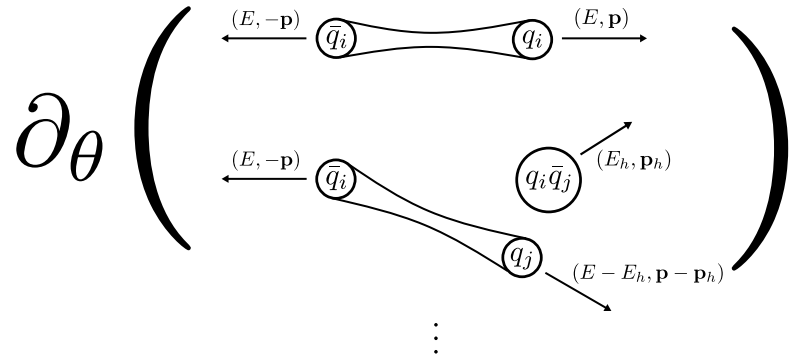


The journey towards differentiable hadronization models

Michigan State University
HEP Seminar
September 30th, 2025

Tony Menzo

Postdoc @ Fermilab and the University of Alabama



Based upon works in [2203.04983](#), [2308.13459](#), [2311.09296](#), [2410.06342](#), [2411.02194](#),
[2505.00142](#), [2503.05667](#), [2509.03592](#)

MLHAD



Jure Zupan



Phil Ilten



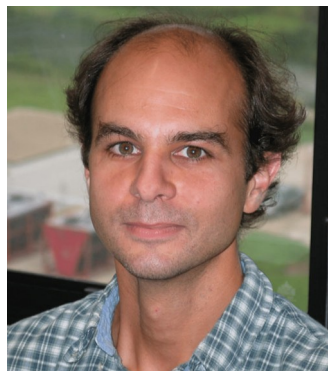
Manuel Szwec



Rikab Gambhir



Christian Bierlich



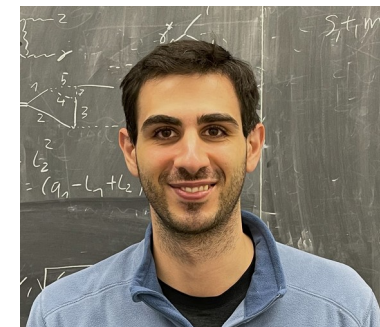
Steve Mrenna



Ahmed Youssef

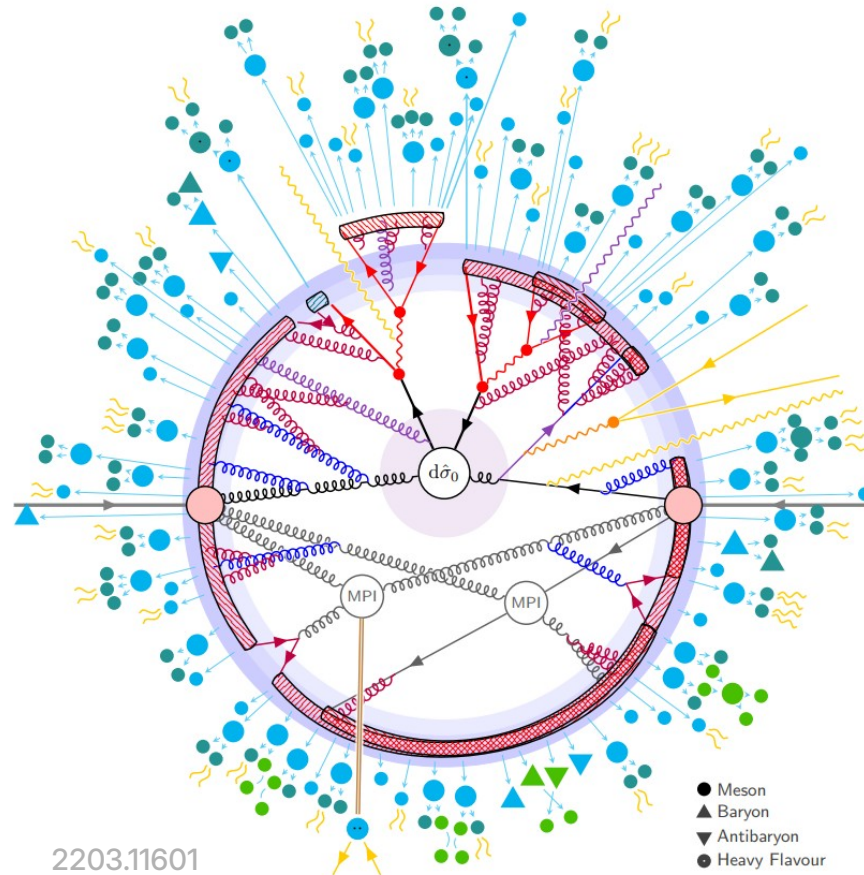


Michael Wilkinson

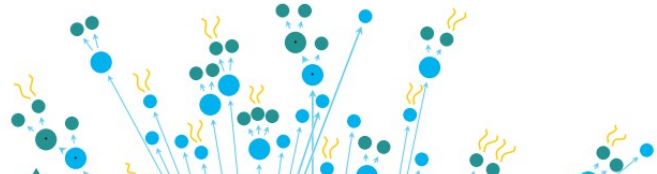


Ben Assi

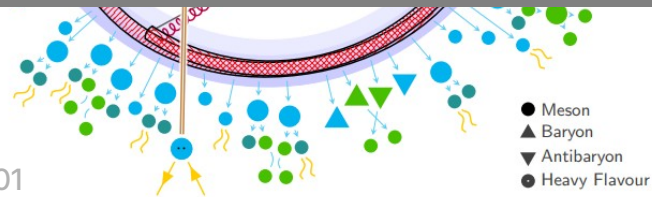
Monte Carlo Event generators



Monte Carlo Event generators



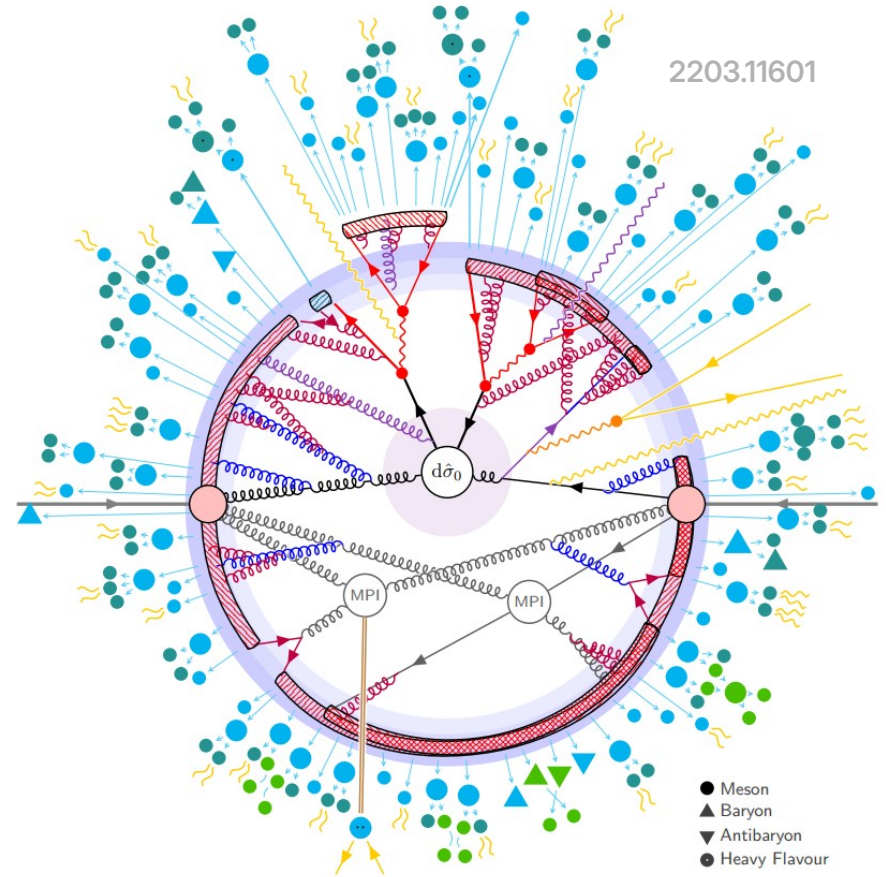
$$\mathcal{G} : \underbrace{\mathcal{S}(\mathcal{D}(\mathcal{H}(\mathcal{P}(\mathcal{M}))))}_{\text{Simulation}} = \mathcal{E} \simeq \underbrace{\left(\begin{array}{c} \{\text{id}, E, p_x, p_y, p_z, \dots\}_1 \\ \{\text{id}, E, p_x, p_y, p_z, \dots\}_2 \\ \vdots \\ \{\text{id}, E, p_x, p_y, p_z, \dots\}_N \end{array} \right)}_{\text{Event record}} .$$



2203.11601

Hadronization

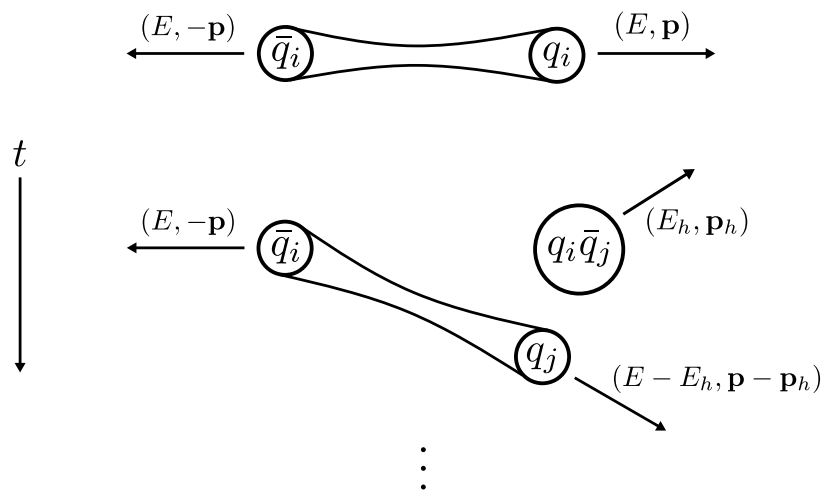
The conversion of a partonic system consisting of quarks, anti-quarks, and gluons ($q\bar{q}$, $qg\bar{q}$, gg , qqq , $qq\bar{q}\bar{q}$, $qqqgg$, ...) to a final state consisting of hadrons (h_1, h_2, \dots, h_n).



Phenomenological models

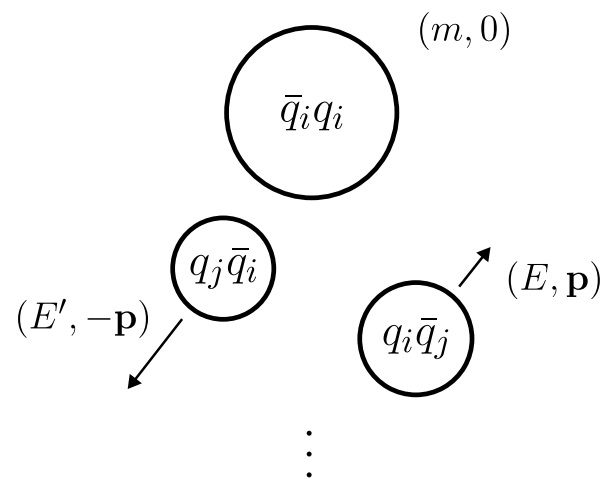
Lund string model

(used in Pythia)



Cluster model

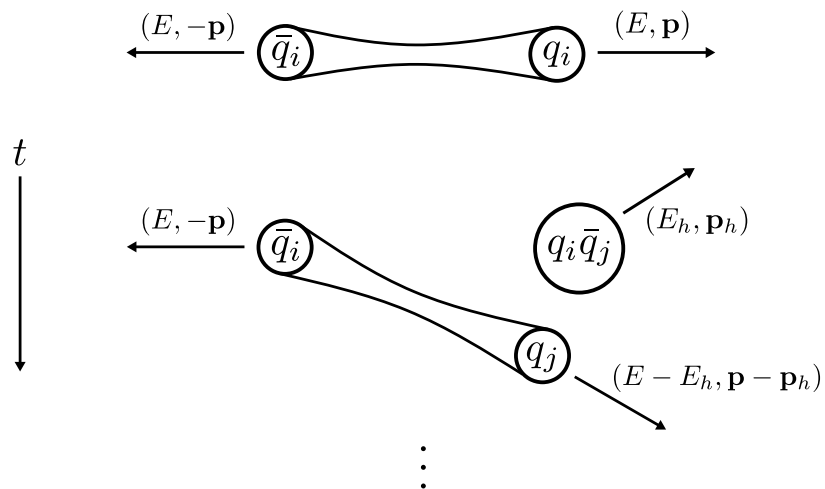
(used in Herwig, Sherpa)



Phenomenological models

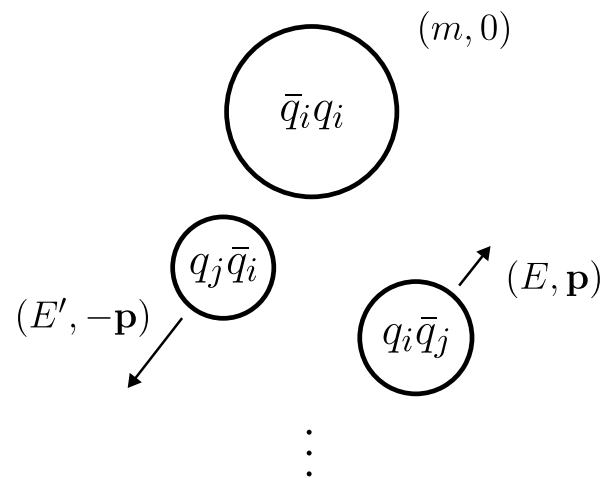
Lund string model

(used in Pythia)



Cluster model

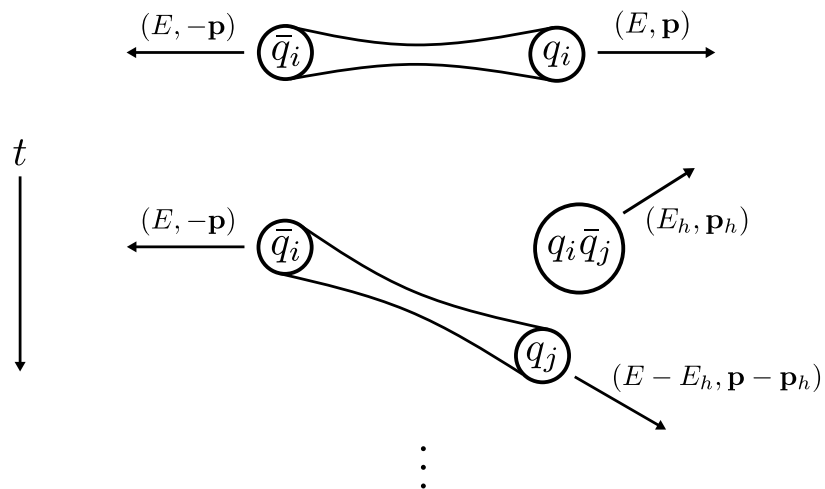
(used in Herwig, Sherpa)



Phenomenological models

Lund string model

(used in Pythia)



Cluster model

(used in Herwig, Sherpa)

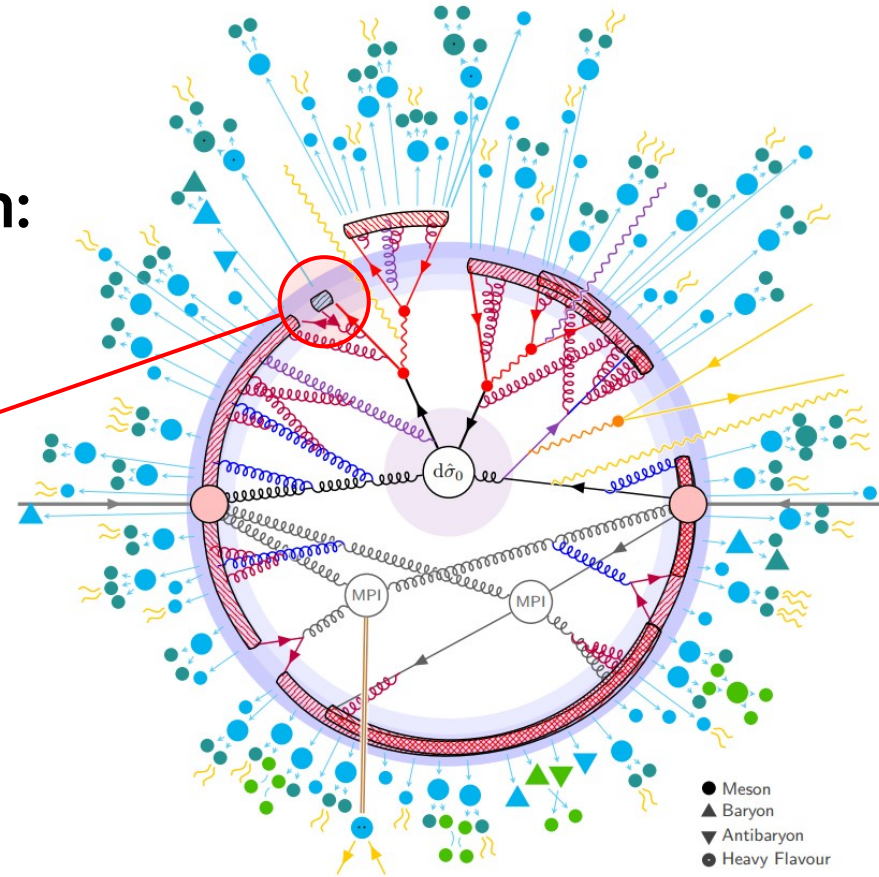
[HadML collaboration](#)

For ML methods applied to the cluster model:

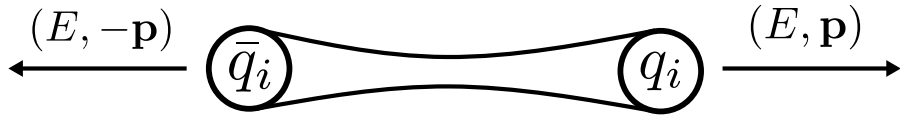
2203.12660, 2305.17169,
2312.08453

⋮

The "atomic" hadronizing system: Quark-antiquark ($q\bar{q}$)



The algorithm ($q\bar{q}$)



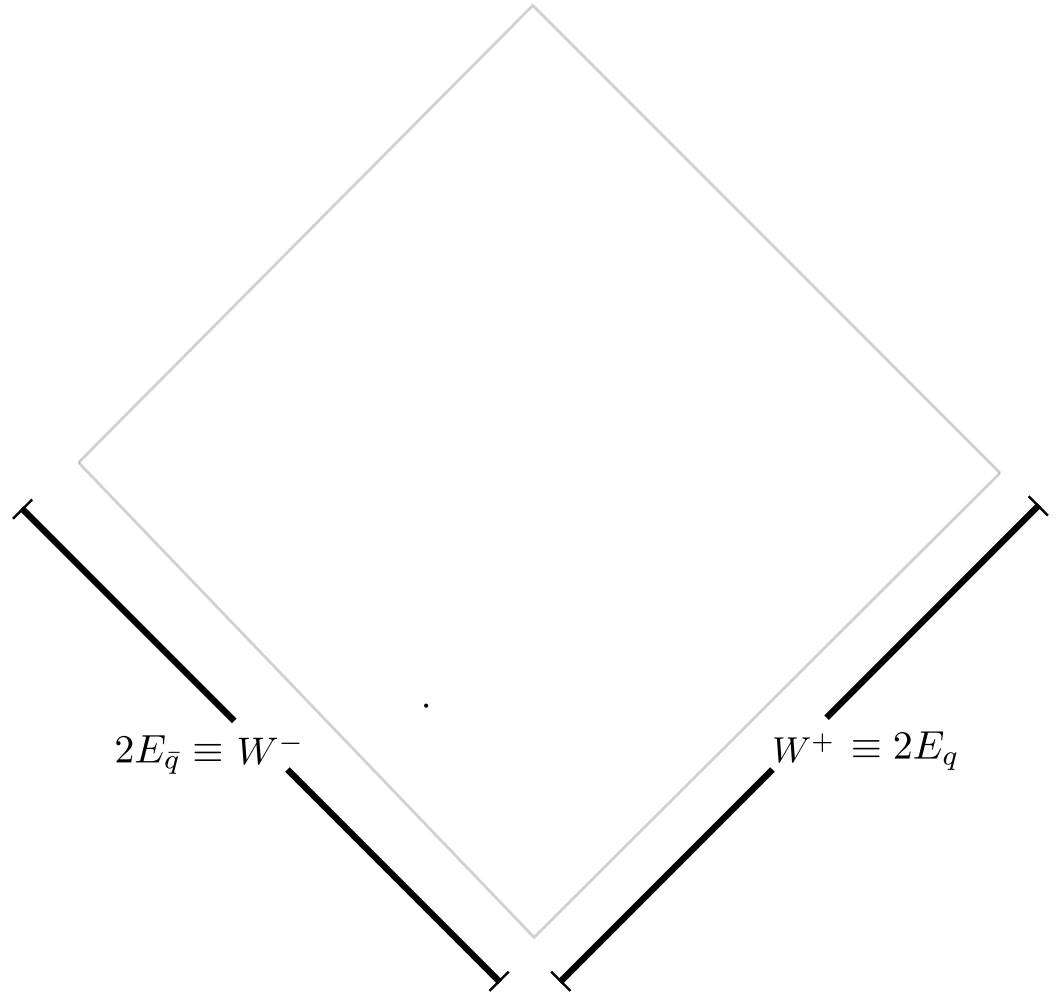
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi \sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

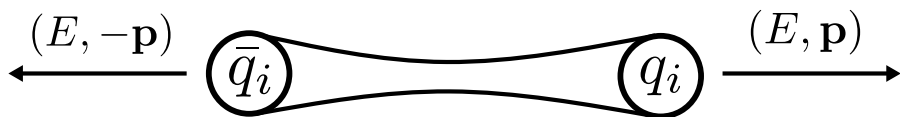
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



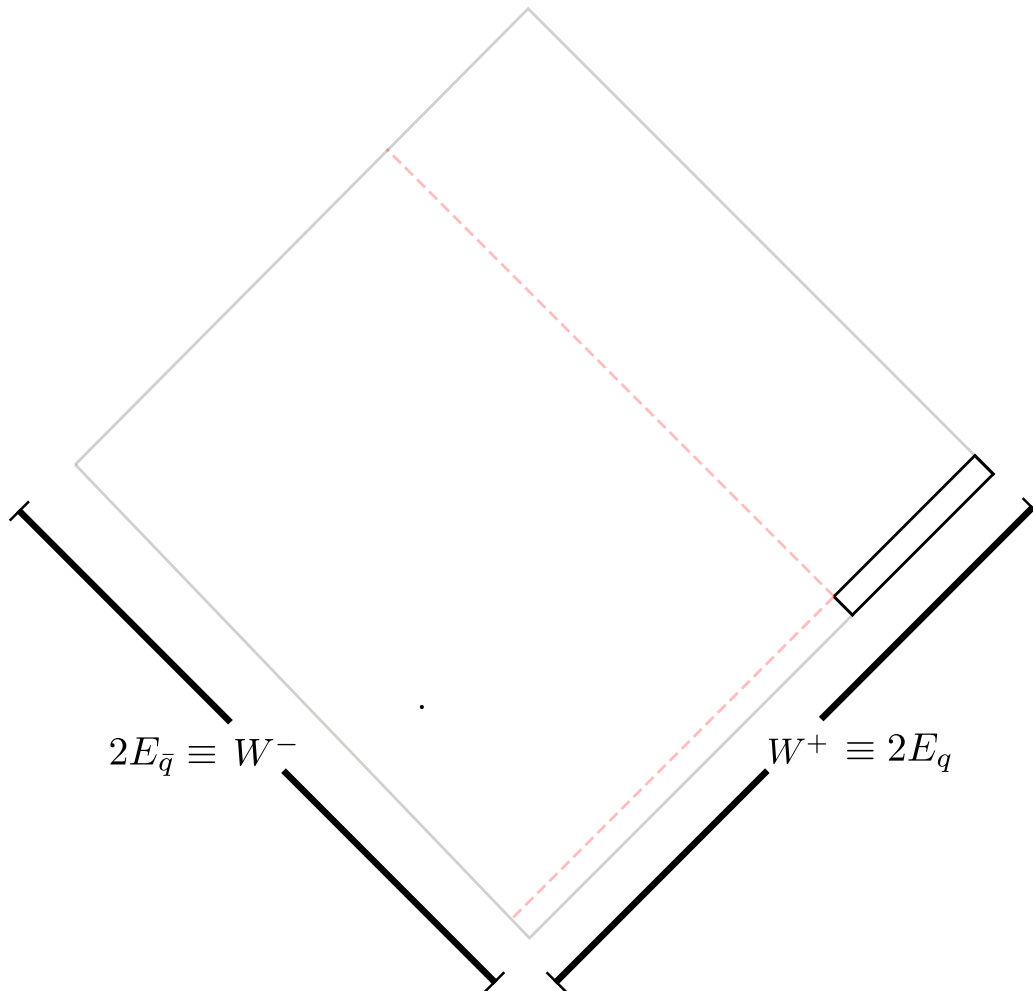
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi\sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

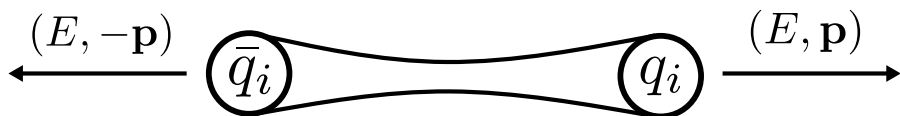
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



