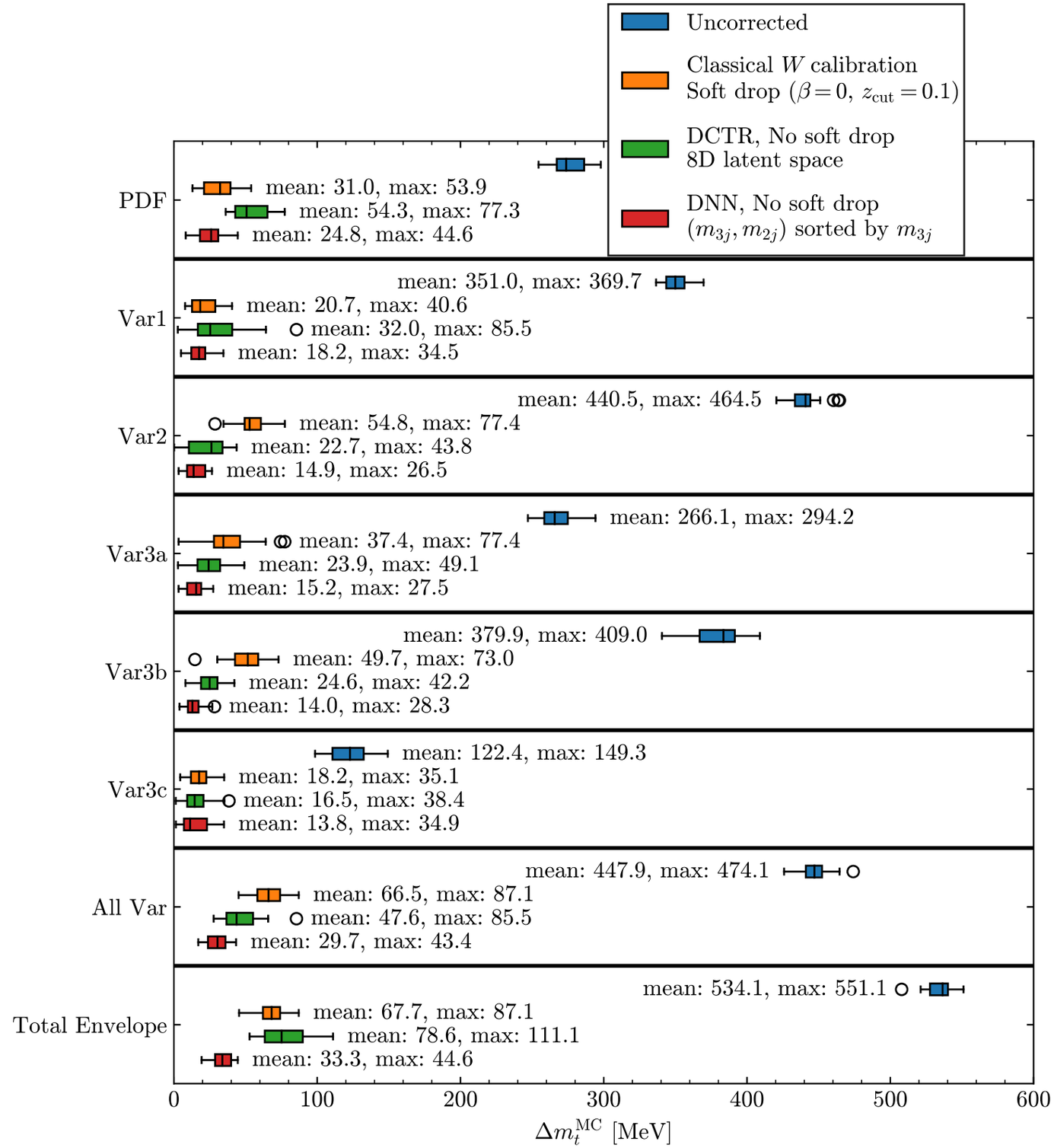




- Network is able to interpolate between masses
- Network learns distributions for the PDFS and tunes
- Grooming does not help
- Other observables did not help (2-jet mass useful for network to calibrate)