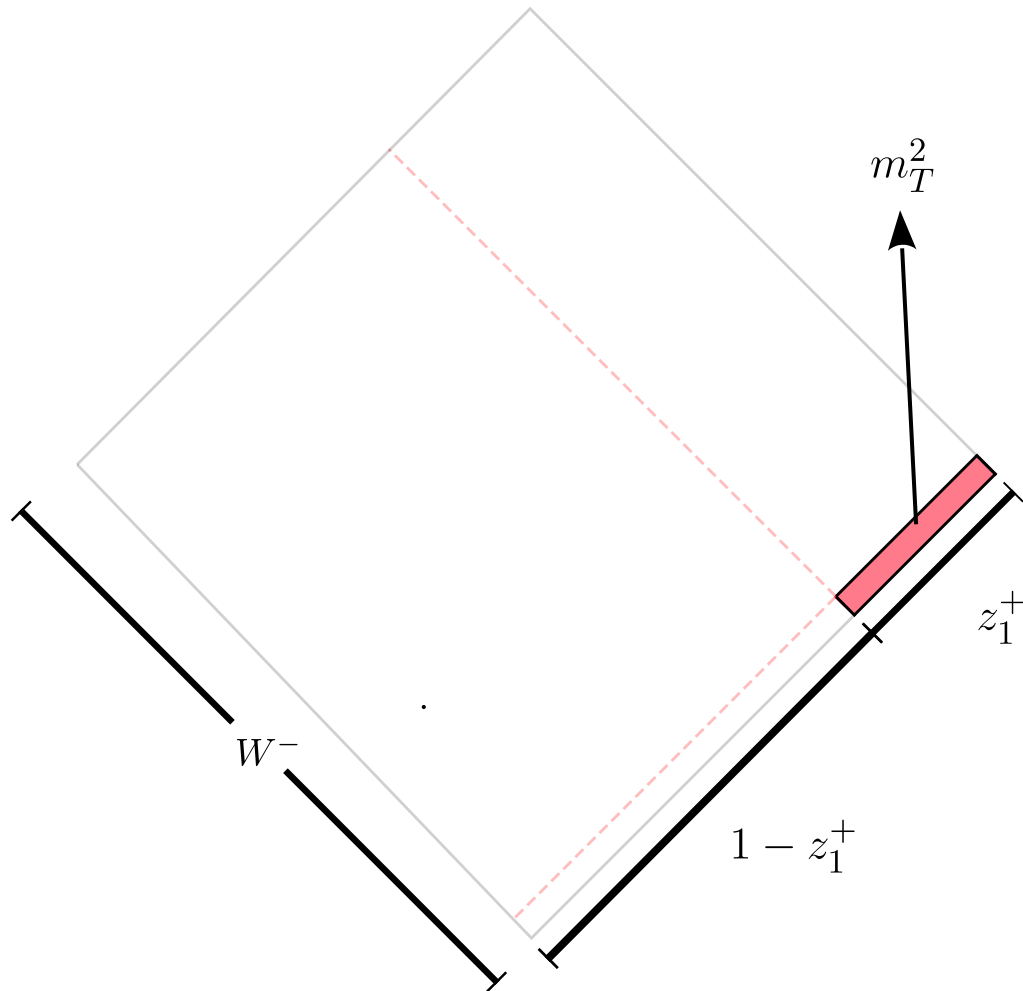
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi\sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

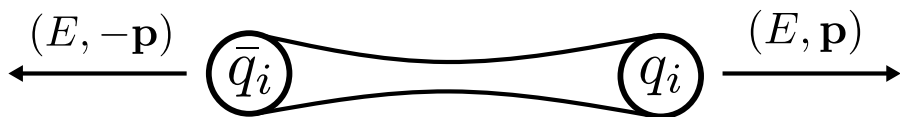
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(\frac{-bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



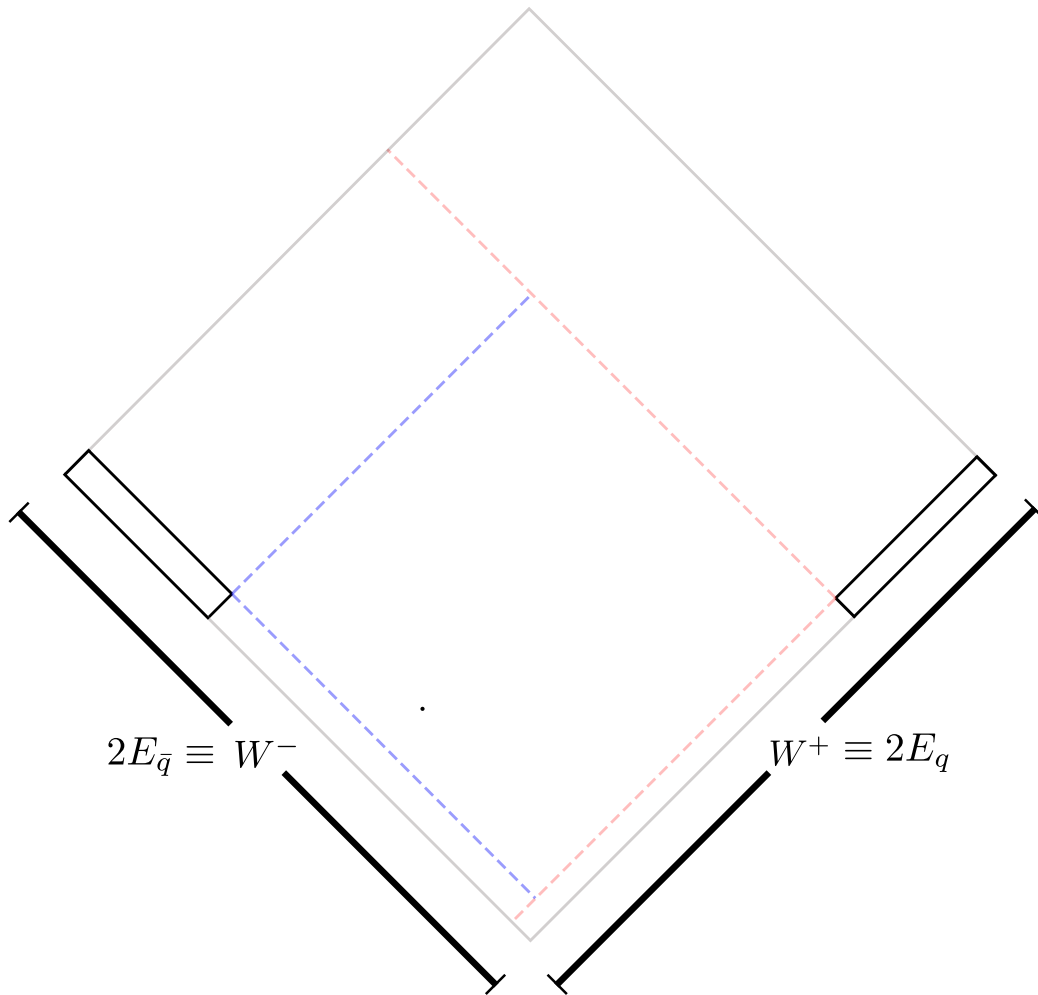
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi\sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

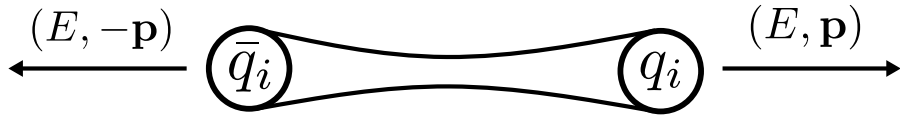
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



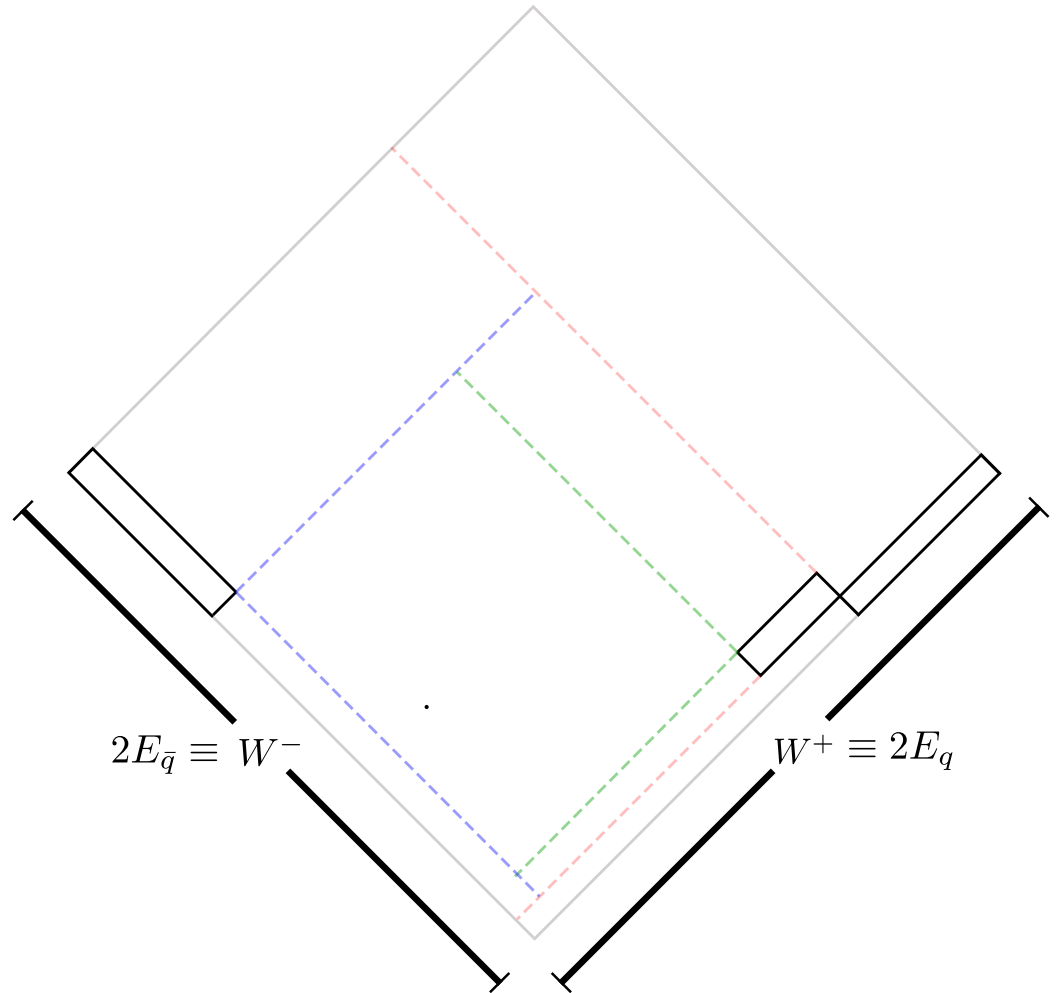
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi\sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

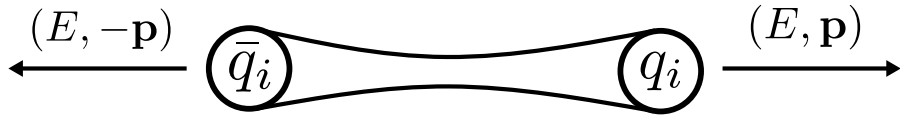
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



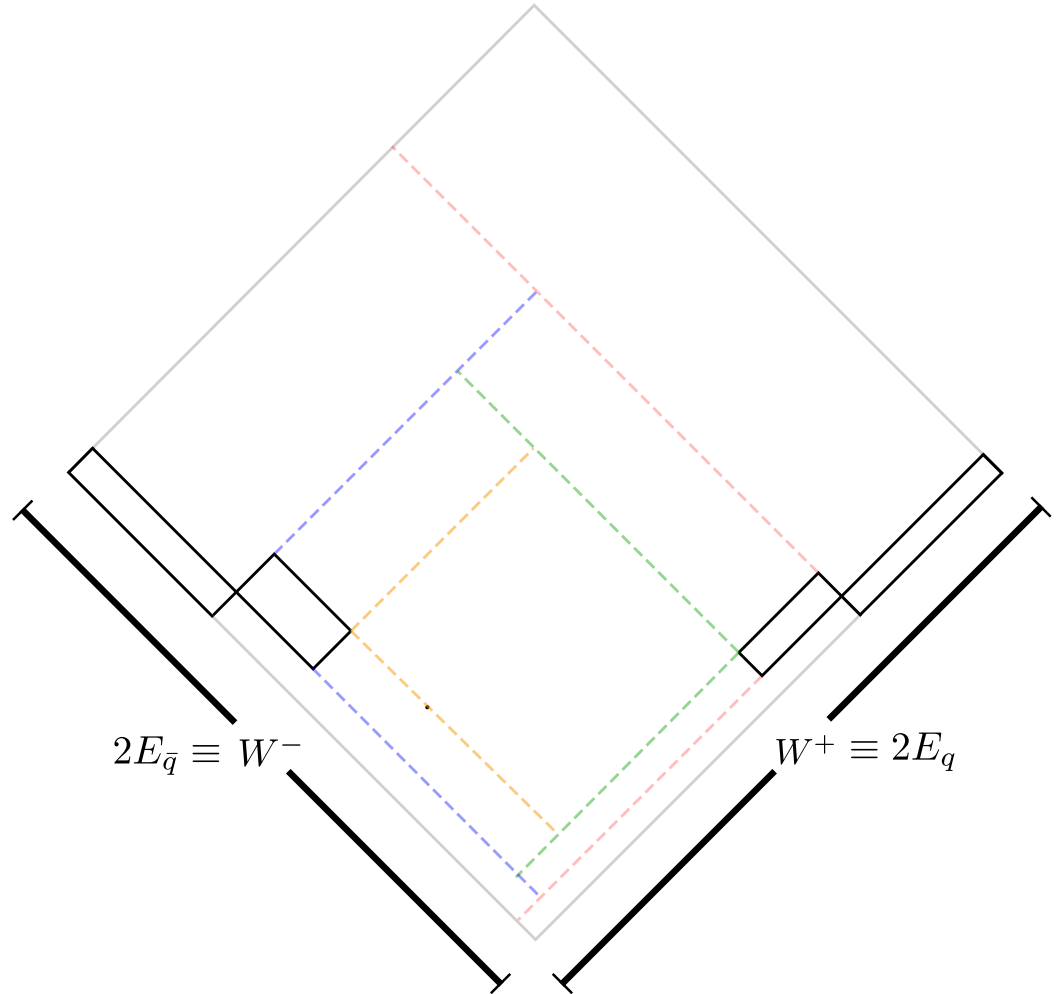
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi\sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

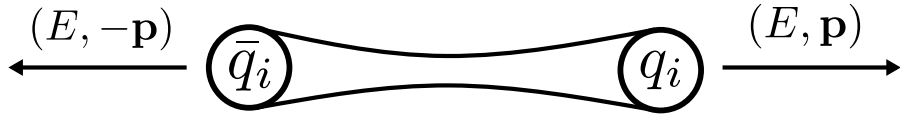
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



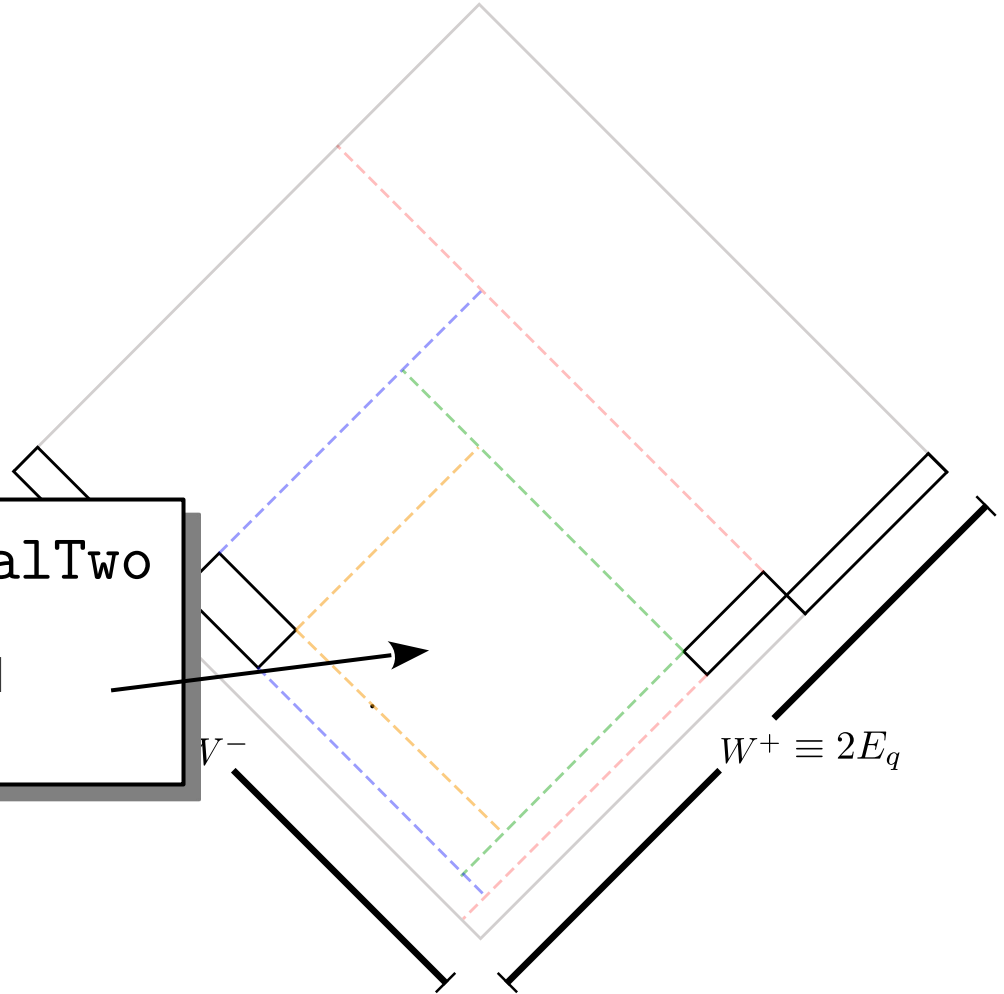
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$\mathcal{P}(z)$
4) Sample new hadron

When E_{CM} goes below $\sim 2 \text{ GeV}$, finalTwo is called. This can fail, when it does, the full string system is re-simulated from the beginning.

$$f(z) \propto \frac{1}{z} \exp\left(-\frac{1}{z}\right), \quad z = \frac{E_q}{2E}$$

- 5) Repeat steps 1-4



The algorithm ($q\bar{q}$)



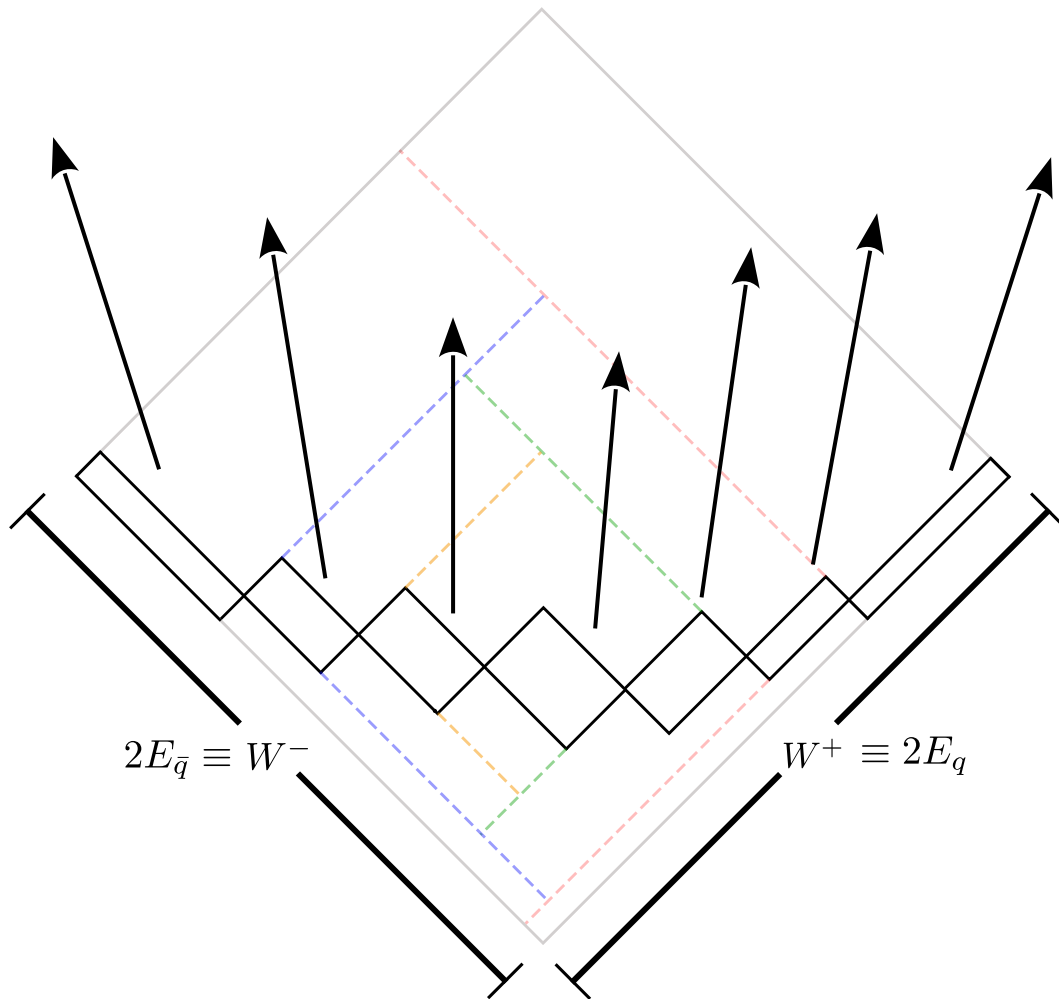
- 1) Randomly select one of the string ends
- 2) Sample string break flavor
- 3) Sample transverse momentum of new quarks

$$\mathcal{P}(p_x, p_y; \sigma_{p_T}) = \frac{1}{\pi \sigma_{p_T}^2} \exp\left(-\frac{p_x^2 + p_y^2}{\sigma_{p_T}^2}\right)$$

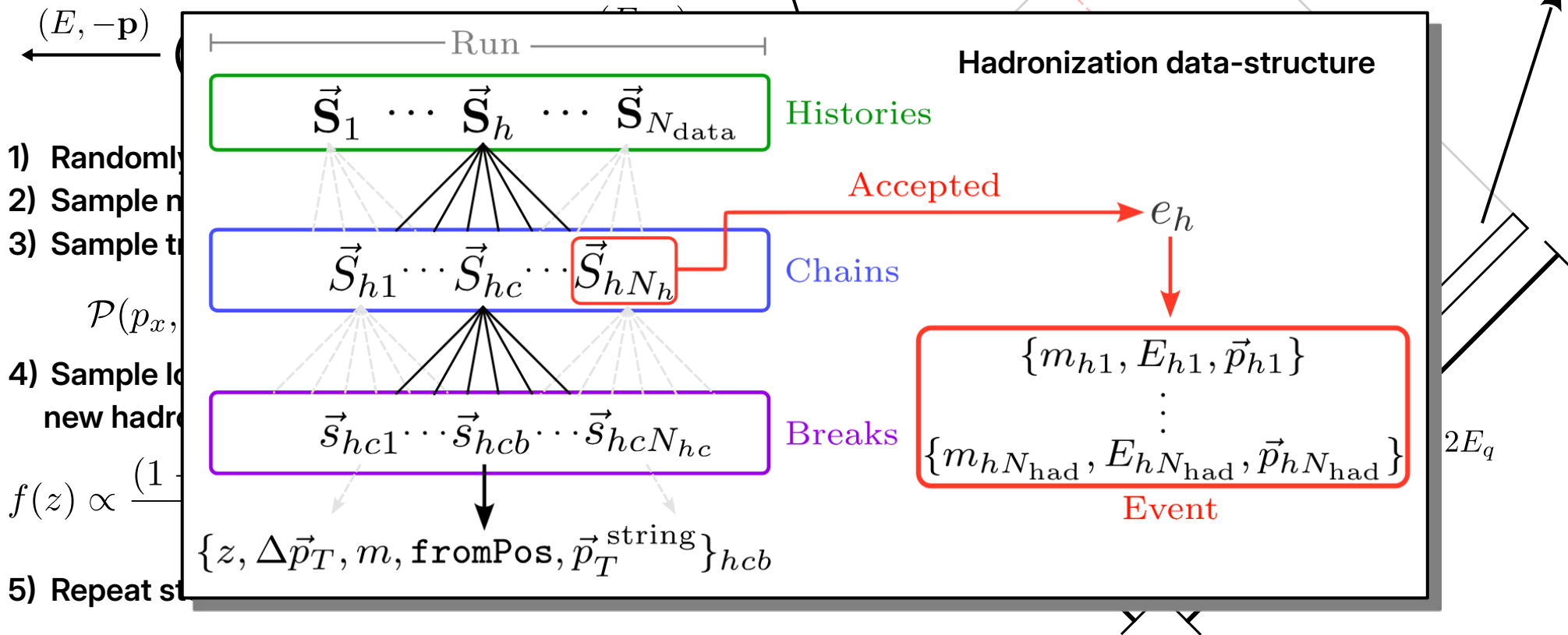
- 4) Sample longitudinal momentum fraction of new hadron

$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(-\frac{bm_T^2}{z}\right), \quad z = \frac{p_z + E_h}{2E}$$

- 5) Repeat steps 1-4



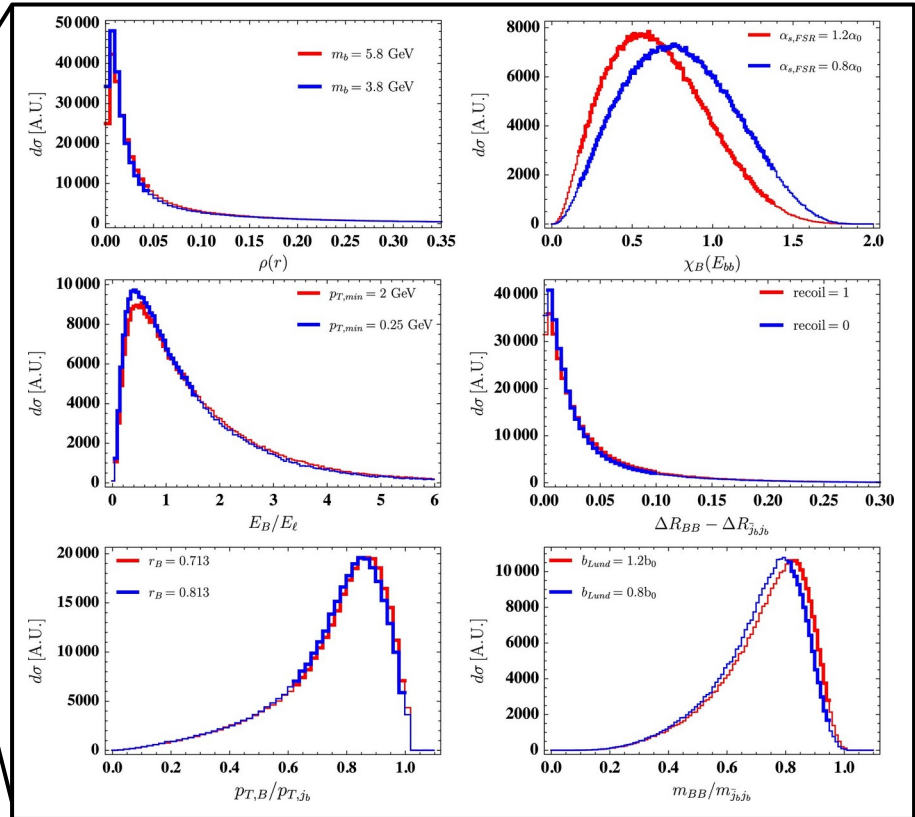
The algorithm (q \bar{q})



Why?

Precision (exclusive) measurements are sensitive to hadronization!

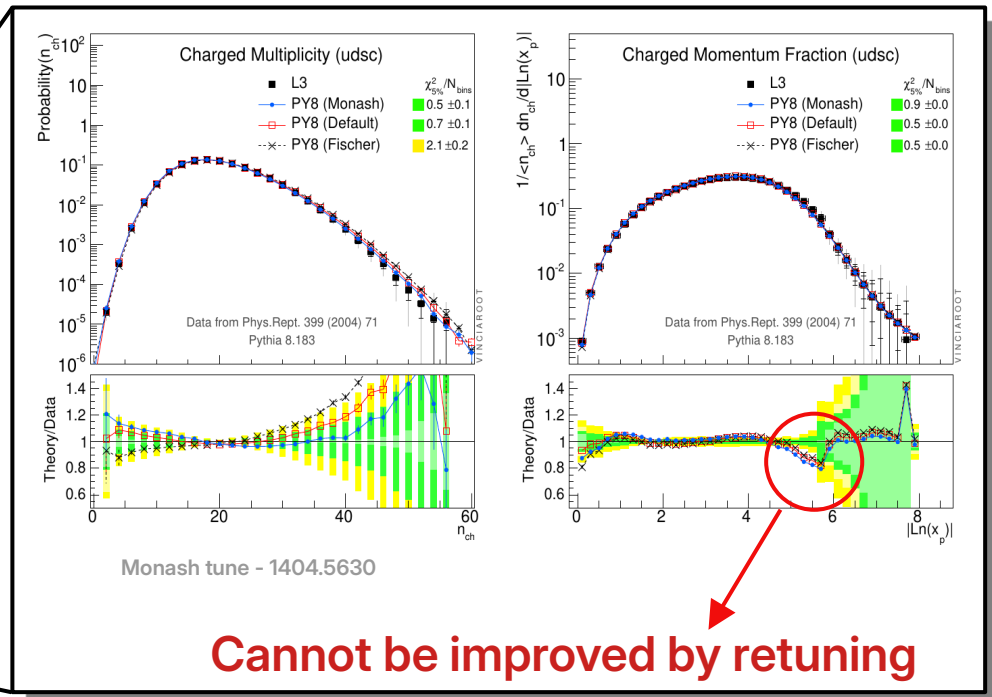
- Uncertainty reduction
 - Top quark mass measurement (r_b)
 - e^+e^- determination of α_s
- Mis-modeling bias
 - High-multiplicity events
 - Tuning discrepancies



Why?

Precision (exclusive) measurements are sensitive to hadronization!

- Uncertainty reduction
 - Top quark mass measurement (r_b)
 - e^+e^- determination of α_s
- Mis-modeling bias
 - High-multiplicity events
 - Tuning discrepancies



Three roads to improvement

- Improve model

- MPIs, rope hadronization, transverse mass suppression, flavor asymmetries, hadronic rescattering, multiscale models (string → hydrodynamical), flavor selector, etc.

Hard to come up with mathematically precise model without established calculational techniques

- Tend towards model independence

- Sample directly from global distributions

Non-universal and extremely difficult to convert into **representative** particle flow data (uninterpretable)

- Improve observables

- Come up with better, hadronization sensitive (IR unsafe), observables

New observables must be measured by experiments = **time + \$\$\$**

Three* roads to improvement

- Improve model

- MPIs, rope hadronization, transverse mass suppression, flavor asymmetries, hadronic rescattering, multiscale models (string → hydrodynamical), flavor selector, etc.

Hard to come up with mathematically precise model without established calculational techniques

- Tend towards model independence

- Sample directly from global distributions

Non-universal and extremely difficult to convert into **representative** particle flow data (uninterpretable)

- Improve observables

- Come up with better, hadronization sensitive (IR unsafe), observables

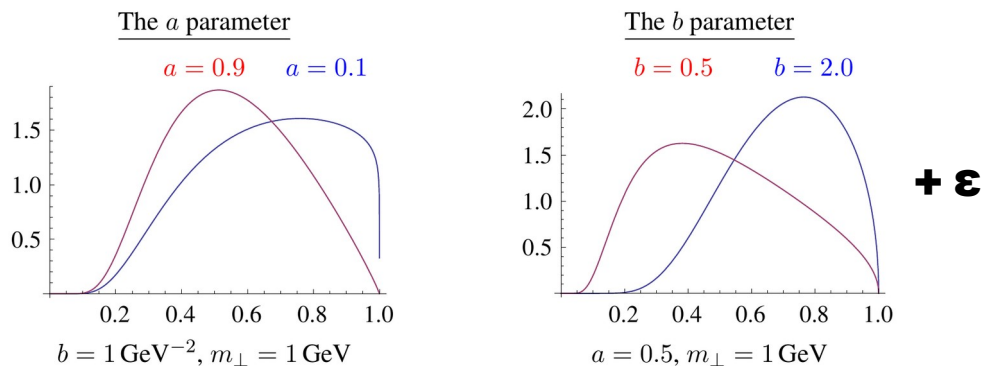
New observables must be measured by experiments = **time + \$\$\$**

***or some combination of all three**

Data-driven fine tuning

The phenomenological models of hadronization already give an acceptable description of a large amount of data.

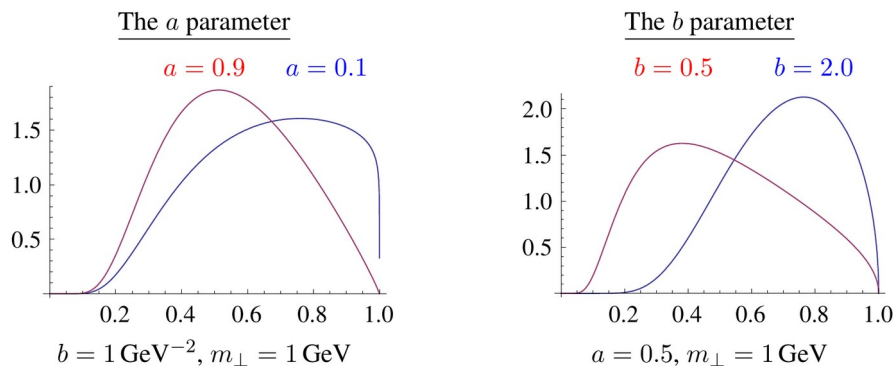
Hybrid approach: Keep the underlying paradigm e.g. strings but modify the microscopic kinematics to accommodate the discrepant global experimental observables.



Data-driven fine tuning

The phenomenological models of hadronization already give an acceptable description of a large amount of data.

Hybrid approach: Keep the underlying paradigm e.g. strings but modify the microscopic kinematics to accommodate the discrepant global experimental observables.

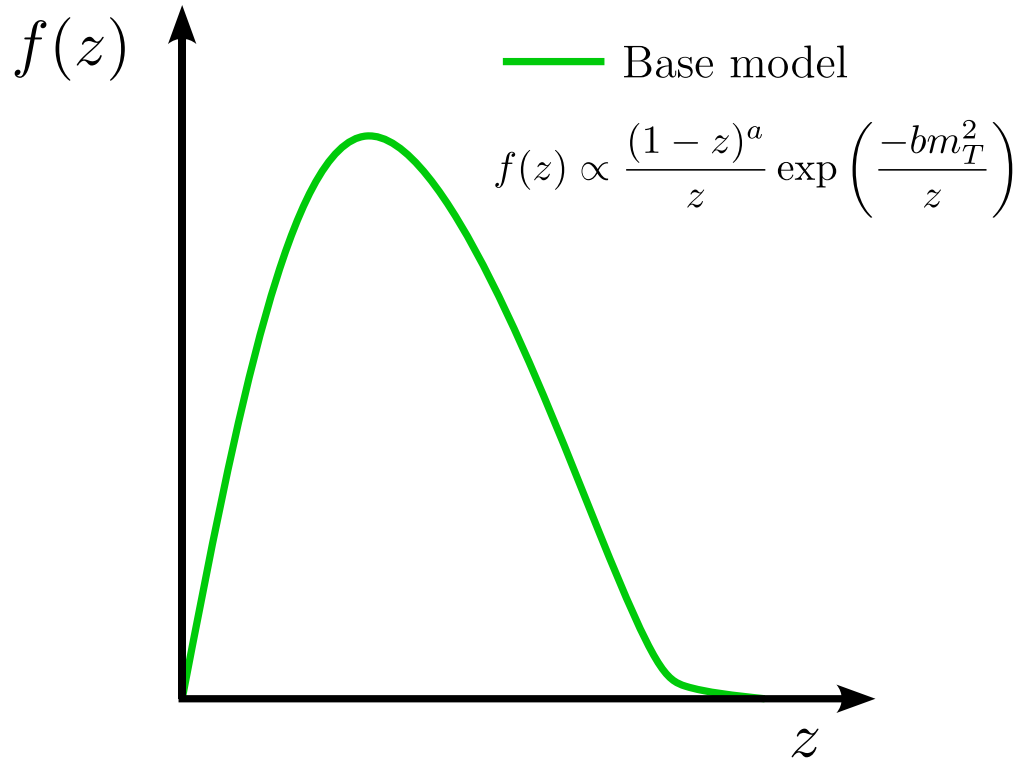


+ ϵ

Machine learning offers a nice framework to tackle this problem

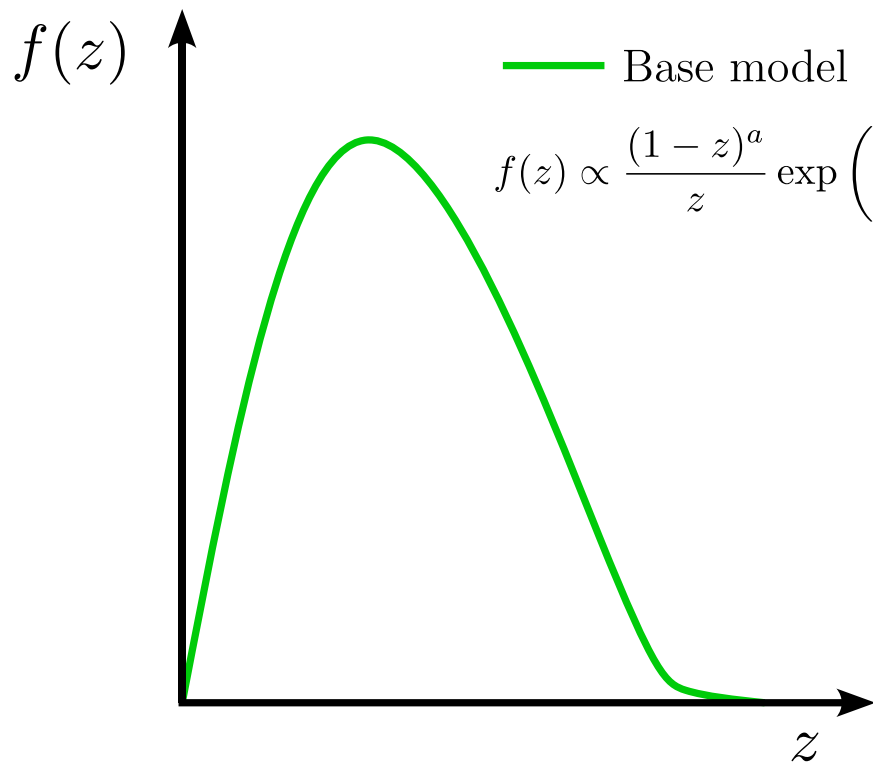
MLHAD **efforts: big picture**

Solving the “inverse problem of hadronization”



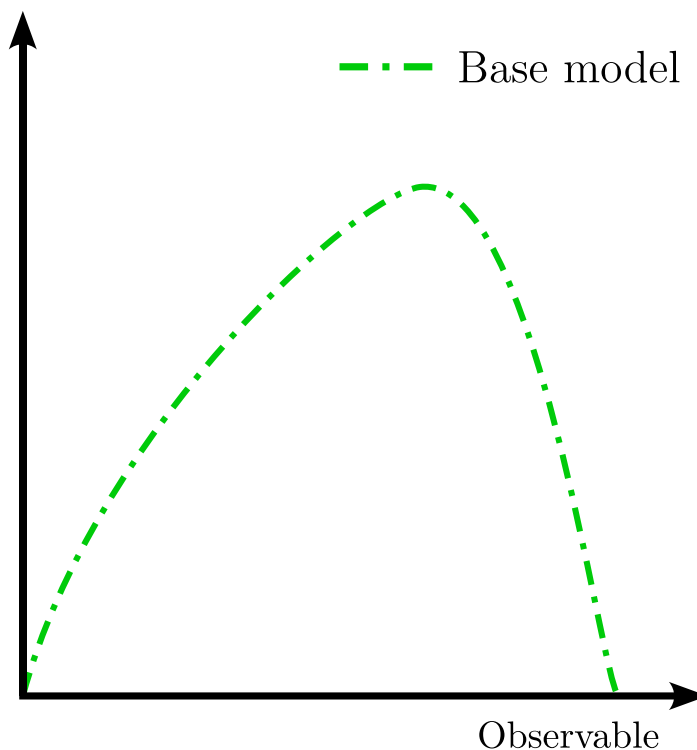
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



— Base model

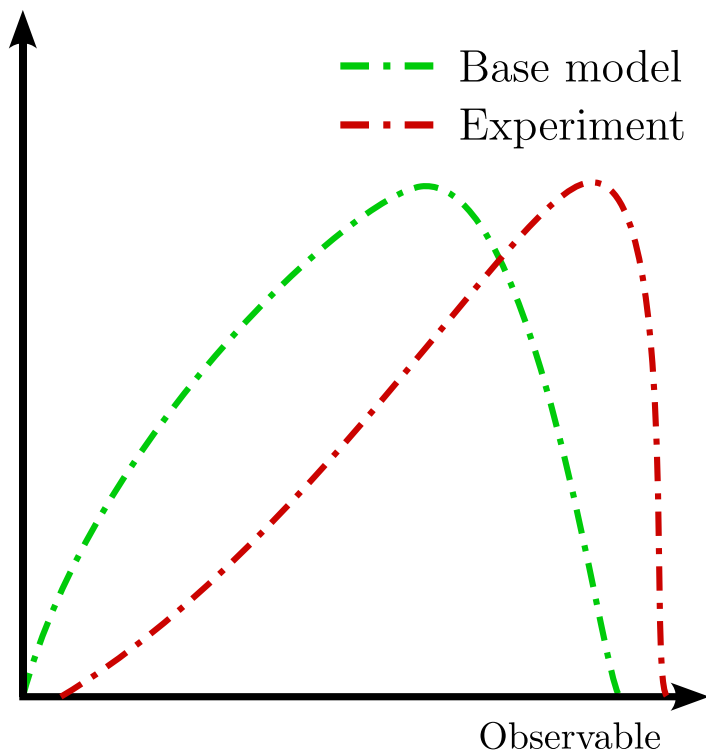
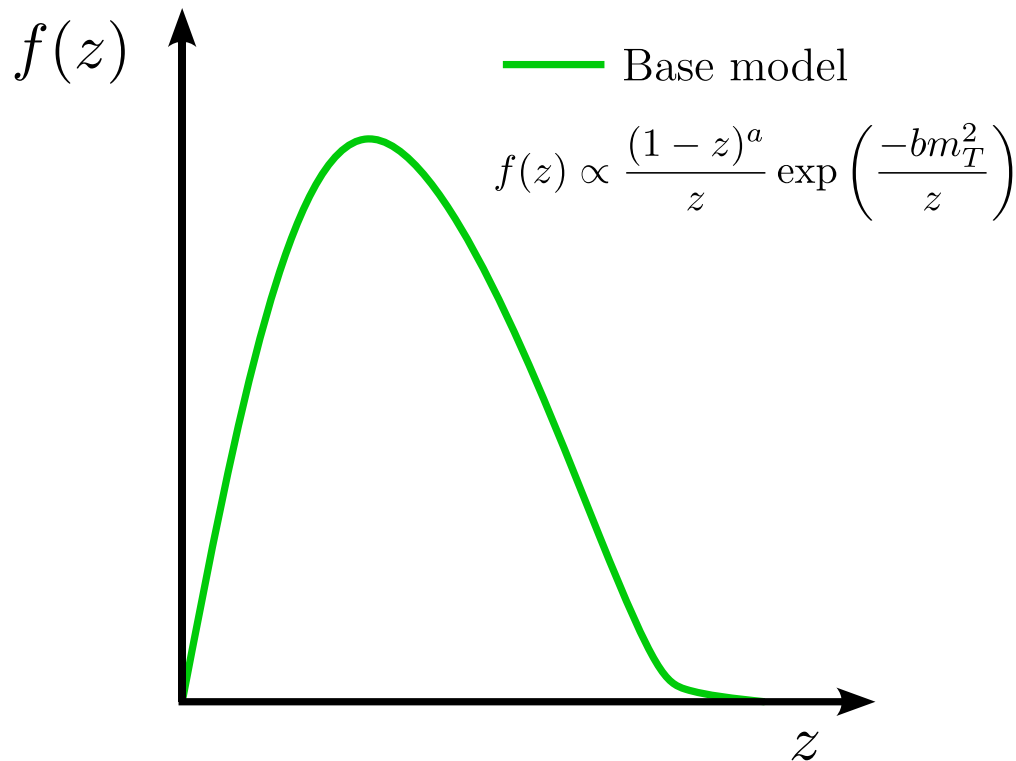
$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(\frac{-bm_T^2}{z}\right)$$



- - - Base model

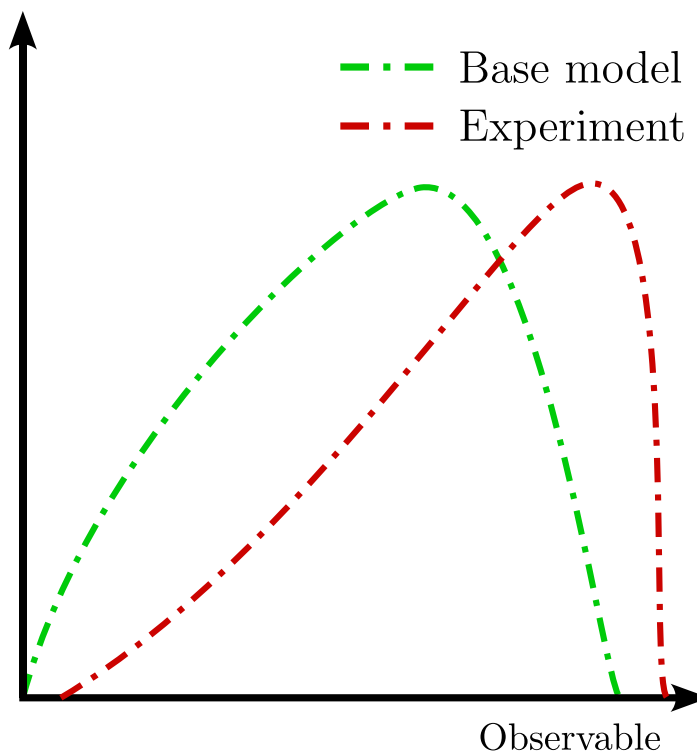
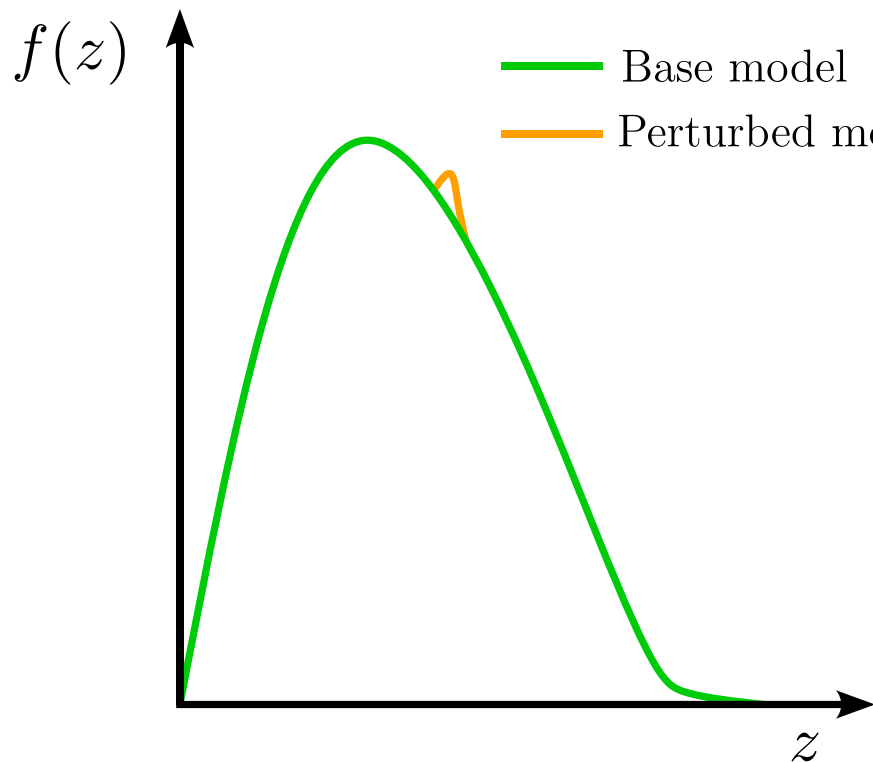
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



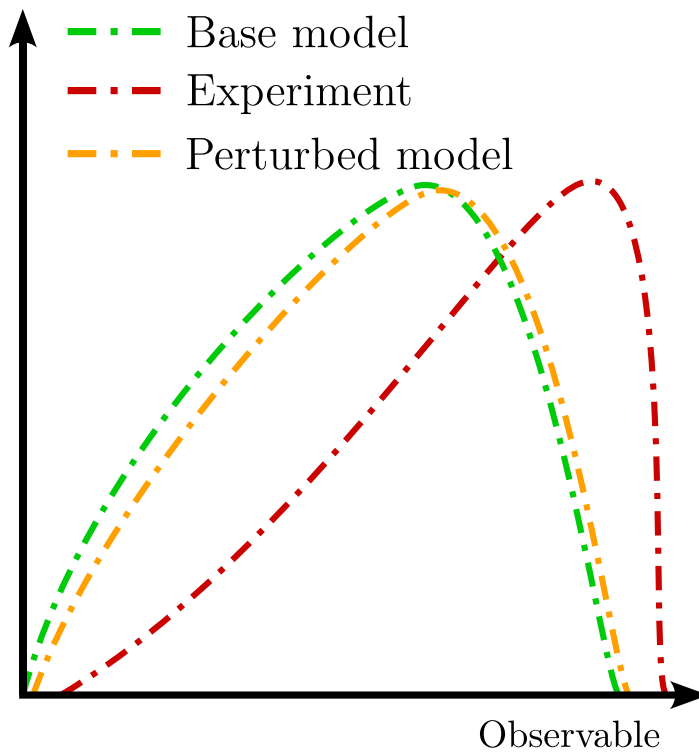
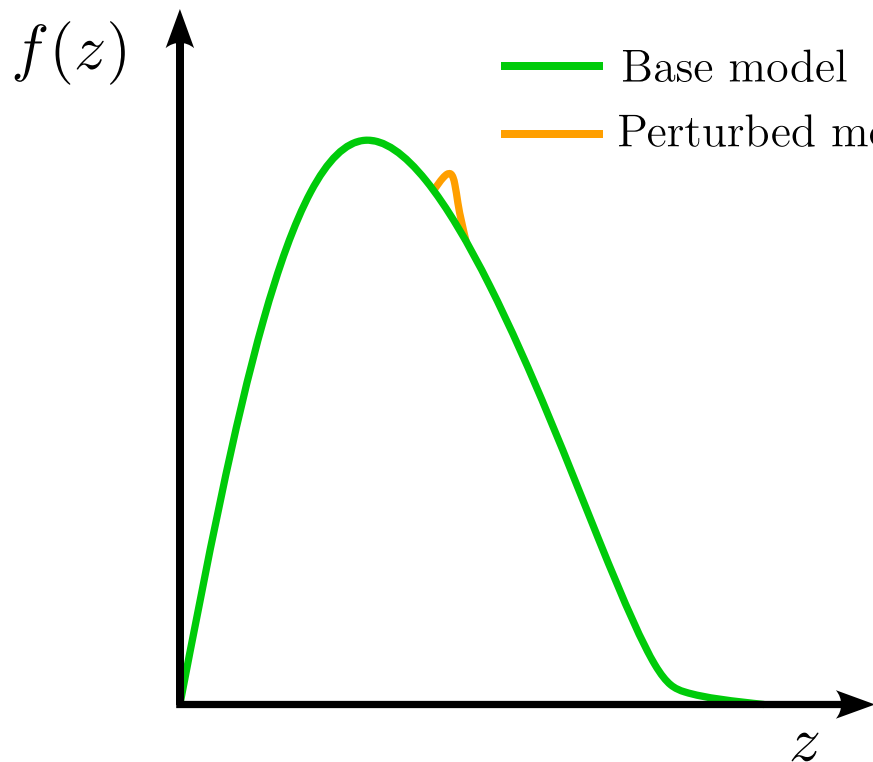
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



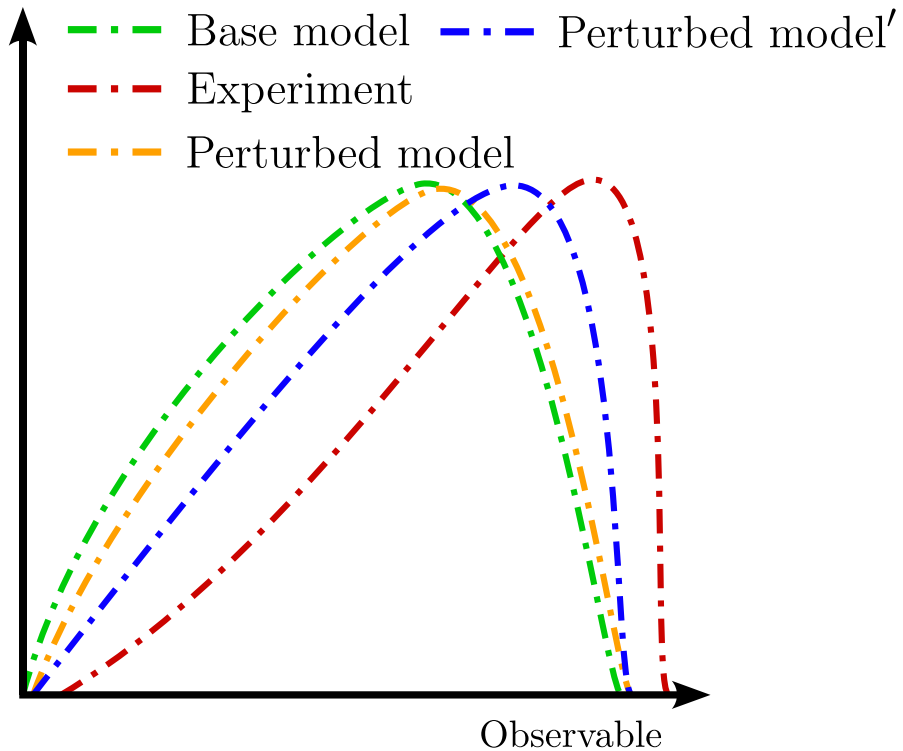
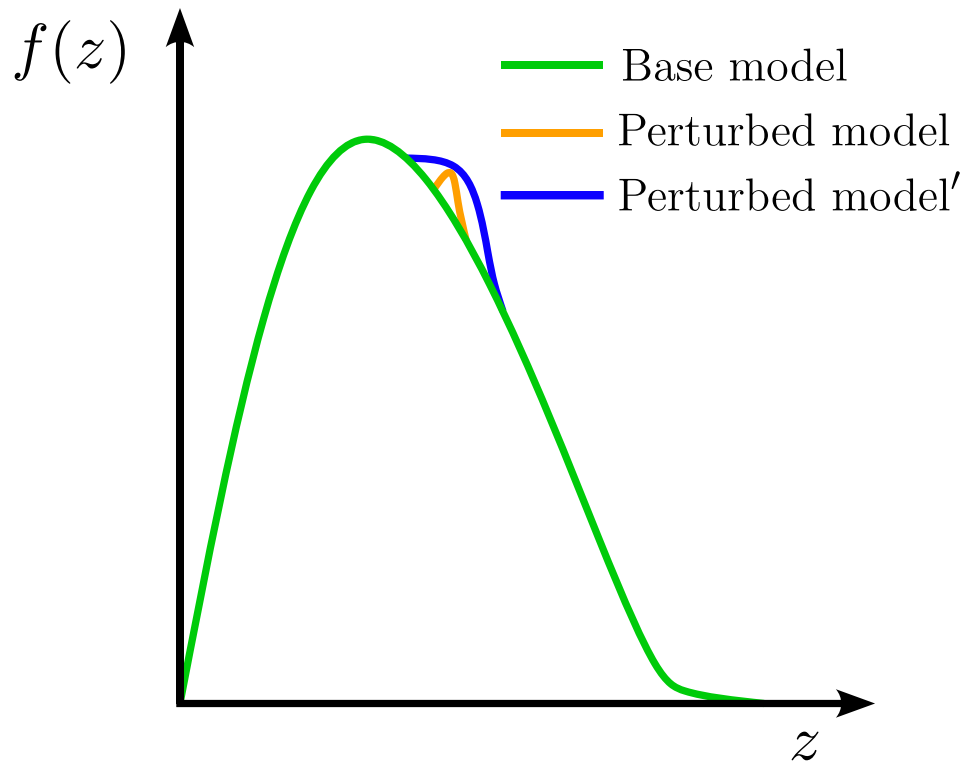
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