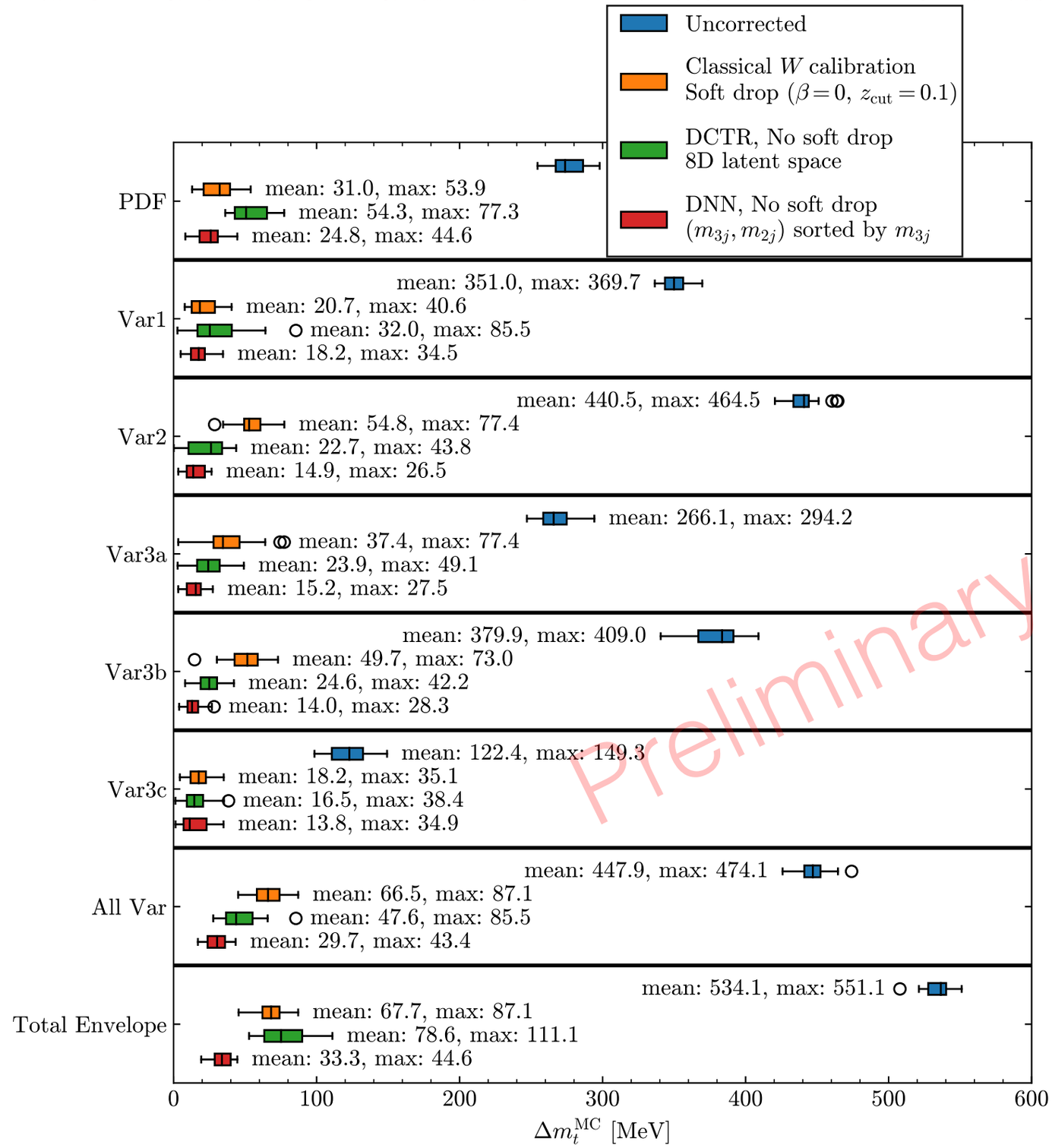
Conclusion

- Precise determination of top quark properties constrains BSM
- Machine learning can reduce uncertainty 28-55% more than soft drop and W calibration
- Methods can be used for other measurements (cross section, width, etc)