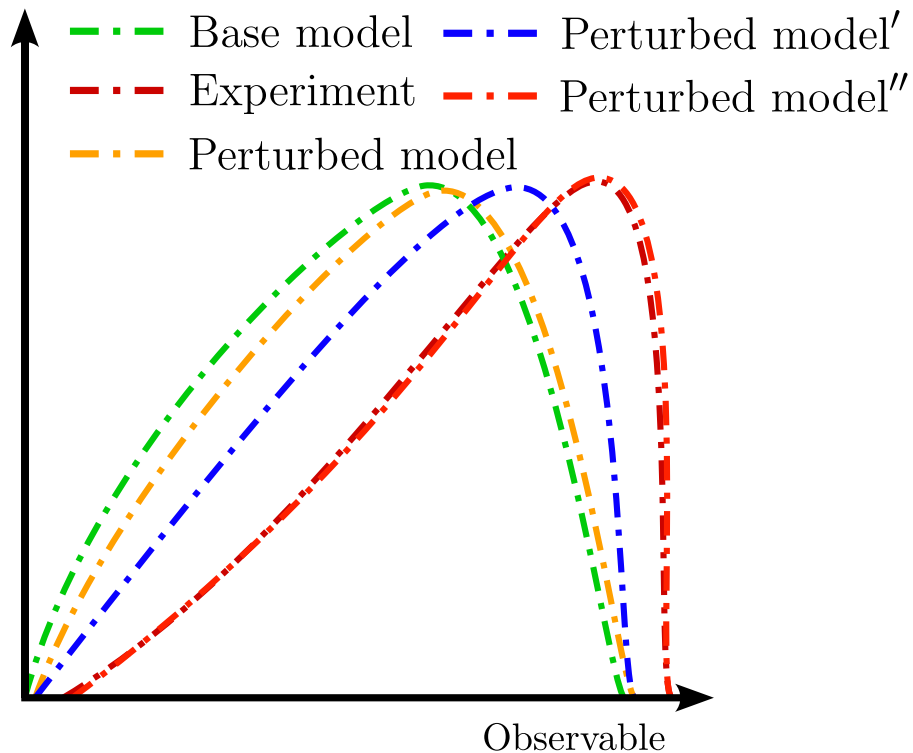
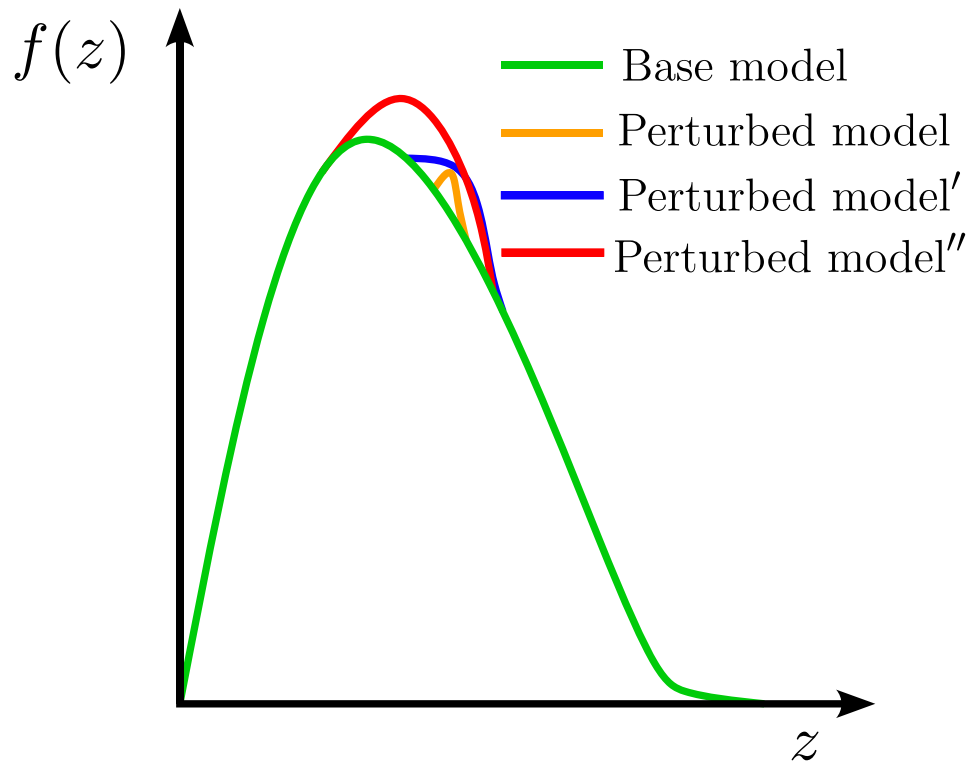
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



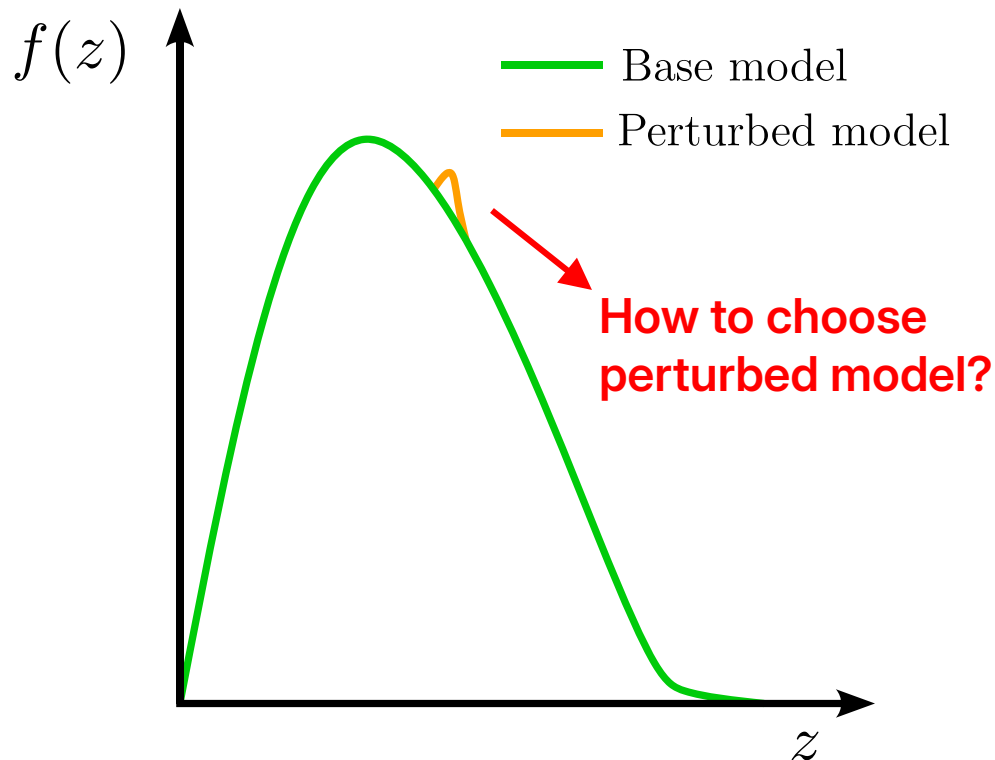
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



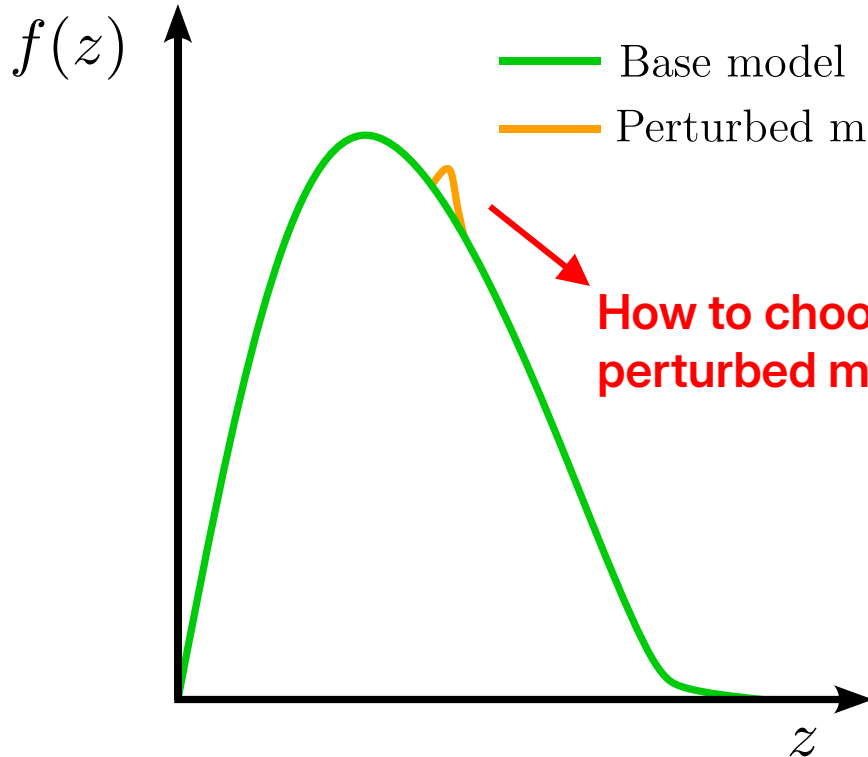
MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



MLHAD efforts: big picture

Solving the “inverse problem of hadronization”



~~ML~~

vs

ML

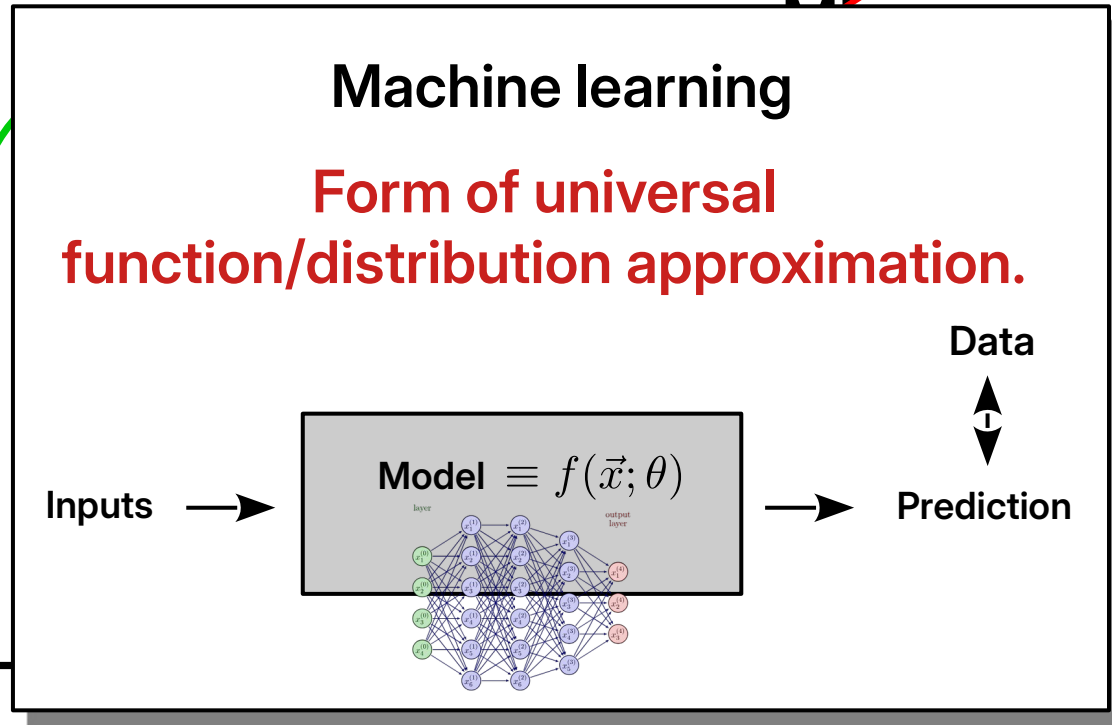
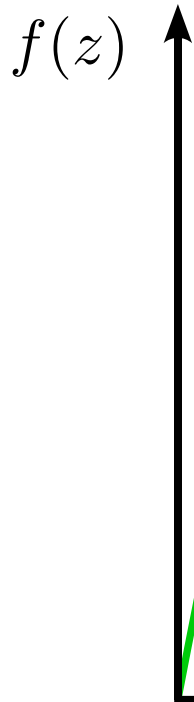
- Vary Lund parameters (traditional “manual”/semi-“manual” tuning)

- ML-based (data-driven) fragmentation function
 - **MAGIC, HOMER**

- Hybrid – keep Lund fragmentation function, promote Lund parameters to differentiable objects
 - **Rejection sampling with autodifferentiation (RSA)**

MLHAD efforts: big picture

Solving the "inverse problem of hadronization"



ML

- ML-based (data-driven) fragmentation function
 - **MAGIC, HOMER**

deep Lund fragmentation function, Lund parameters, tunable objects, on sampling with reweighting (RSA)

The "reweighting revolution"

2308.13459 ,2411.02194, 2505.00142

In collaboration w/

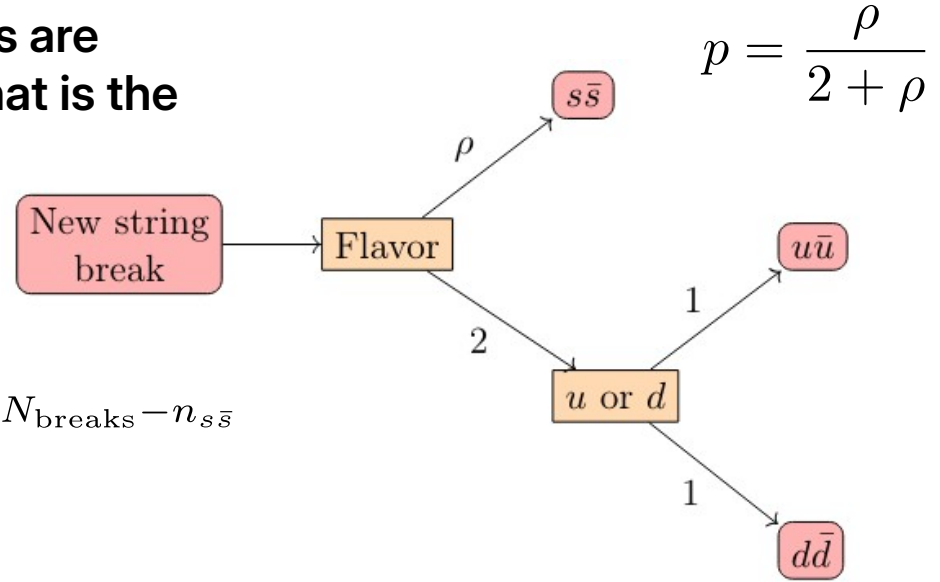
MLHAD 

+

Nick Heller, Ben Nachman, Andrzej Siodmok

Flavor reweighting (2505.00142)

Consider N_{breaks} string breaks. If strange breaks are suppressed by ρ compared to light quarks, what is the probability to producing $n_{s\bar{s}}$ breaks?



$$P(n_{s\bar{s}} | N_{\text{breaks}}) = \binom{N_{\text{breaks}}}{n_{s\bar{s}}} p^{n_{s\bar{s}}} (1 - p)^{N_{\text{breaks}} - n_{s\bar{s}}}$$

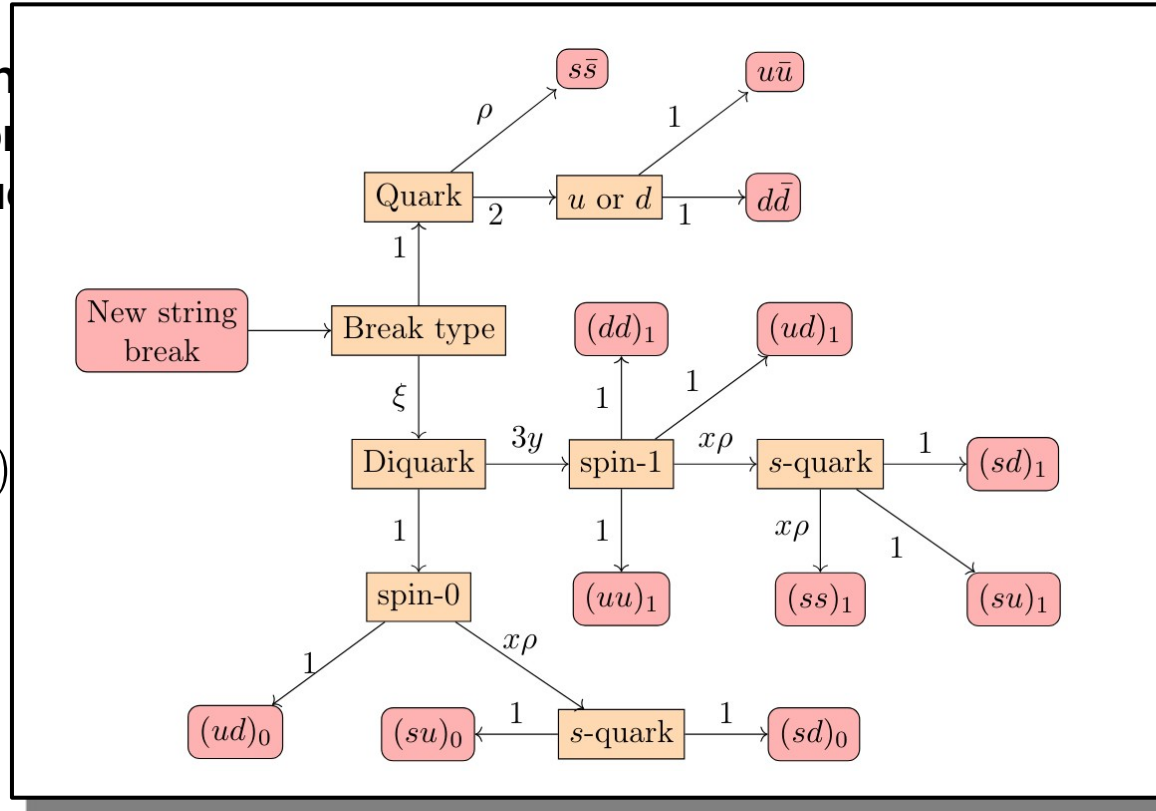
$$w = \frac{P(\rho')}{P(\rho)} = \left(\frac{\rho'}{\rho}\right)^{n_{s\bar{s}}} \left(\frac{1 - \rho'}{1 - \rho}\right)^{N_{\text{breaks}} - n_{s\bar{s}}}$$

Flavor reweighting (2505.00142)

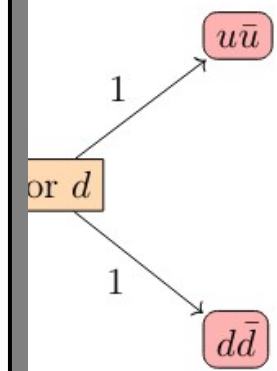
Consider N_{breaks} string breaks suppressed by ρ compared to the probability to produce

$$P(n_{s\bar{s}} | N_{\text{breaks}})$$

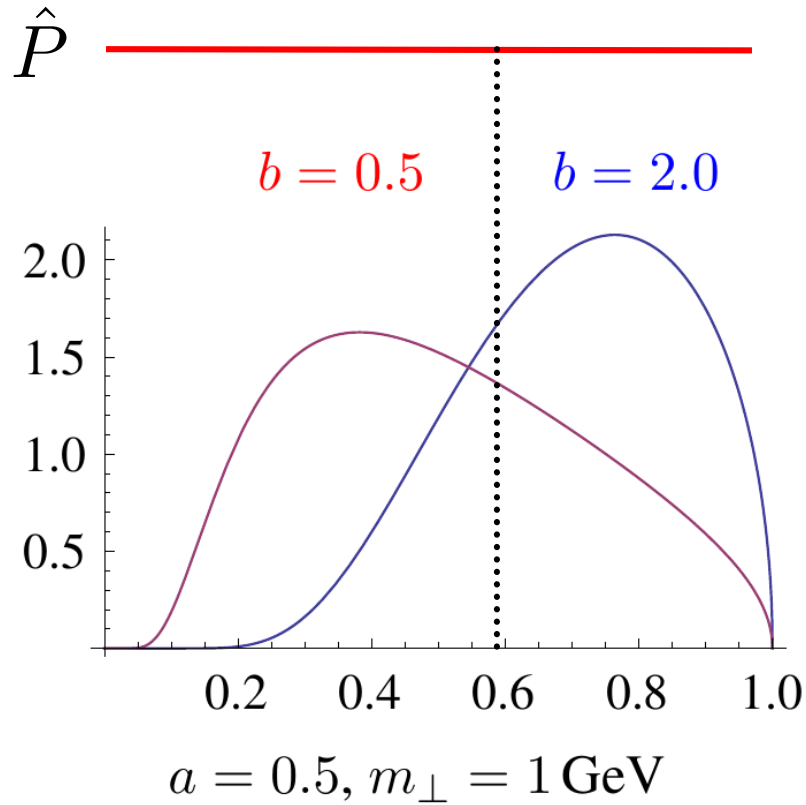
$$w = \frac{P(\rho')}{P(\rho)}$$



$$p = \frac{\rho}{2 + \rho}$$



Kinematic reweighting (2308.13459)



Rejection sampling:

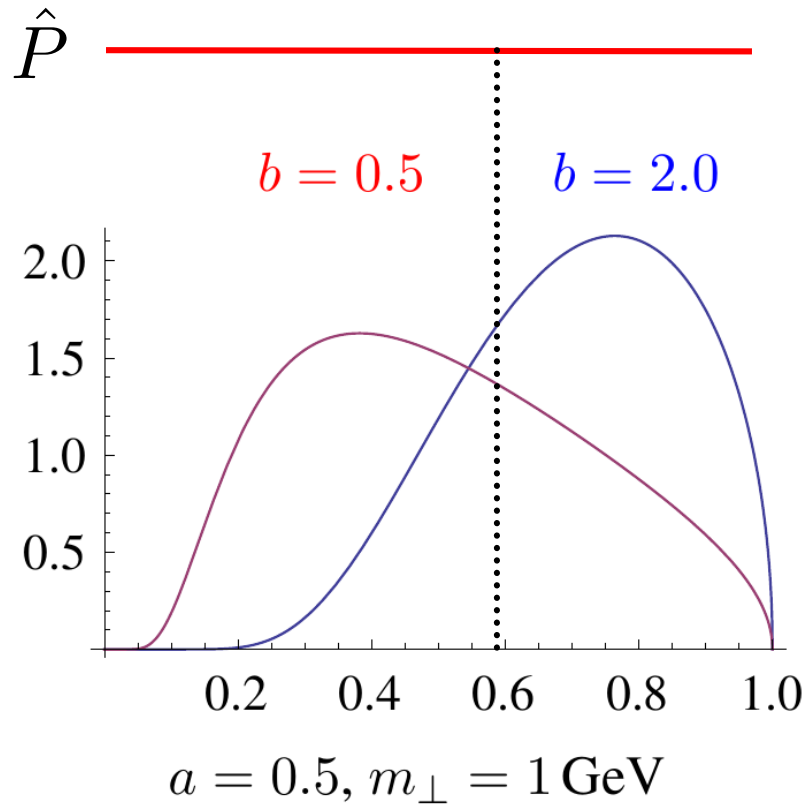
$$\text{Acceptance probability: } P_{\text{accept}} = \frac{p(z, \theta)}{\hat{P}}$$

$$\text{Rejection probability: } P_{\text{reject}} = 1 - P_{\text{accept}}$$

$$w_{\text{accept}} = \frac{P_{\text{accept}}(z; \theta')}{P_{\text{accept}}(z; \theta)}$$

$$w_{\text{reject}} = \frac{1 - P_{\text{accept}}(z; \theta')}{1 - P_{\text{accept}}(z; \theta)}$$

Kinematic reweighting (2308.13459)



Rejection sampling:

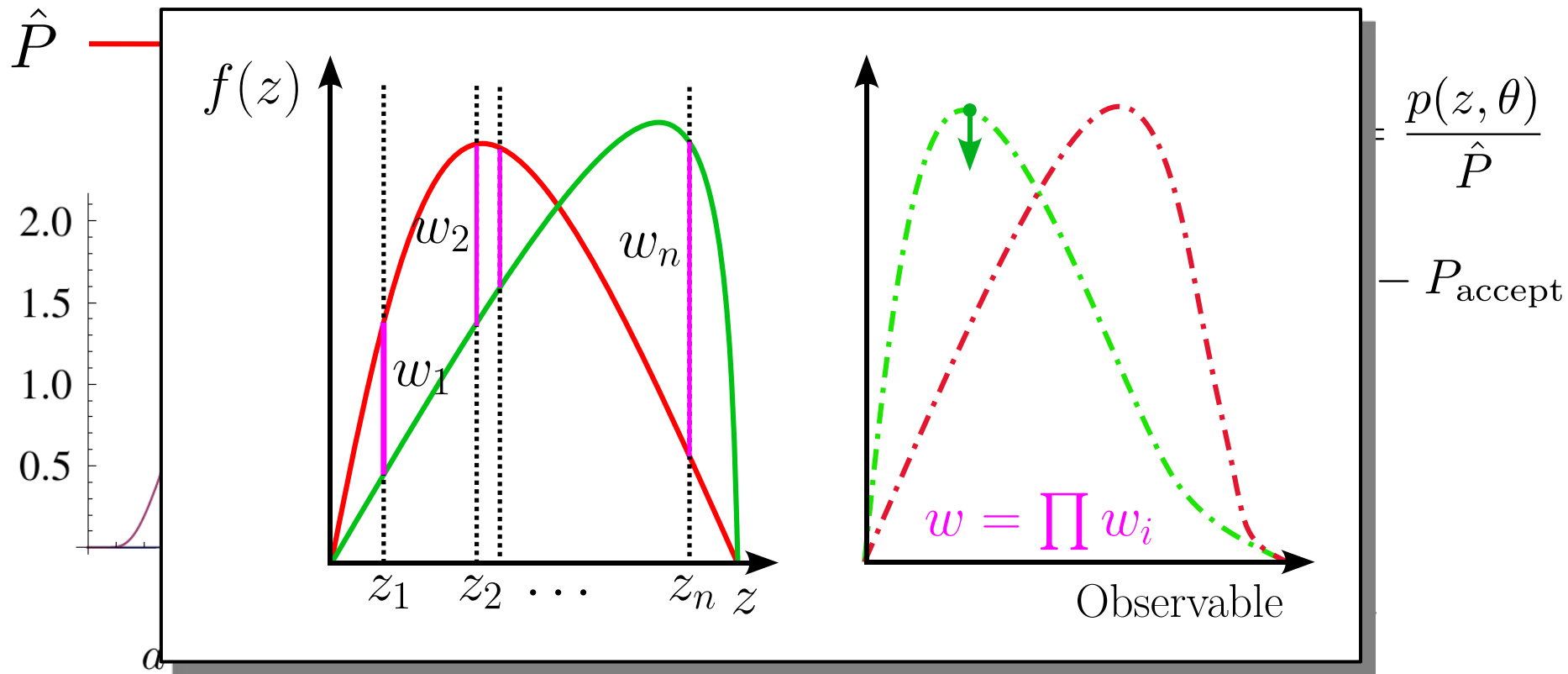
$$\text{Acceptance probability: } P_{\text{accept}} = \frac{p(z, \theta)}{\hat{P}}$$

$$\text{Rejection probability: } P_{\text{reject}} = 1 - P_{\text{accept}}$$

$$w_{\text{accept}} = \frac{p(z; \theta')}{p(z; \theta)}$$

$$w_{\text{reject}} = \frac{\hat{P} - p(z; \theta')}{\hat{P} - p(z; \theta)}$$

Kinematic reweighting (2308.13459)



Kinematic reweighting (2308.13459)

\hat{P}

Rejection sampling:

Data-structure:

$$z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_N = \dots \end{pmatrix} = \begin{pmatrix} \left\{ m_T^{h_1}, z_{\text{accept}}^{h_1}, z_{\text{reject}}^{1,h_1}, \dots, z_{\text{reject}}^{n_{h_1},h_1} \right\} \\ \left\{ m_T^{h_2}, z_{\text{accept}}^{h_2}, z_{\text{reject}}^{1,h_2}, \dots, z_{\text{reject}}^{n_{h_2},h_2} \right\} \\ \left\{ m_T^{h_3}, z_{\text{accept}}^{h_3}, z_{\text{reject}}^{1,h_3}, \dots, z_{\text{reject}}^{n_{h_3},h_3} \right\} \\ \vdots \\ \left\{ m_T^{h_4}, z_{\text{accept}}^{h_4}, z_{\text{reject}}^{1,h_4}, \dots, z_{\text{reject}}^{n_{h_4},h_4} \right\} \\ \vdots \end{pmatrix}$$

$$w_n = \prod_{i=1}^{\tilde{N}_{h,n}} \left(\frac{f(z_{\text{accept}}^{h_i}; \{a, b\}_P)}{f(z_{\text{accept}}^{h_i}; \{a, b\}_B)} \right) \times \prod_{j=1}^{n_{h_i}} \left(\frac{\hat{f} - f(z_{\text{reject}}^{j,h_i}; \{a, b\}_P)}{\hat{f} - f(z_{\text{reject}}^{j,h_i}; \{a, b\}_B)} \right)$$

θ

accept

$a = 0.5, m_{\perp} = 1 \text{ GeV}$

$1 - P(z, \theta)$

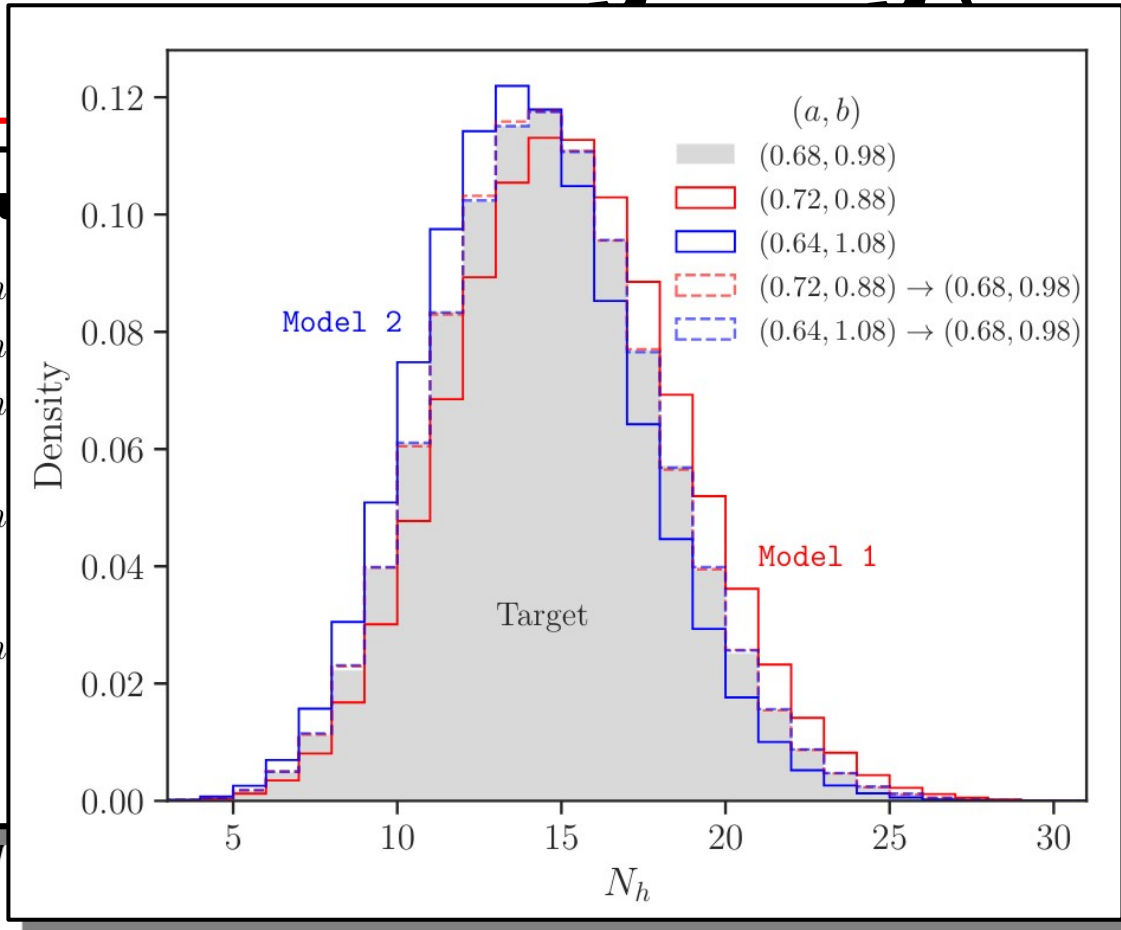
Kinematic reweighting (2308.13459)

\hat{P}

Data-structure

$$z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} \{m_1\} \\ \{m_2\} \\ \{m_3\} \\ \{m_4\} \\ \{m_5\} \\ \{m_6\} \\ \{m_7\} \\ \{m_8\} \\ \{m_9\} \\ \{m_{10}\} \\ \{m_{11}\} \\ \{m_{12}\} \\ \{m_{13}\} \\ \{m_{14}\} \\ \{m_{15}\} \\ \{m_{16}\} \\ \{m_{17}\} \\ \{m_{18}\} \\ \{m_{19}\} \\ \{m_{20}\} \\ \{m_{21}\} \\ \{m_{22}\} \\ \{m_{23}\} \\ \{m_{24}\} \\ \{m_{25}\} \\ \{m_{26}\} \\ \{m_{27}\} \\ \{m_{28}\} \\ \{m_{29}\} \\ \{m_{30}\} \\ \{m_{31}\} \\ \{m_{32}\} \\ \{m_{33}\} \\ \{m_{34}\} \\ \{m_{35}\} \\ \{m_{36}\} \\ \{m_{37}\} \\ \{m_{38}\} \\ \{m_{39}\} \\ \{m_{40}\} \\ \{m_{41}\} \\ \{m_{42}\} \\ \{m_{43}\} \\ \{m_{44}\} \\ \{m_{45}\} \\ \{m_{46}\} \\ \{m_{47}\} \\ \{m_{48}\} \\ \{m_{49}\} \\ \{m_{50}\} \\ \{m_{51}\} \\ \{m_{52}\} \\ \{m_{53}\} \\ \{m_{54}\} \\ \{m_{55}\} \\ \{m_{56}\} \\ \{m_{57}\} \\ \{m_{58}\} \\ \{m_{59}\} \\ \{m_{60}\} \\ \{m_{61}\} \\ \{m_{62}\} \\ \{m_{63}\} \\ \{m_{64}\} \\ \{m_{65}\} \\ \{m_{66}\} \\ \{m_{67}\} \\ \{m_{68}\} \\ \{m_{69}\} \\ \{m_{70}\} \\ \{m_{71}\} \\ \{m_{72}\} \\ \{m_{73}\} \\ \{m_{74}\} \\ \{m_{75}\} \\ \{m_{76}\} \\ \{m_{77}\} \\ \{m_{78}\} \\ \{m_{79}\} \\ \{m_{80}\} \\ \{m_{81}\} \\ \{m_{82}\} \\ \{m_{83}\} \\ \{m_{84}\} \\ \{m_{85}\} \\ \{m_{86}\} \\ \{m_{87}\} \\ \{m_{88}\} \\ \{m_{89}\} \\ \{m_{90}\} \\ \{m_{91}\} \\ \{m_{92}\} \\ \{m_{93}\} \\ \{m_{94}\} \\ \{m_{95}\} \\ \{m_{96}\} \\ \{m_{97}\} \\ \{m_{98}\} \\ \{m_{99}\} \\ \{m_{100}\} \end{pmatrix}$$

$a = 0.5, \tau$



$$\left(\frac{\text{cept}}{\text{cept}} \right)$$

$$\left(\frac{\text{cept}; \{a, b\}_P}{\text{cept}; \{a, b\}_B} \right)$$

$$(z, \sigma)$$

Kinematic reweighting (2308.13459)

\hat{P}

Rejection sampling:

Data-structure:

$$z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_N = \dots \end{pmatrix} = \begin{pmatrix} \left\{ m_T^{h_1}, z_{\text{accept}}^{h_1}, z_{\text{reject}}^{1,h_1}, \dots, z_{\text{reject}}^{n_{h_1},h_1} \right\} \\ \left\{ m_T^{h_2}, z_{\text{accept}}^{h_2}, z_{\text{reject}}^{1,h_2}, \dots, z_{\text{reject}}^{n_{h_2},h_2} \right\} \\ \left\{ m_T^{h_3}, z_{\text{accept}}^{h_3}, z_{\text{reject}}^{1,h_3}, \dots, z_{\text{reject}}^{n_{h_3},h_3} \right\} \\ \left\{ m_T^{h_4}, z_{\text{accept}}^{h_4}, z_{\text{reject}}^{1,h_4}, \dots, z_{\text{reject}}^{n_{h_4},h_4} \right\} \\ \vdots \\ \left\{ m_T^{h_N}, z_{\text{accept}}^{h_N}, z_{\text{reject}}^{1,h_N}, \dots, z_{\text{reject}}^{n_{h_N},h_N} \right\} \end{pmatrix}$$

Differentiable! → RSA

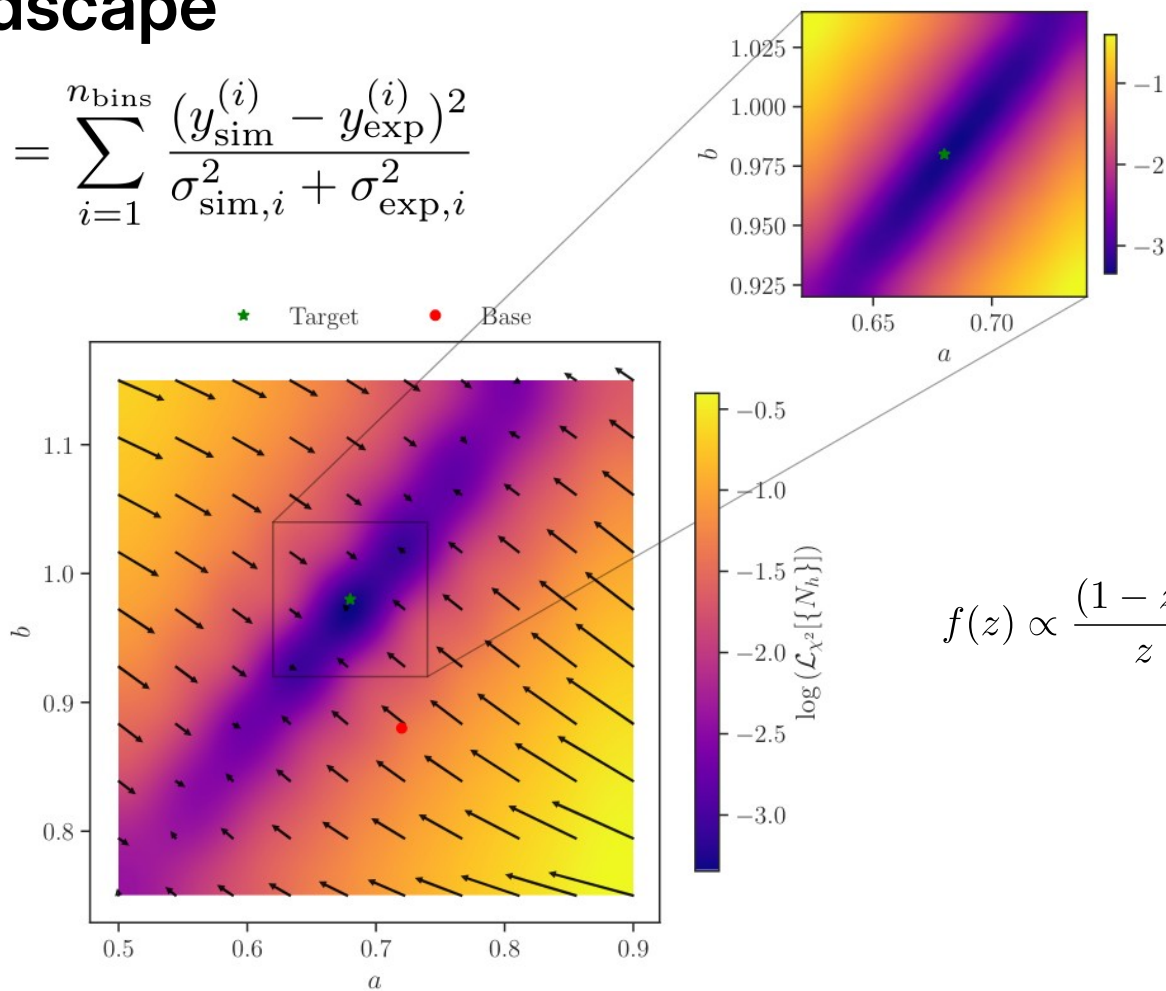
$$w_n = \prod_{i=1}^{\tilde{N}_{h,n}} \left(\frac{f(z_{\text{accept}}^{h_i}; \{a, b\}_P)}{f(z_{\text{accept}}^{h_i}; \{a, b\}_B)} \right) \times \prod_{j=1}^{n_{h_i}} \left(\frac{\hat{f} - f(z_{\text{reject}}^{j,h_i}; \{a, b\}_P)}{\hat{f} - f(z_{\text{reject}}^{j,h_i}; \{a, b\}_B)} \right)$$

$a = 0.5, m_{\perp} = 1 \text{ GeV}$

$1 - P(z, \theta)$

χ^2 -loss landscape

$$\mathcal{L}_{\chi^2}(\mathbf{y}_{\text{sim}}, \mathbf{y}_{\text{exp}}; \mathbf{w}) = \sum_{i=1}^{n_{\text{bins}}} \frac{(y_{\text{sim}}^{(i)} - y_{\text{exp}}^{(i)})^2}{\sigma_{\text{sim},i}^2 + \sigma_{\text{exp},i}^2}$$

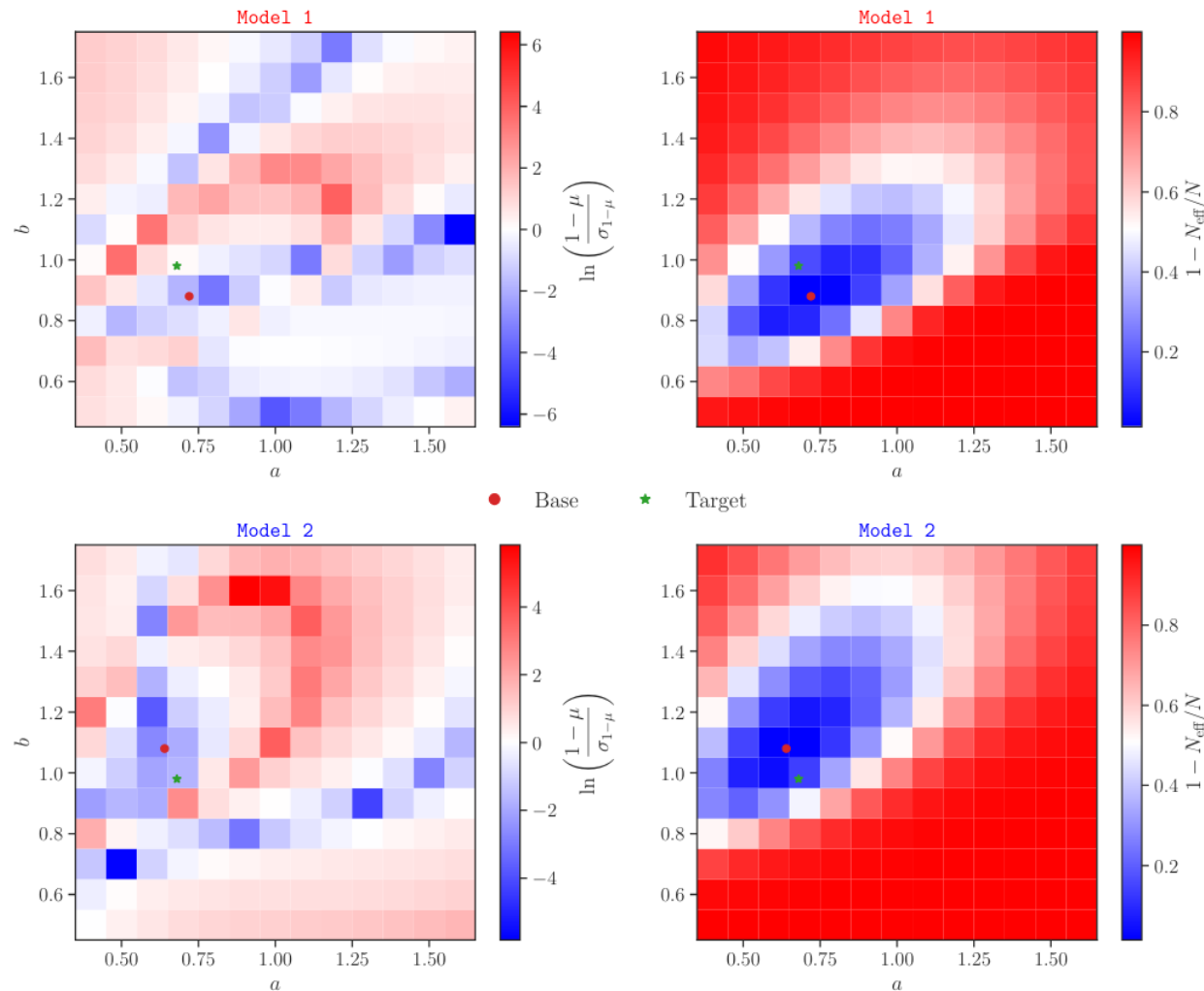


$$f(z) \propto \frac{(1-z)^a}{z} \exp\left(\frac{-bm_T^2}{z}\right)$$

No free lunch

Statistical power drops off quickly as you move away from the base parameterization

$$\mu \equiv \sum_{i=1}^N \frac{w_i}{N}, \quad N_{\text{eff}} = \frac{\left(\sum_{i=1}^N w_i\right)^2}{\sum_{i=1}^N w_i^2}$$



ML-based observables

- Train a deepsets classifier to distinguish simulation from experiment

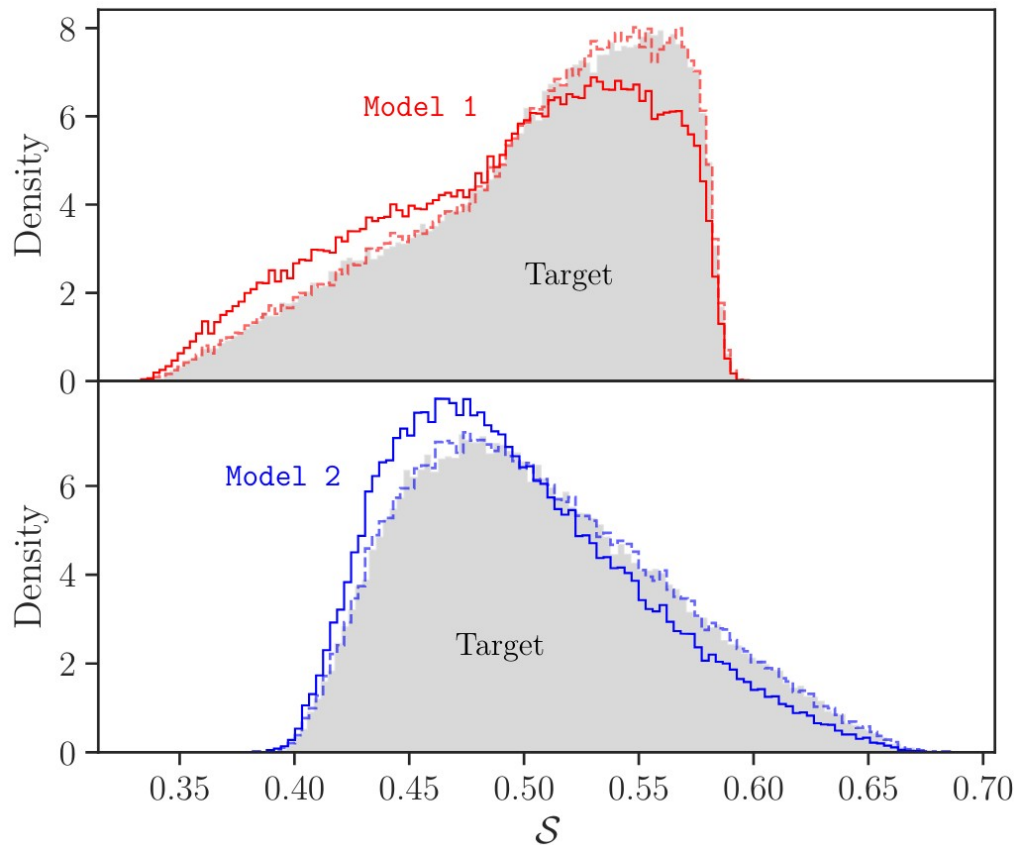
- Takes full event information as input

$(E, p_x, p_y, p_z)_1$

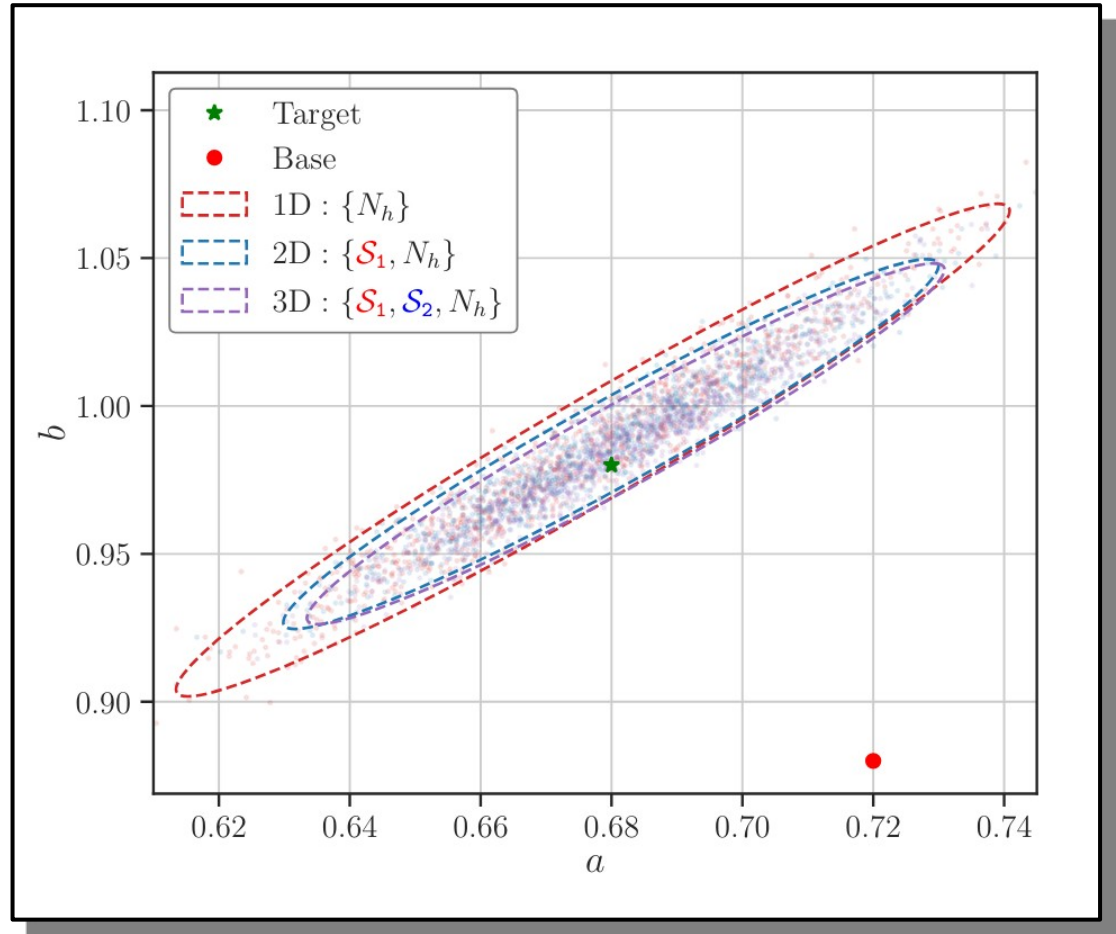
$(E, p_x, p_y, p_z)_2$

...