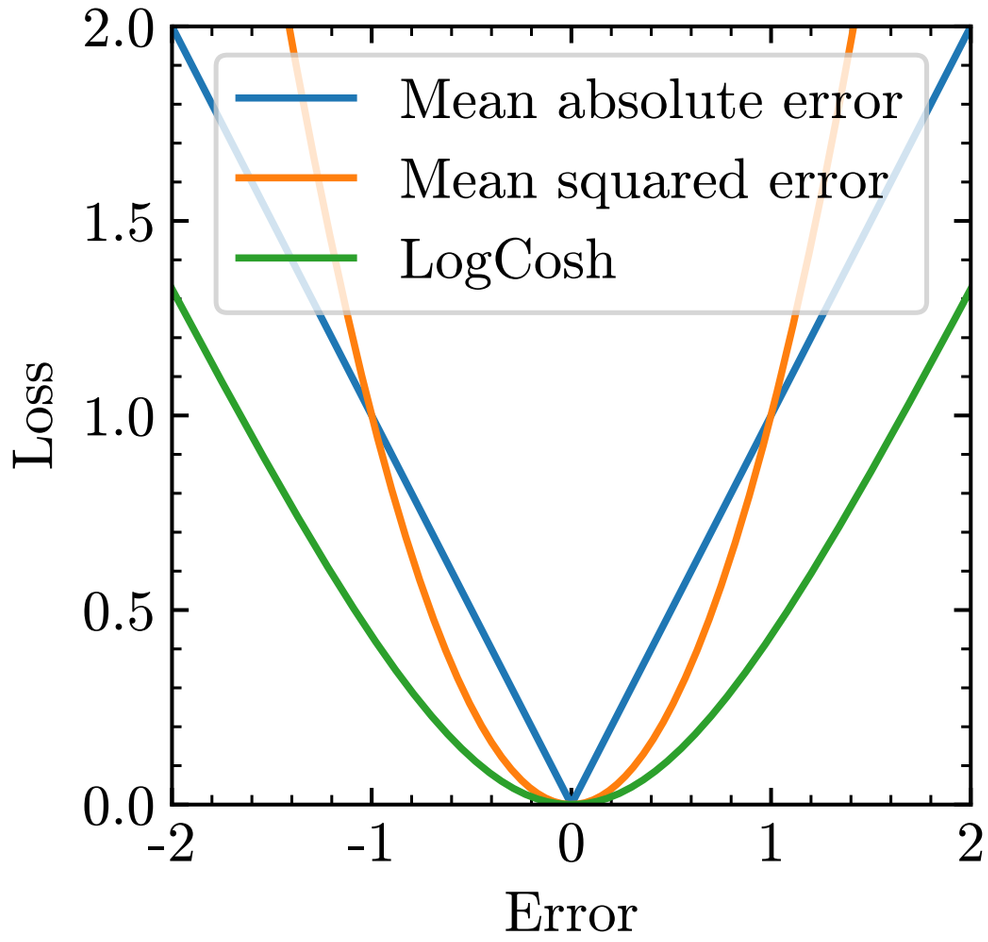
Conclusion

- Precise determination of top quark properties constrains BSM
- Machine learning can reduce uncertainty 28-55% more than soft drop and W calibration
- Methods can be used for other measurements (cross section, width, etc)



Backup

Loss options



The LogCosh loss has a shallower slope for small errors, allowing for easier minimization

Return

Tuning with DCTR

Slide stolen from Anders Andreassen BOOST 2019 talk

- A well trained model $f(x, \theta)$minimizes the loss for any θ

$$f(x,\theta) = \underset{f'}{\text{argmax}} \sum_{i \in \theta_{\mathbf{0}}} \log f'(x_i, \theta) + \sum_{i \in \theta} \log\left(1 - f'(x_i, \theta)\right)$$

Tuning with DCTR

Slide stolen from Anders Andreassen BOOST 2019 talk

- A well trained model $f(x, \theta)$minimizes the loss for any θ

$$f(x,\theta) = \underset{f'}{\text{argmax}} \sum_{i \in \theta_{\mathbf{0}}} \log f'(x_i, \theta) + \sum_{i \in \theta} \log\left(1 - f'(x_i, \theta)\right)$$

- Given a sample with θ_1 unknown (e.g. data), then

$$\theta^* = \underset{\theta'}{\text{argmax}} \sum_{i \in \theta_{\mathbf{0}}} \log f(x_i, \theta') + \sum_{i \in \theta_{\mathbf{1}}} \log\left(1 - f(x_i, \theta')\right)$$

imples that $\theta^* = \theta_1$

Tuning with DCTR

Slide stolen from Anders Andreassen BOOST 2019 talk

- A well trained model $f(x, \theta)$minimizes the loss for any θ

$$f(x,\theta) = \underset{f'}{\text{argmax}} \sum_{i \in \theta_\mathbf{0}} \log f'(x_i, \theta) + \sum_{i \in \theta} \log\left(1 - f'(x_i, \theta)\right)$$

- Given a sample with θ_1 unknown (e.g. data), then

$$\theta^* = \underset{\theta'}{\text{argmax}} \sum_{i \in \theta_\mathbf{0}} \log f(x_i, \theta') + \sum_{i \in \theta_\mathbf{1}} \log\left(1 - f(x_i, \theta')\right)$$

  imples that $\theta^* = \theta_1$

- Can find solution by gradient descending on θ'