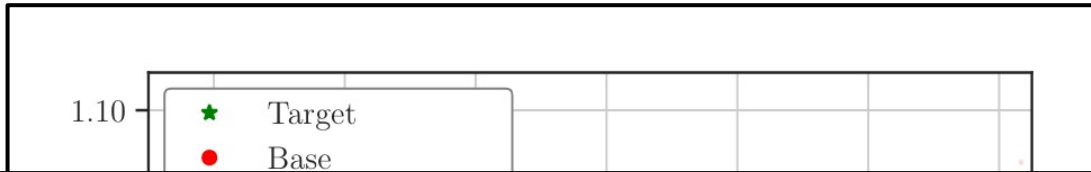
Use trained classifier output
(score) as an observable



Classifier score with full event info improves tuning convergence

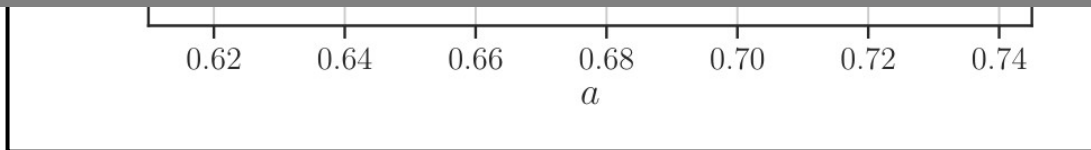


Clas
info
con



Analytic form of $f(z)$ is useful but potentially limiting when trying to describe data.

Can we extract the fragmentation function directly from data?



Microscopic Alterations Generated from IR Collections

+ (2311.09296, 25xx.xxxx)

Histories and Observables for Monte Carlo Event Reweighting

(2410.06342, 2503.05667, 2509.03592)

MLHAD 

MAGIC

+ (2311.09296, 25xx.xxxx)

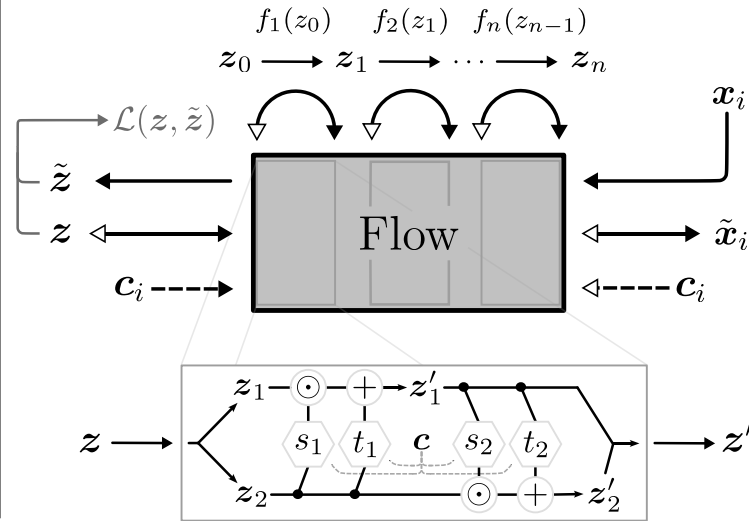
HOMER

(2410.06342, 2503.05667, 2509.03592)

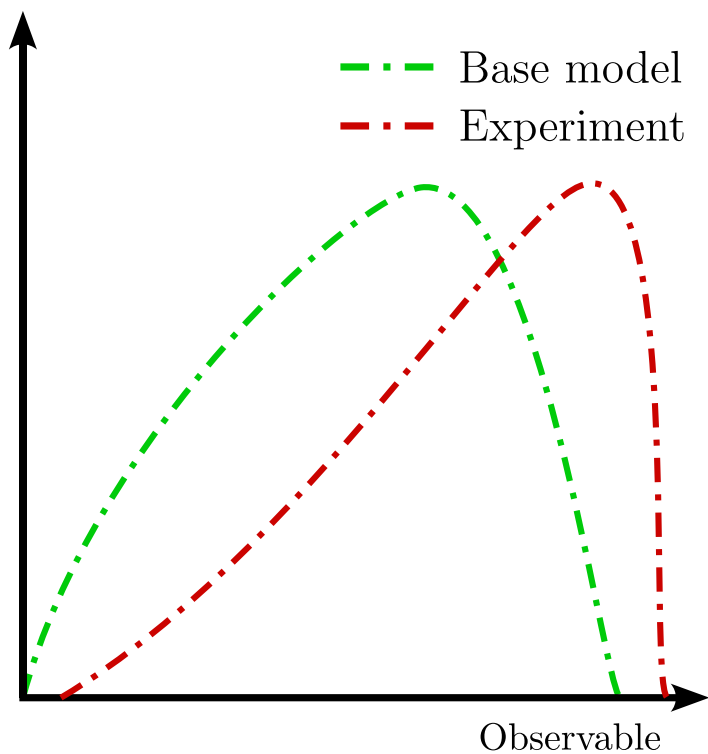
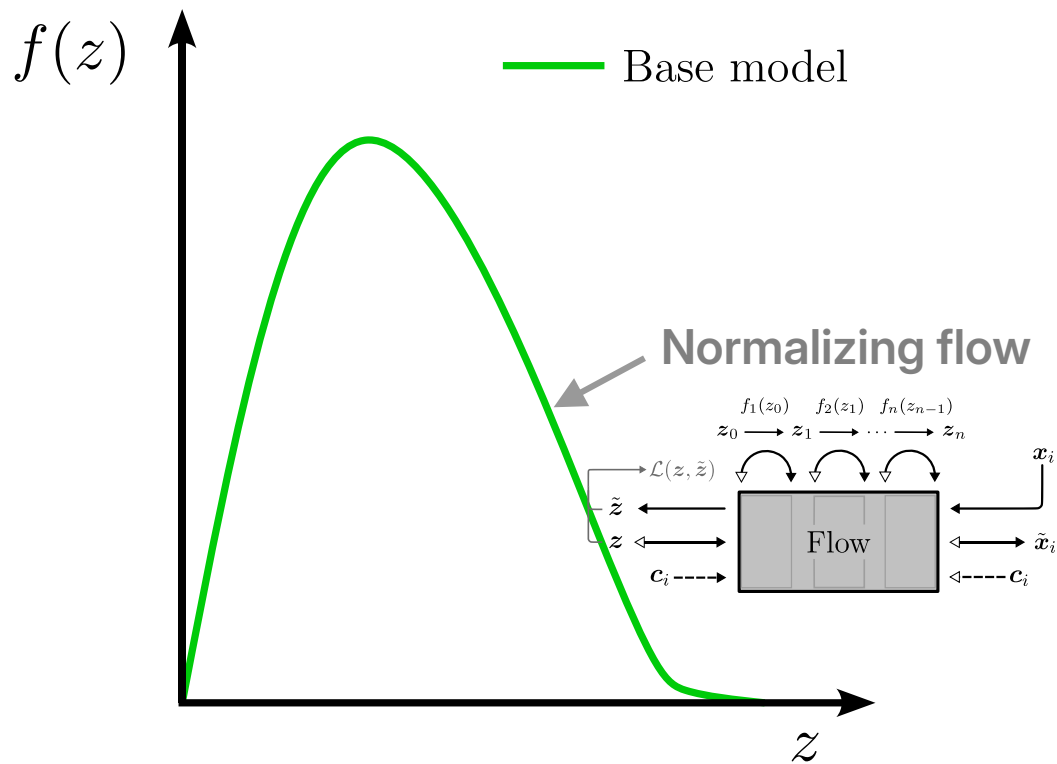
MLHAD 

MAGIC: Invertible neural networks (INN)

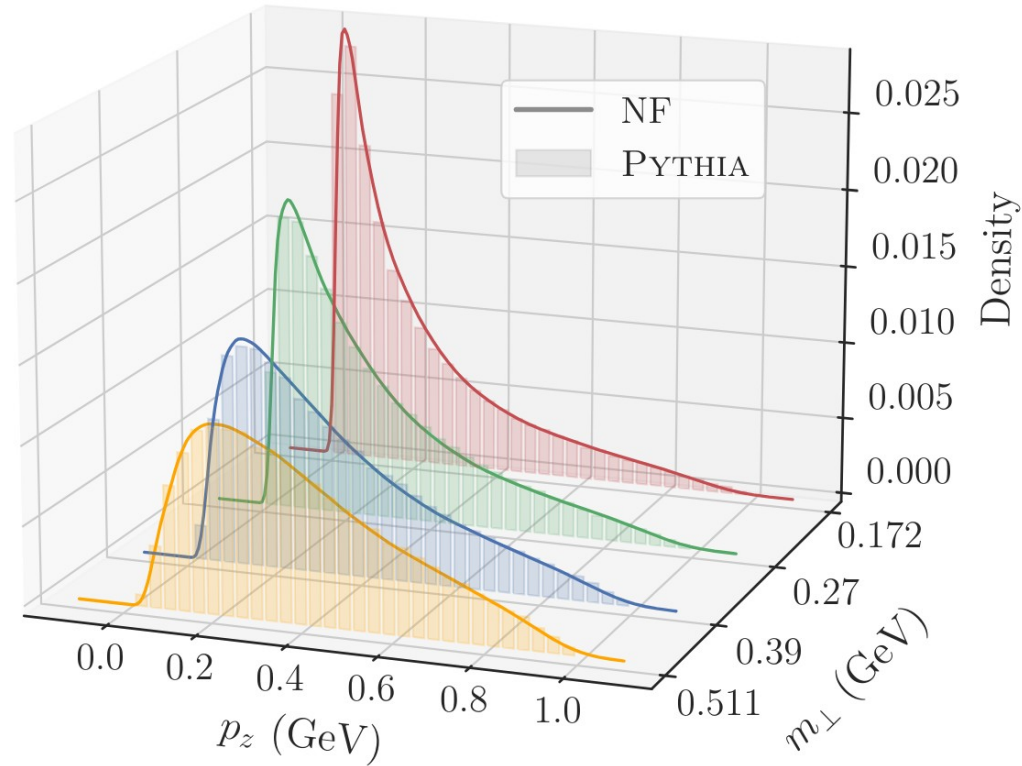
a.k.a normalizing flow



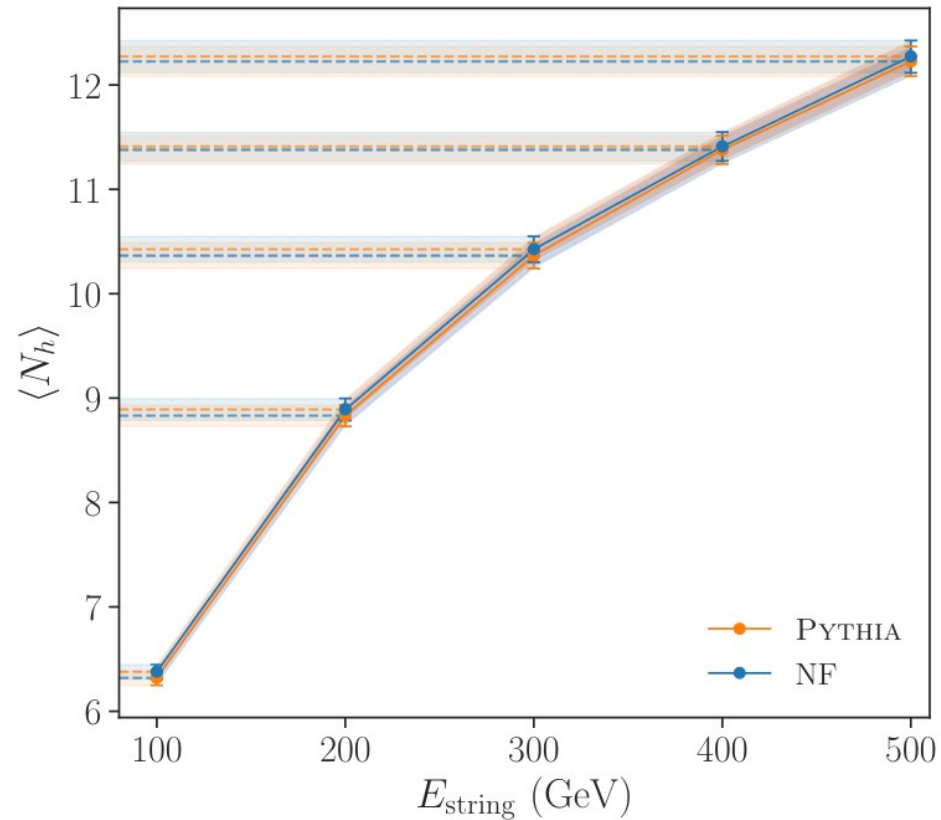
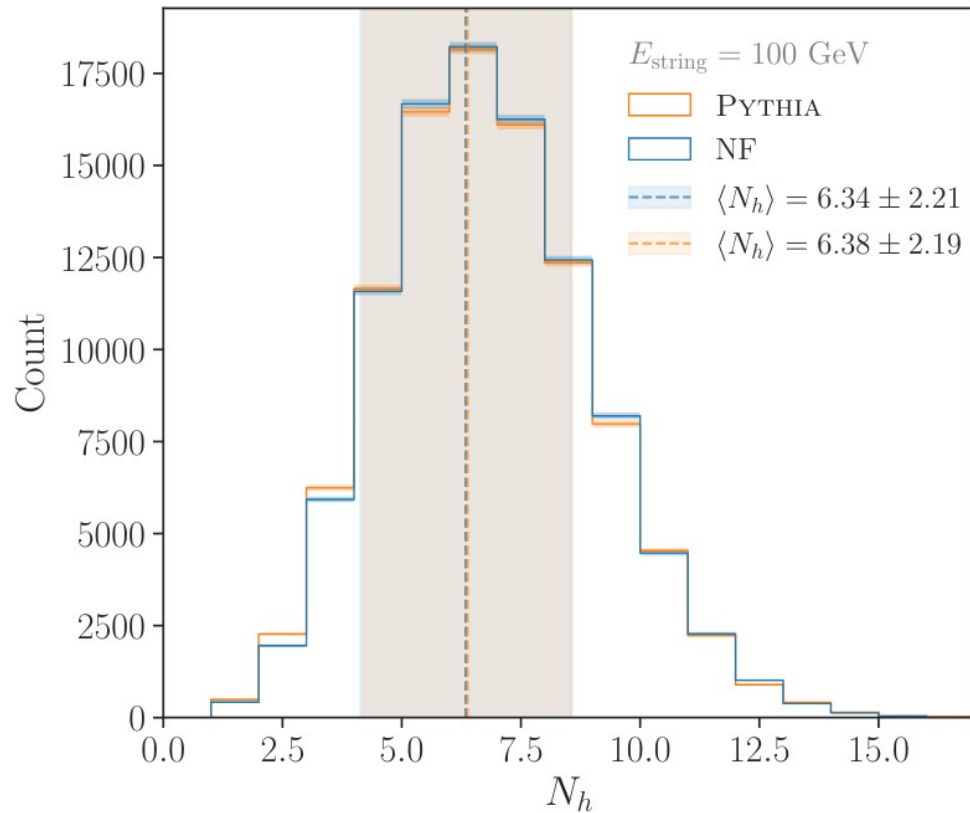
MLHAD efforts: MAGIC



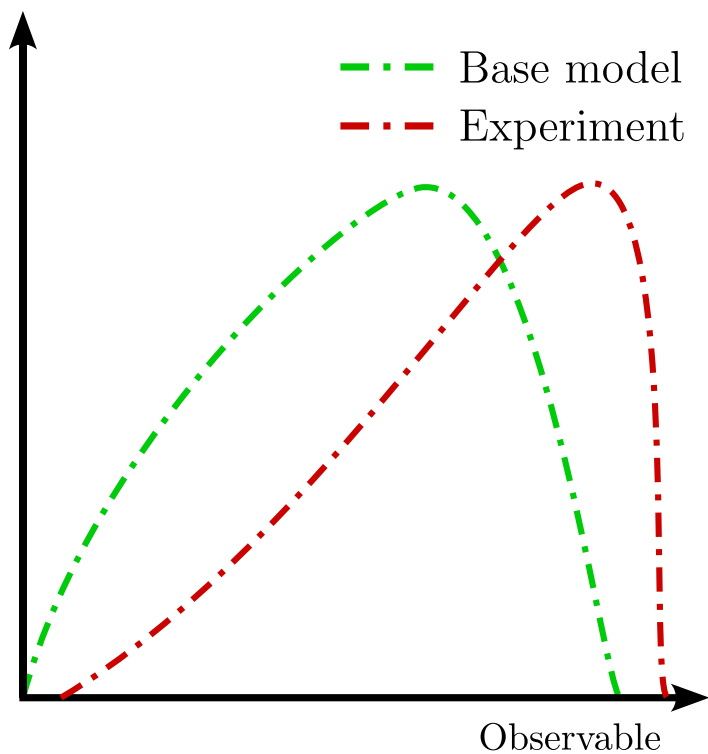
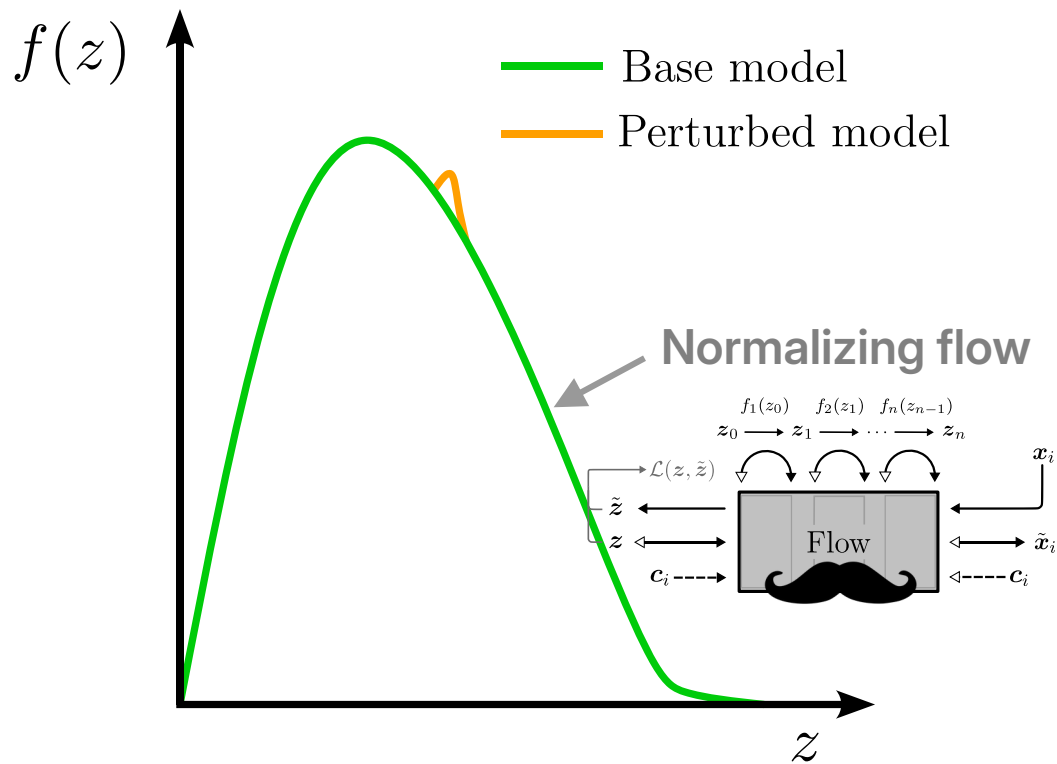
MAGIC: Base model



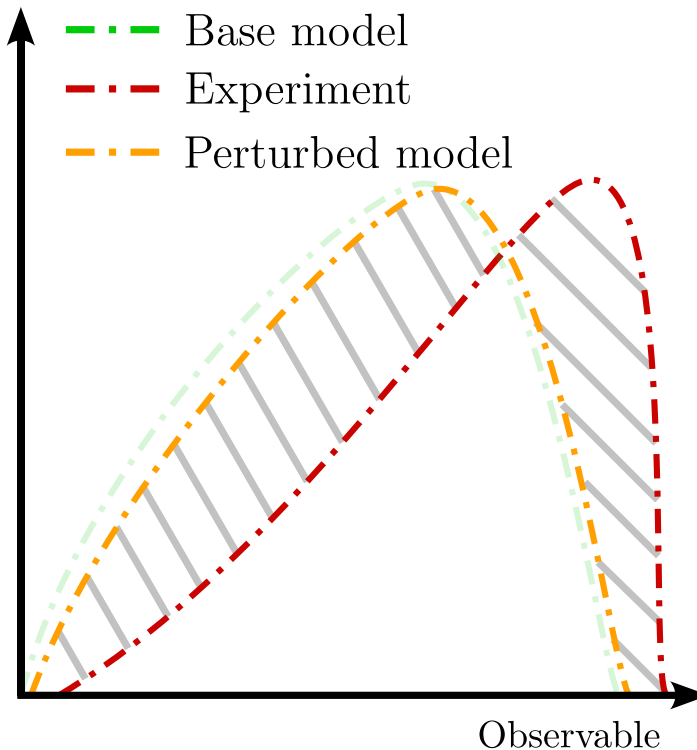
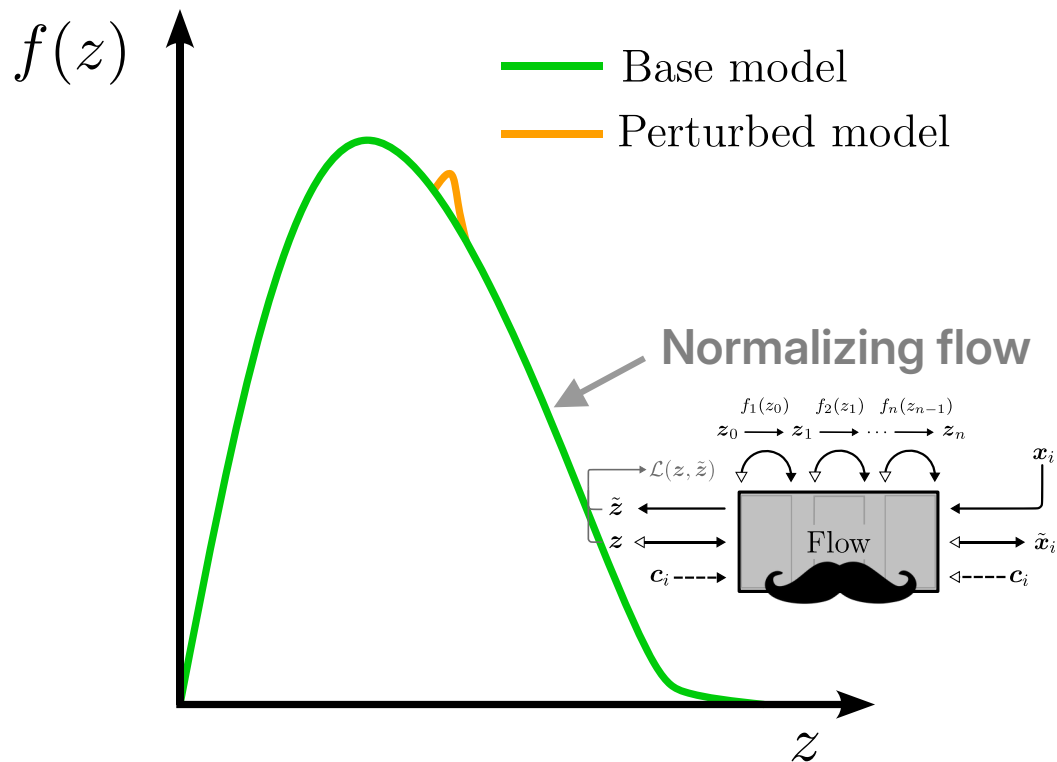
MAGIC: Base model



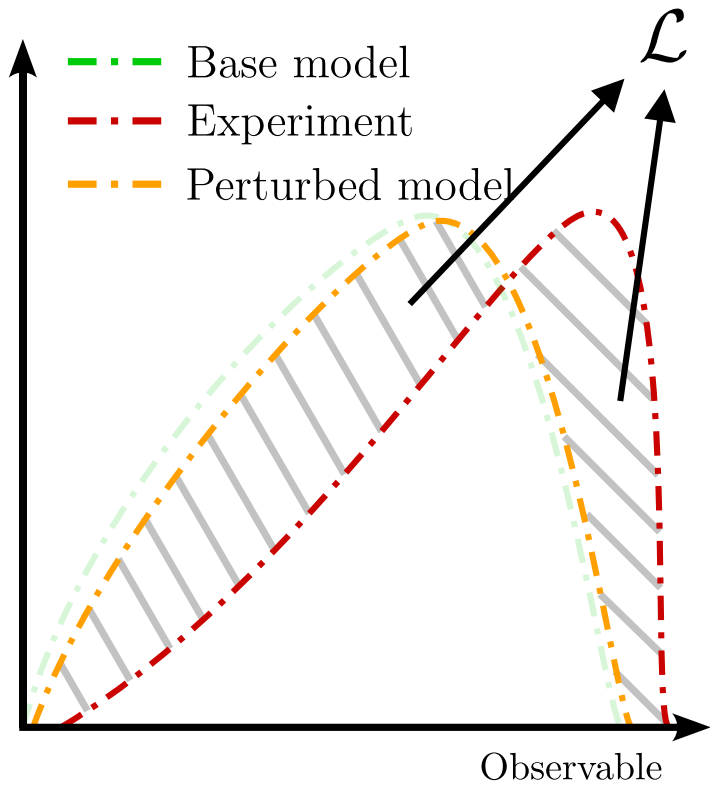
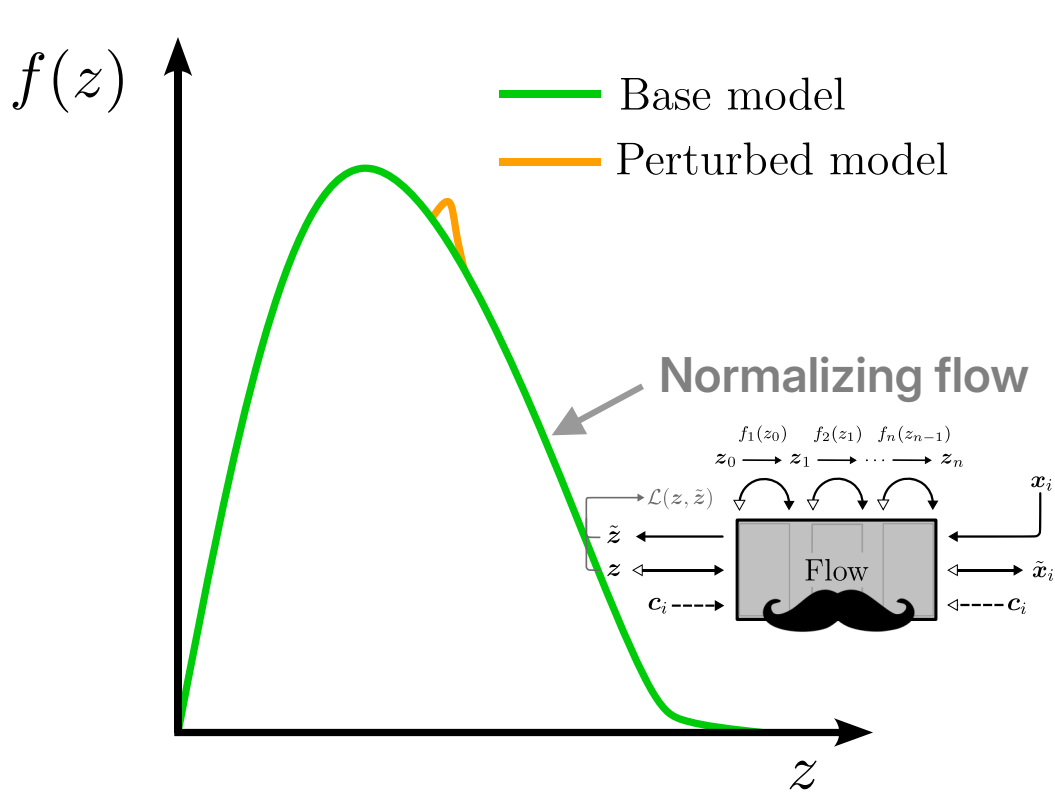
MLHAD efforts: **MAGIC**



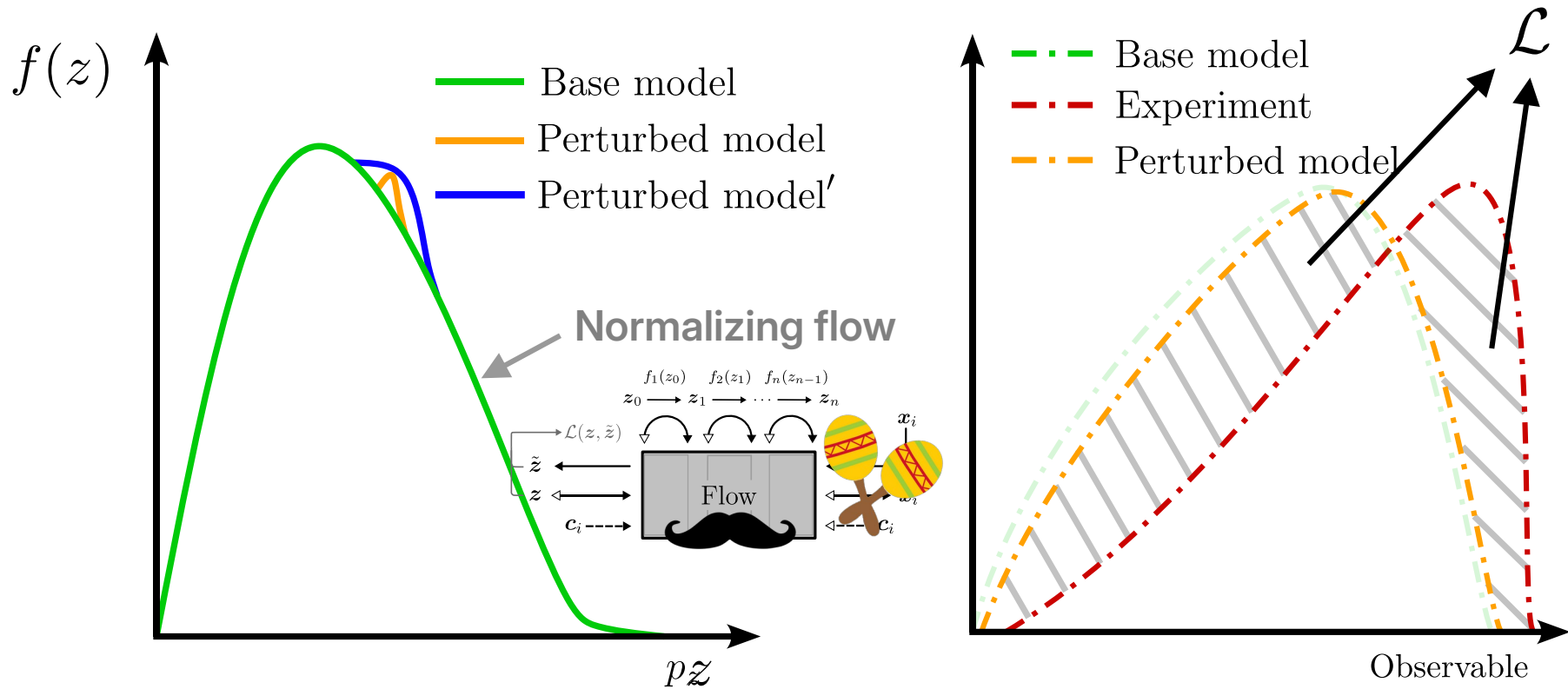
MLHAD efforts: **MAGIC**



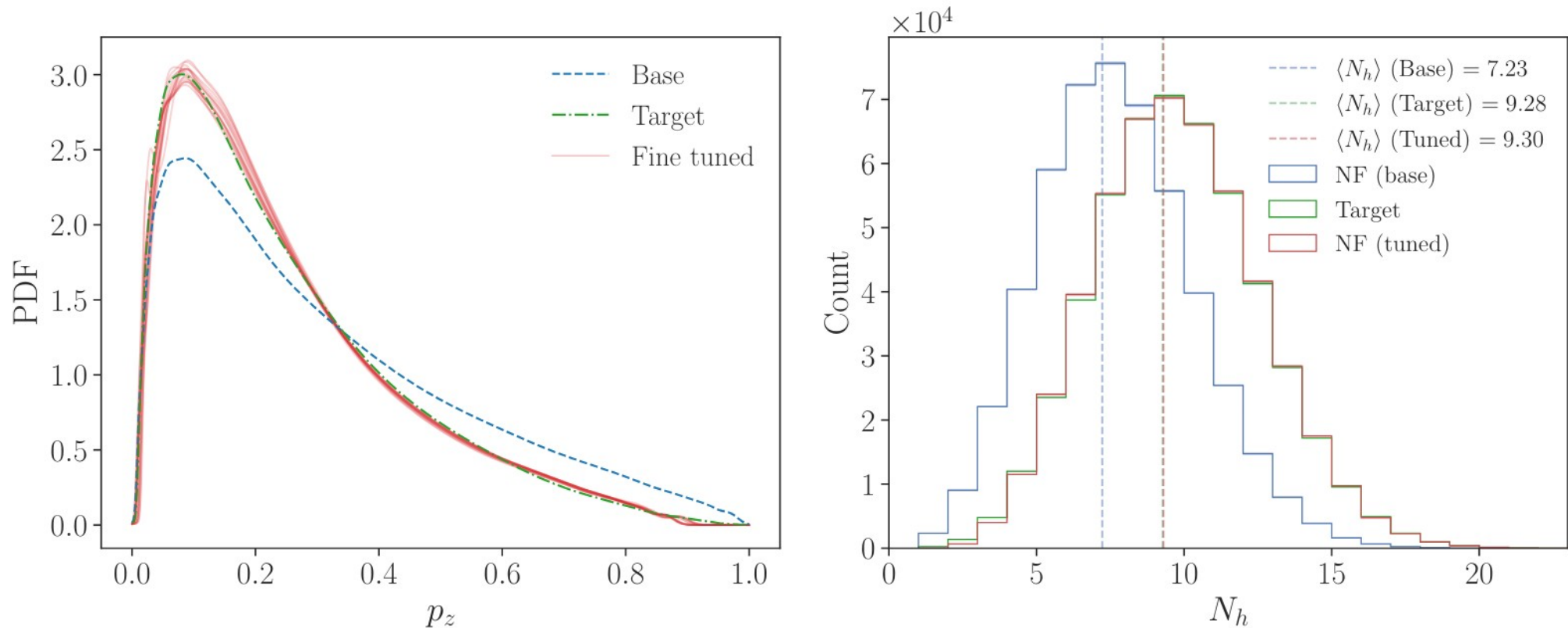
MLHAD efforts: **MAGIC**



MLHAD efforts: MAGIC



MAGIC: $q\bar{q}$

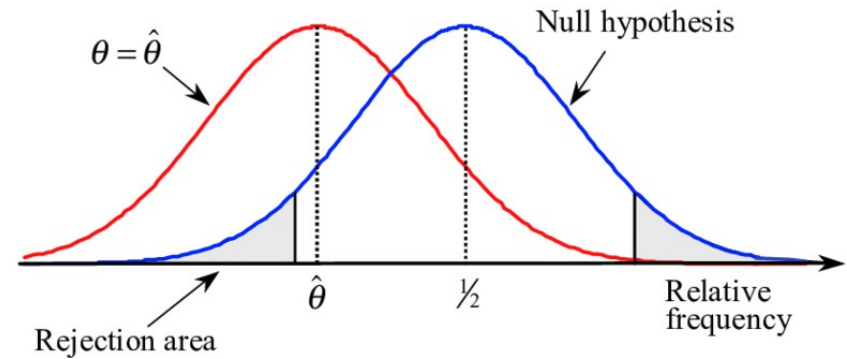


HOMER

$$\operatorname{argmin}_f L[f] = \frac{p(x | \theta_0)}{p(x | \theta_1)} = \mathcal{L}(x)$$

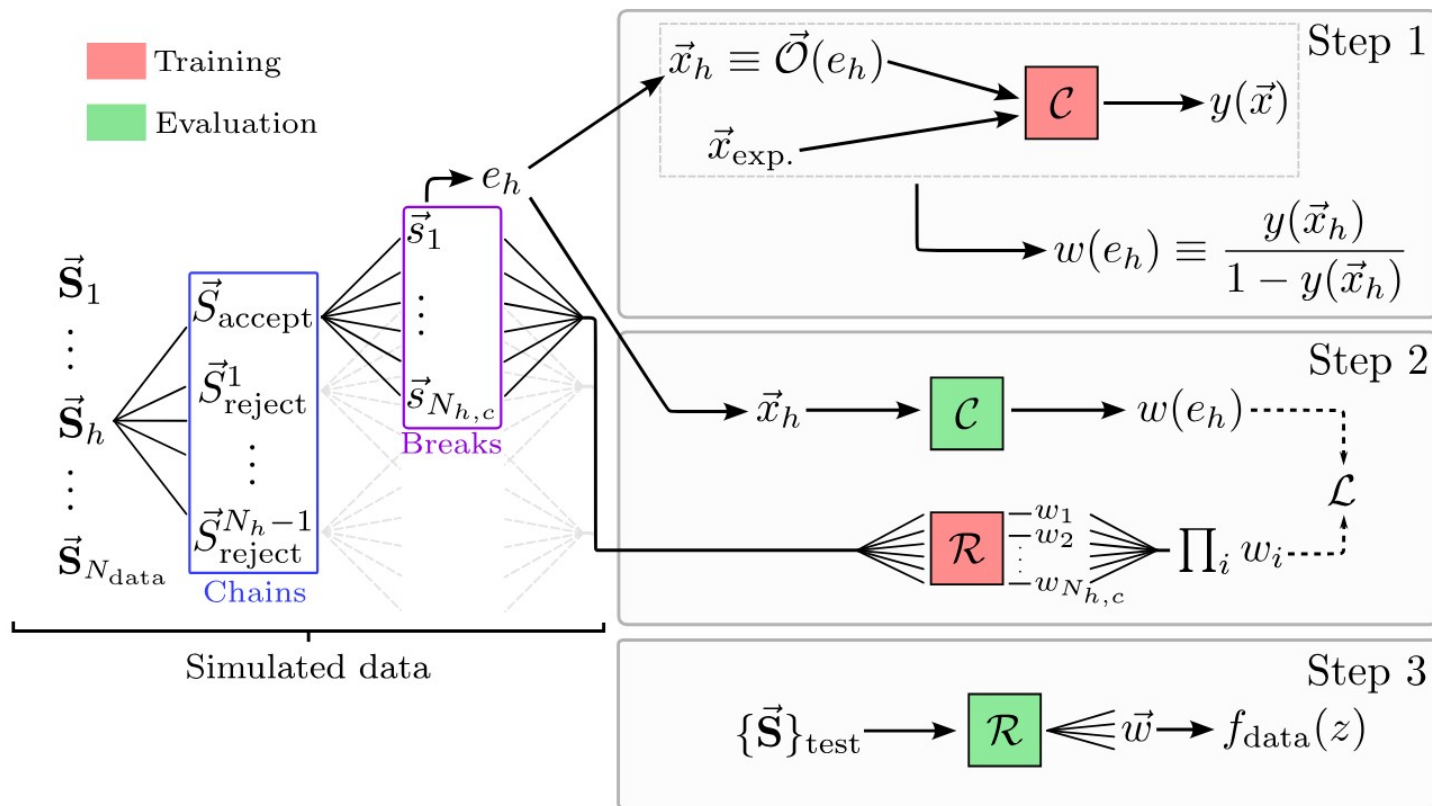
Minimize binary cross-entropy

$$f(x) = \frac{p(x | \theta_0)}{p(x | \theta_0) + p(x | \theta_1)}$$

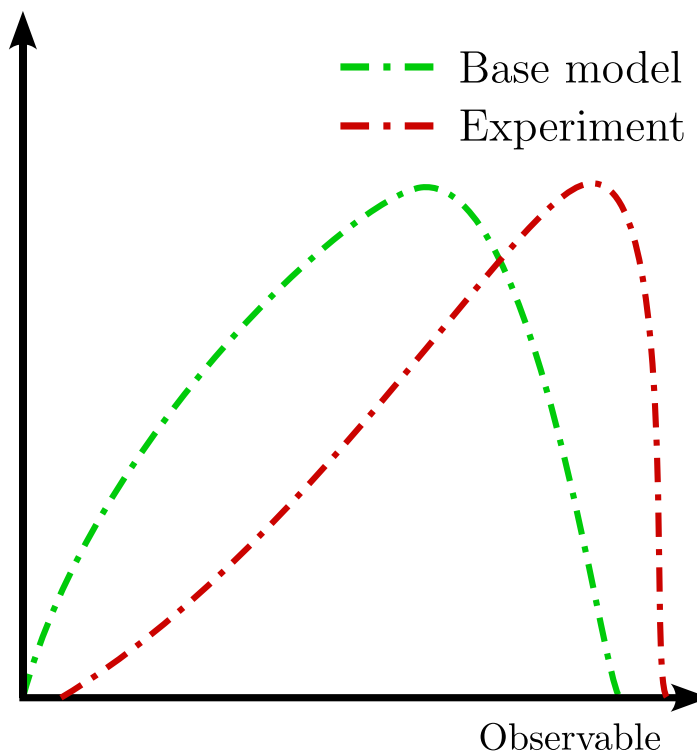
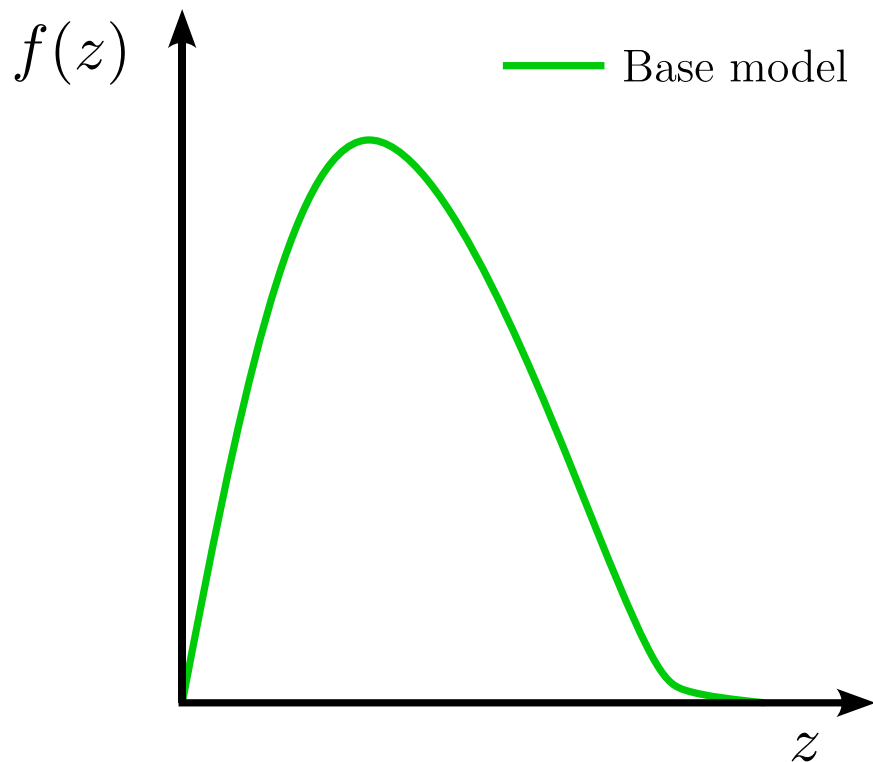


$$\frac{f(x)}{1 - f(x)} = \frac{\frac{p(x|\theta_0)}{p(x|\theta_0)+p(x|\theta_1)}}{1 - \frac{p(x|\theta_0)}{p(x|\theta_0)+p(x|\theta_1)}} = \frac{p(x | \theta_0)}{p(x | \theta_0) + p(x | \theta_1) - p(x | \theta_0)} = \frac{p(x | \theta_0)}{p(x | \theta_1)} = \mathcal{L}(x)$$

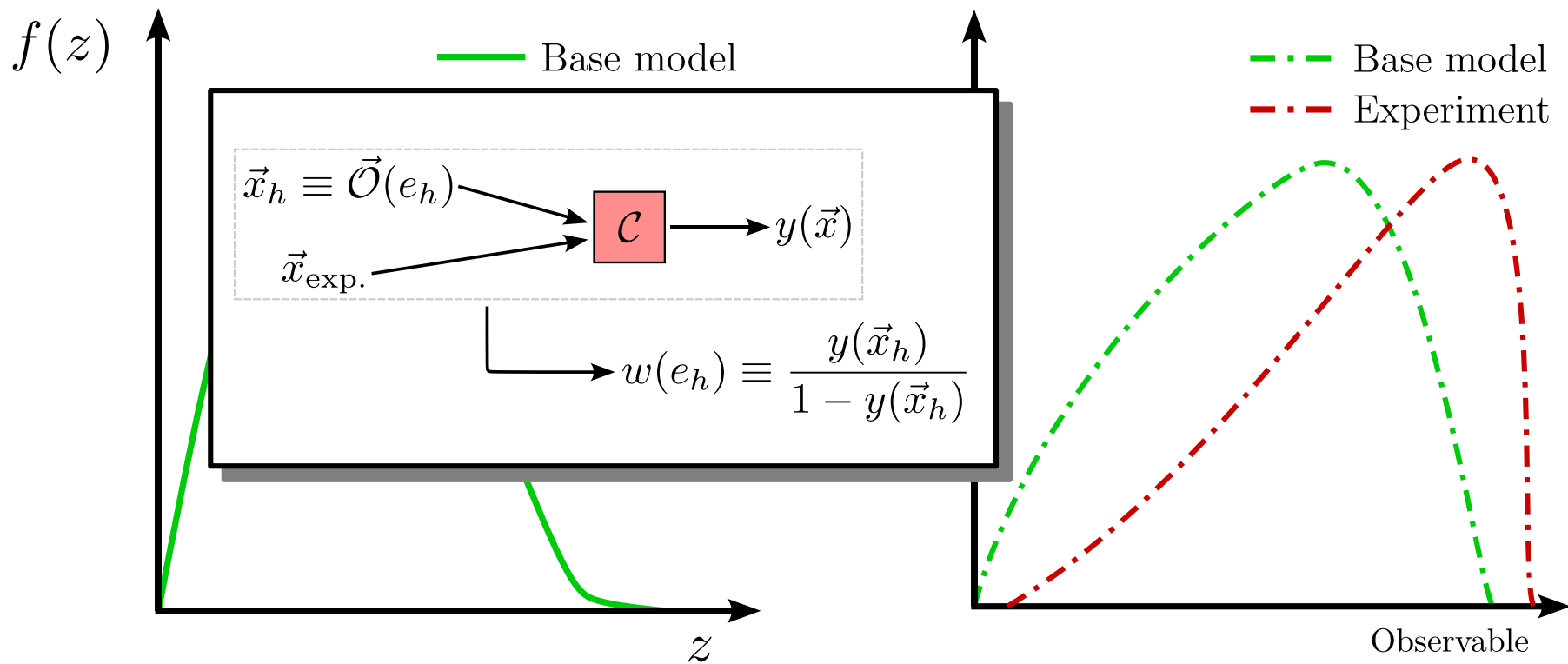
HOMER: $q\bar{q}$



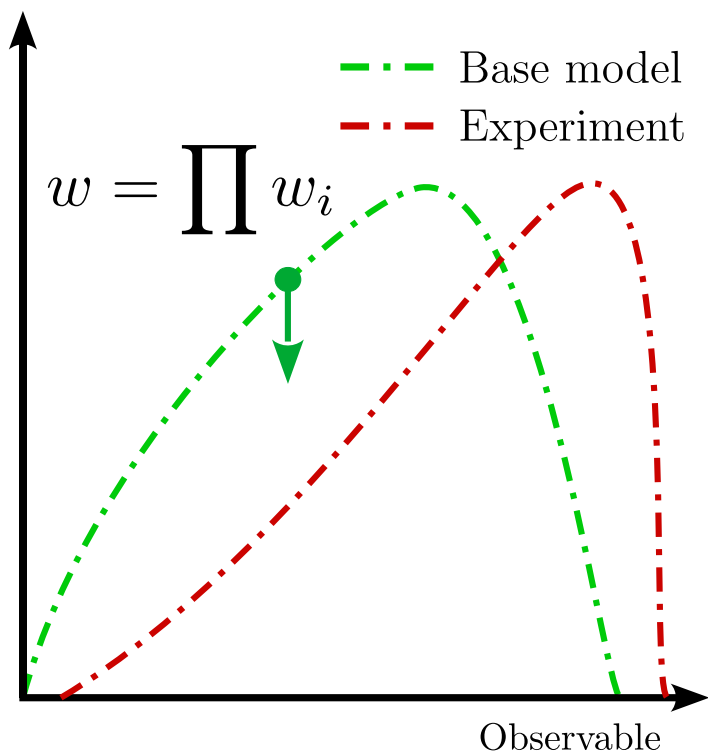
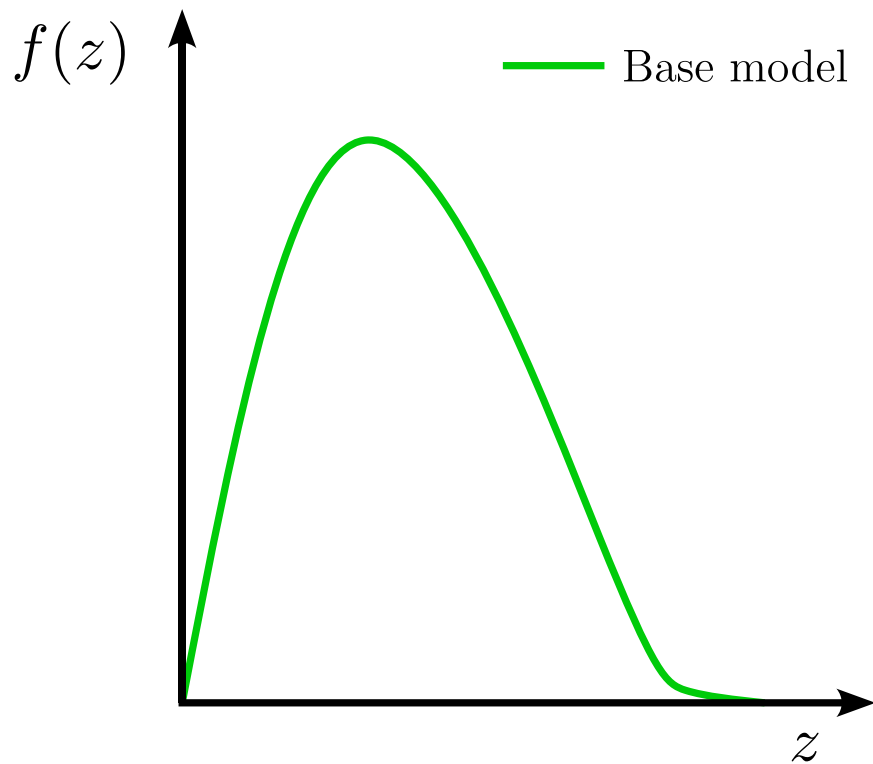
MLHAD efforts: HOMER



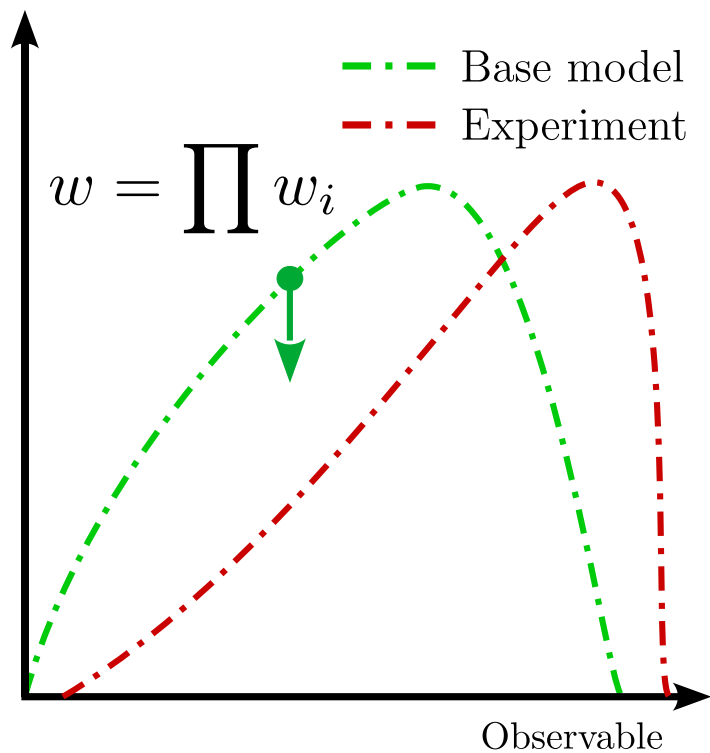
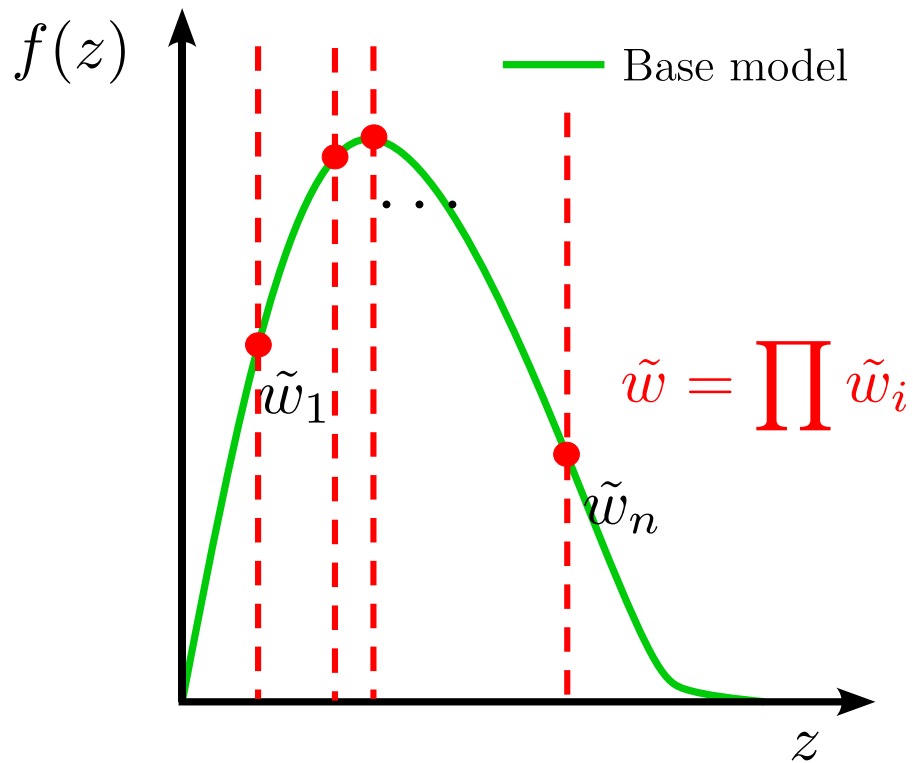
MLHAD efforts: HOMER



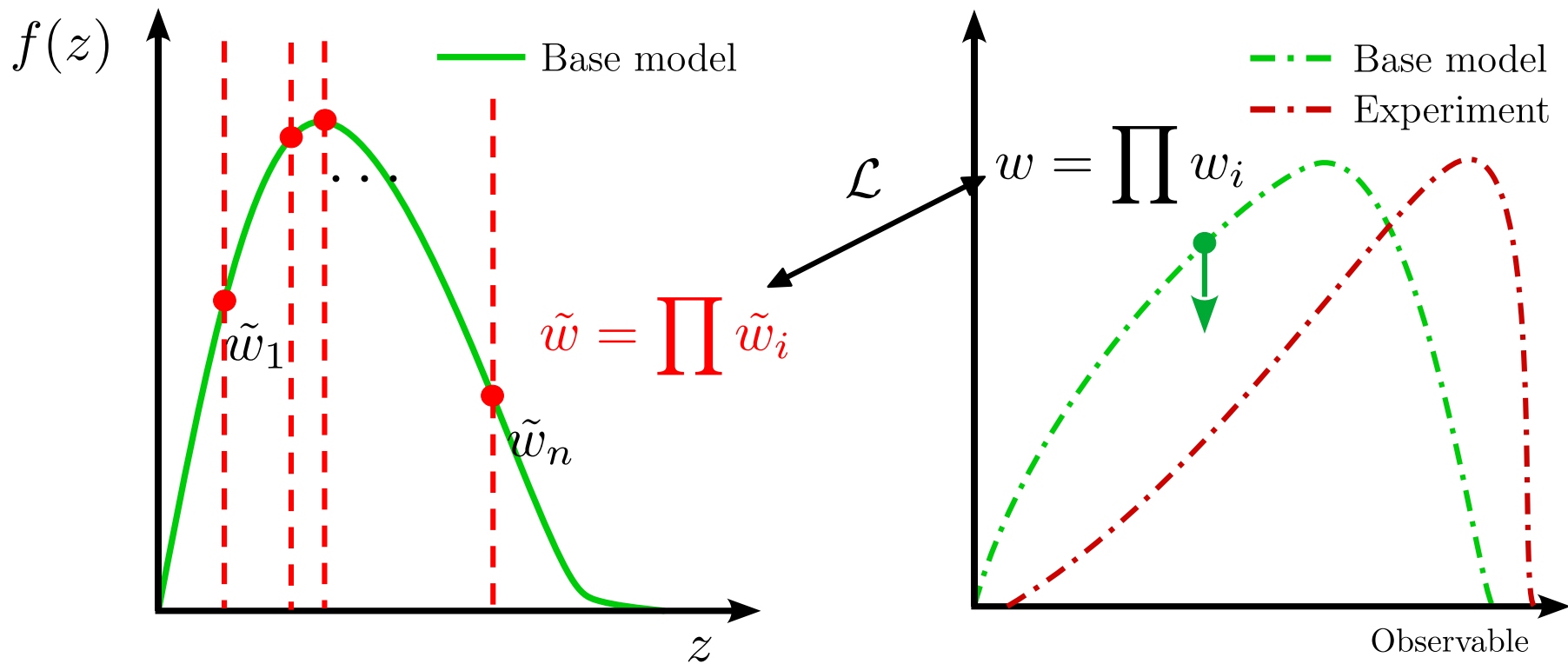
MLHAD efforts: HOMER



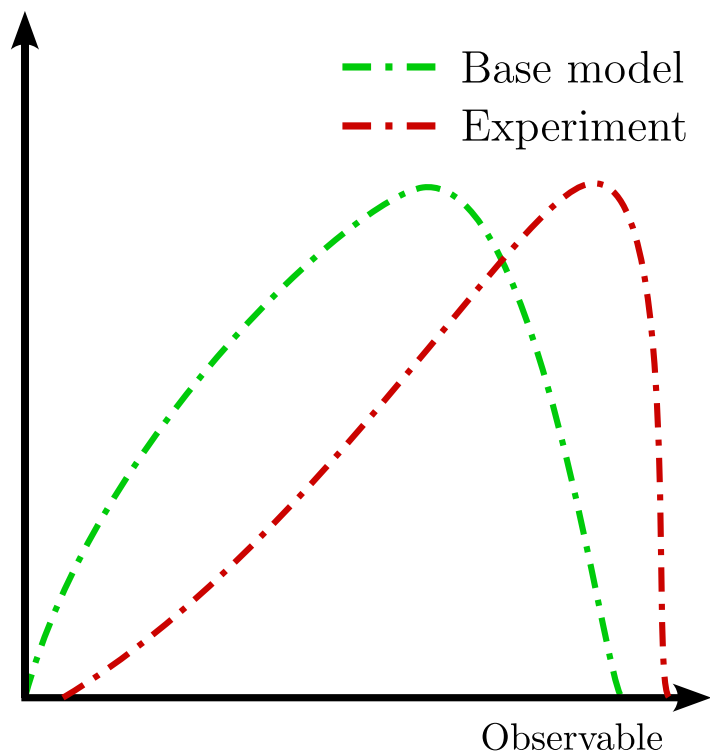
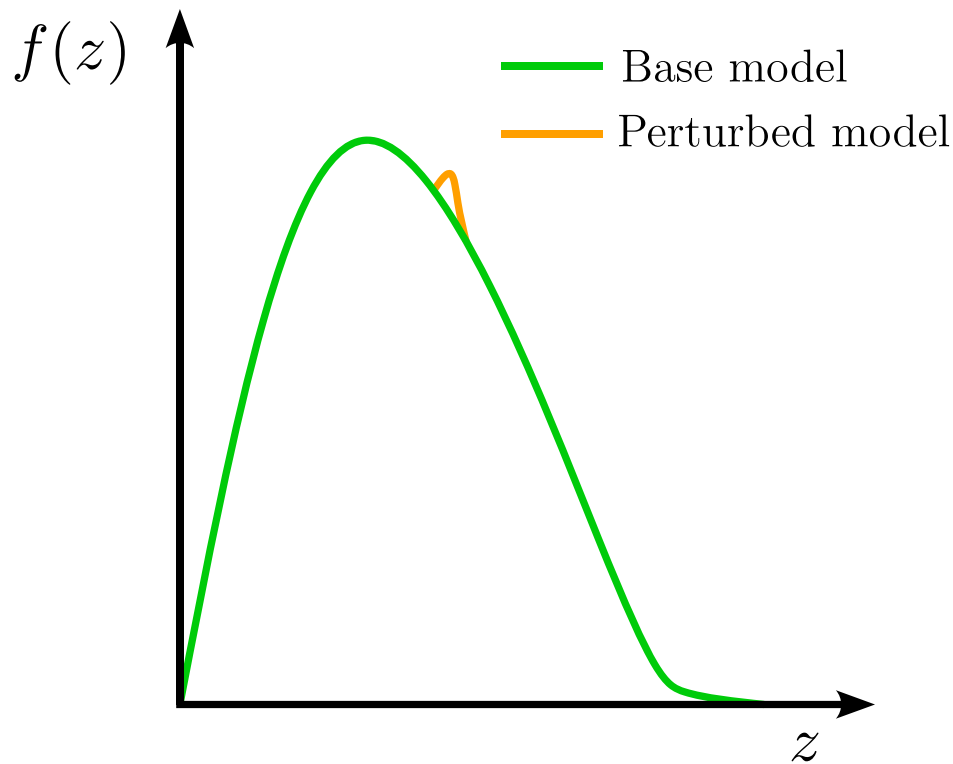
MLHAD efforts: HOMER



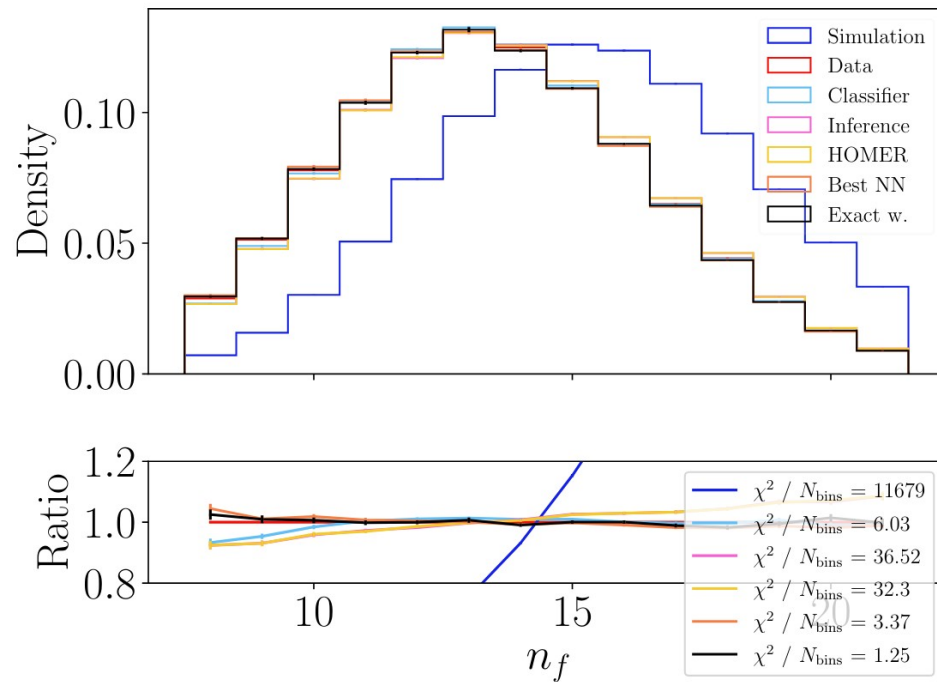
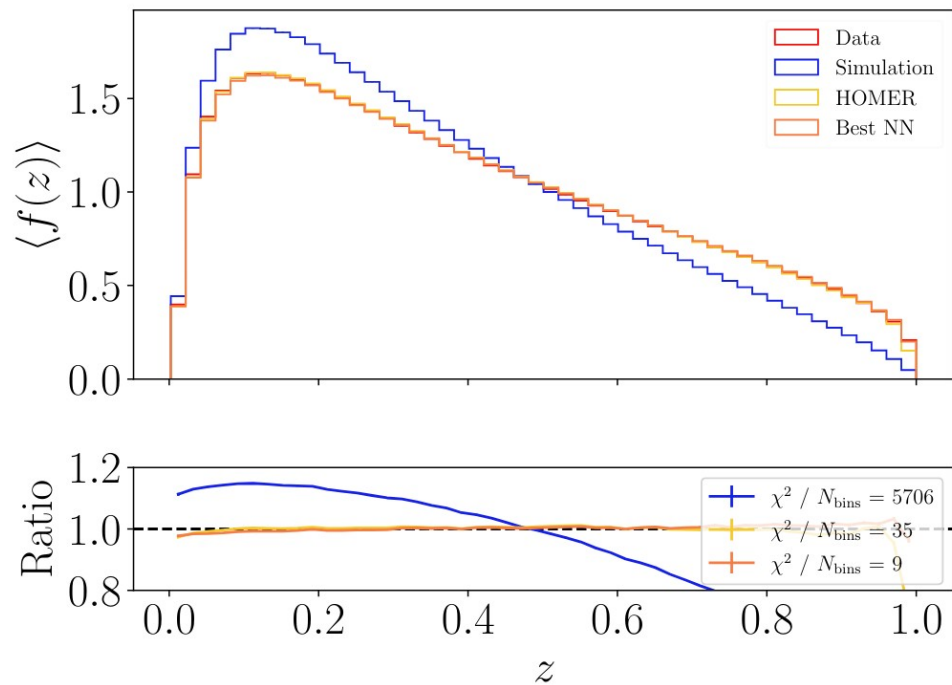
MLHAD efforts: HOMER



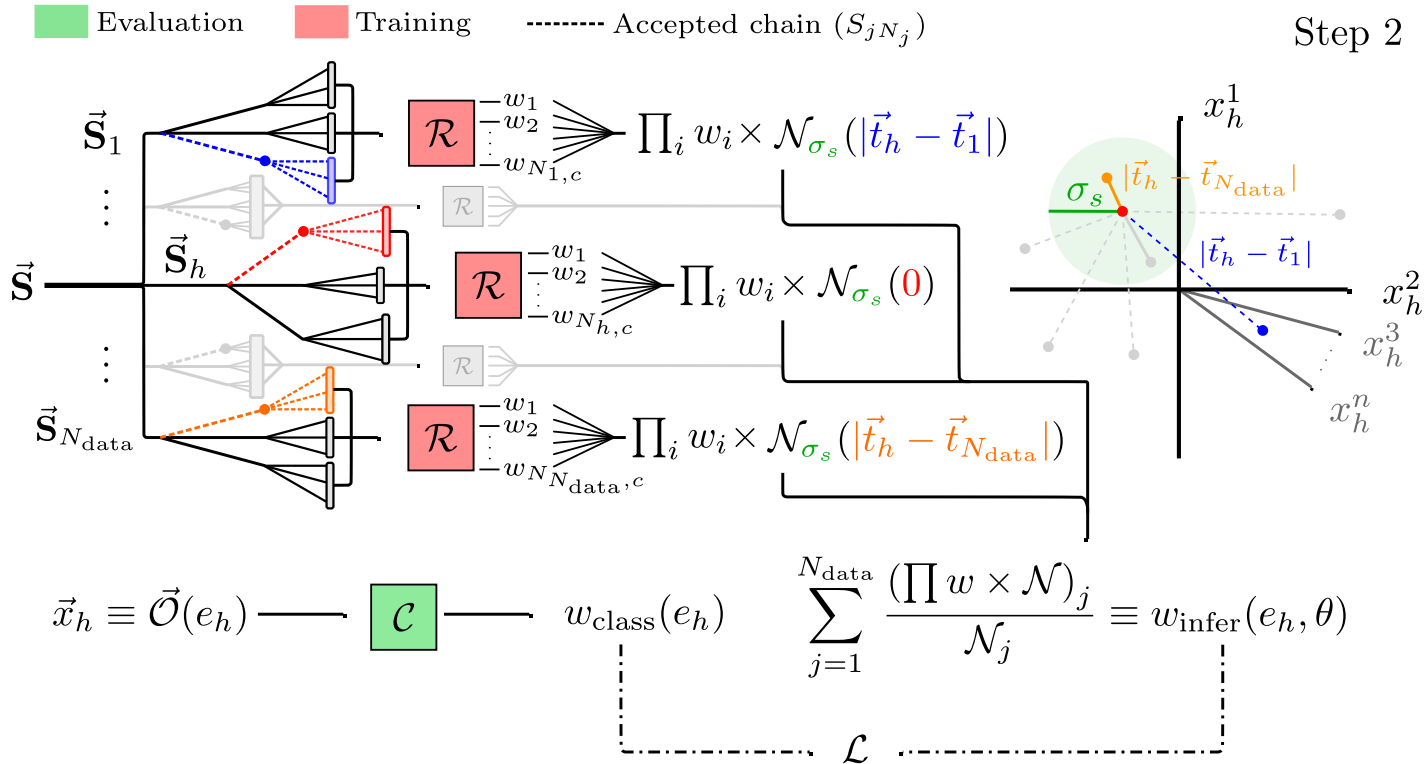
MLHAD efforts: HOMER



HOMER: $q\bar{q}$



HOMER: w/ gluons



Two “solutions” to the inverse problem of hadronization

MAGIC

- “Top-down”
- Relies on the construction of an explicit likelihood (normalizing flow)
- Must be embedded into the simulation pipeline

HOMER

- “Bottom-up”
- Relies on the reconstruction of a classifier weight (derived from the score) from a product of string-break weights
- No modifications to the simulation pipeline required

Conclusions

- **RSA: flexible, scalable, replicable** approach to tuning
 - **ML-based observables, optimizers, infrastructure**
 - **Uncertainty quantification, scaling with parameter dimension**
- **Data-driven fragmentation function:**
 - **HOMER and MAGIC offer as complementary methods**
 - **HOMER + flavor, tackle real data soon?**
 - **Better observables needed**
 - **HOMER x MAGIC**
 - **Better likelihood models? (MAGIC+)**

MLHAD 

<https://uchep.gitlab.io/mlhad-docs/>

Conclusions

- **RSA: flexible, scalable, replicable** approach to tuning
 - **ML-based observables, optimizers, infrastructure**
 - **Uncertainty quantification, scaling with parameter dimension**
- **Data-driven fragmentation function:**
 - **HOMER and MAGIC offer as complementary methods**
 - **HOMER + flavor, tackle real data soon?**
 - **Better observables needed**
 - **HOMER x MAGIC**
 - **Better likelihood models? (MAGIC+)**

MLHAD 

<https://uchep.gitlab.io/mlhad-docs/>

Thank you :)

Back-up

Stringy hadronization: overview

Consider the simplest hadronizing system:

A $q\bar{q}$ pair oriented along the z-axis, with equal and opposite momentum.

Treat this as a semi-classical system with potential

$$V(r) = \kappa r$$

A color flux tube forms between the $q\bar{q}$ pair and in the absence of string breaks the system follows a 'yo-yo' motion

MAGIC: Invertible neural networks (INN)

a.k.a normalizing flow

Invertible Real NVP transformations:

$$z'_1 = z_1 \odot \exp(s_1(z_2)) + t_1(z_2),$$

$$z'_2 = z_2 \odot \exp(s_2(z'_1)) + t_2(z'_1),$$

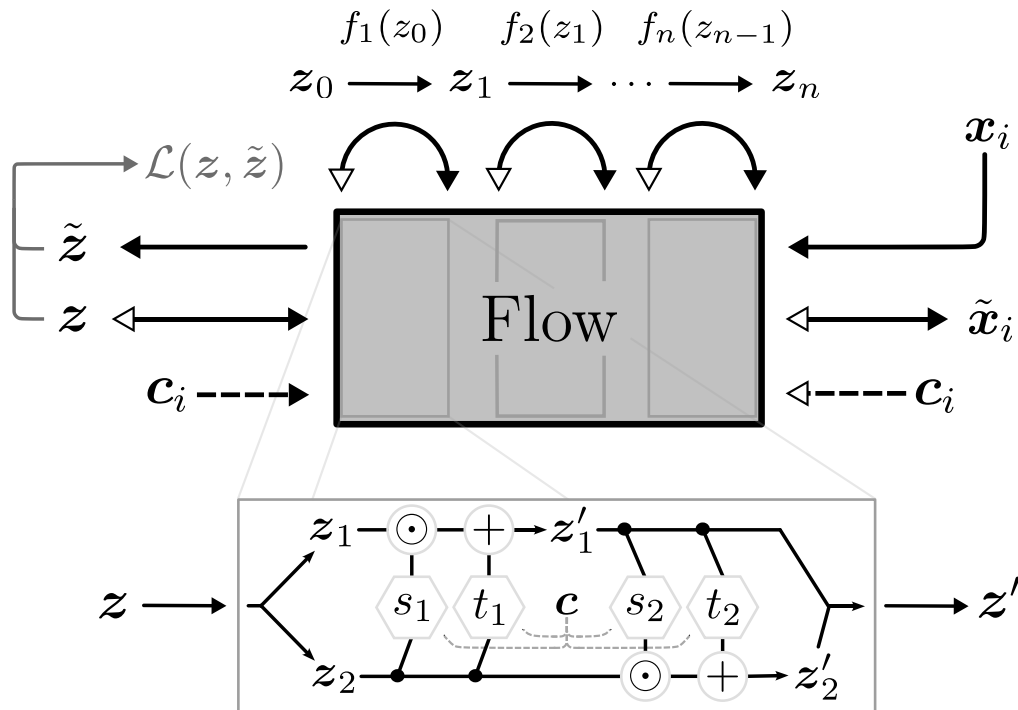
Scale transform

Translation transform

Inverse:

$$z_2 = (z'_2 - t(z'_1)) \odot \exp(-s(z'_1)),$$

$$z_1 = (z'_1 - t(z_2)) \odot \exp(-s(z_2)),$$



Invertible neural networks (INN)

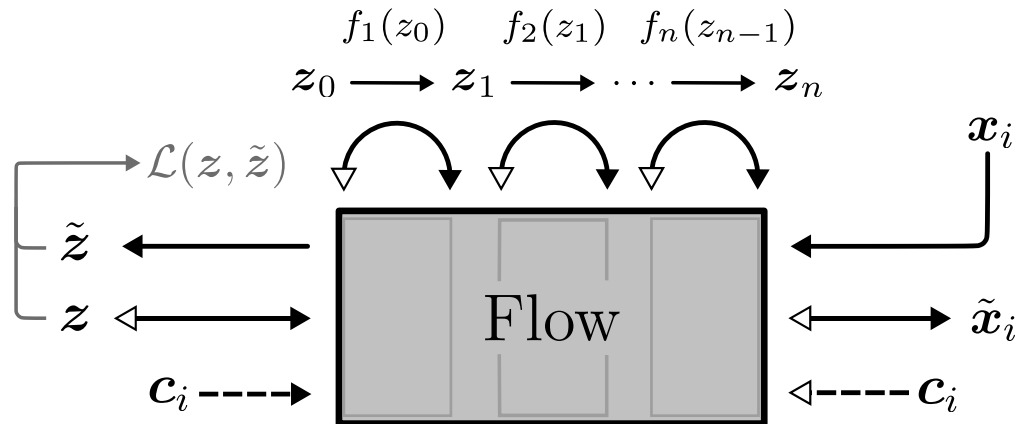
a.k.a normalizing flow

Learn a composition of n independent bijective transformations that relate a probability distribution $p_Z(\mathbf{z})$ on latent space Z to the target distribution $p_X(\mathbf{x})$ on target space X .

The probability distribution for the random variable $\mathbf{x} = f(\mathbf{z})$ is given by

$$p_X^f(\mathbf{x}) = p_Z(\mathbf{z}) |\det J_f(\mathbf{z})|^{-1}$$

$$J_f = \partial f / \partial \mathbf{x}$$



For n iterative transformations:

$$p_X^F(\mathbf{x}) = p_Z(\mathbf{z}_0) \prod_{i=1}^n |\det J_{f_i}(\mathbf{z}_{i-1})|^{-1}$$

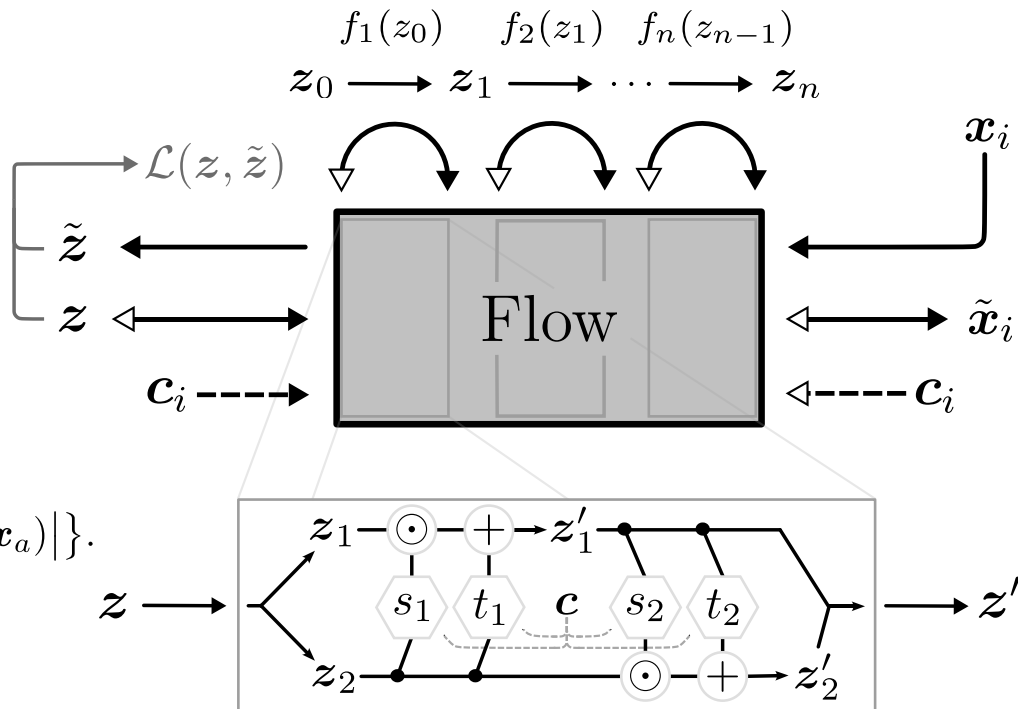
Invertible neural networks (INN)

a.k.a normalizing flow

$$p_X^F(\mathbf{x}) = p_Z(\mathbf{z}_0) \prod_{i=1}^n |\det J_{f_i}(\mathbf{z}_{i-1})|^{-1}$$

Train with the negative log likelihood:

$$\begin{aligned} \mathcal{L}_{\text{NF}} &= - \sum_{a=1}^N \log p_X^F(\mathbf{x}_a; \boldsymbol{\theta}, \mathbf{c}_a) \\ &= \sum_{a=1}^N \{ - \log p_Z(F^{-1}(\mathbf{x}_a; \boldsymbol{\theta}, \mathbf{c}_a)) + \log |\det J_{F^{-1}}(\mathbf{x}_a)| \}. \end{aligned}$$

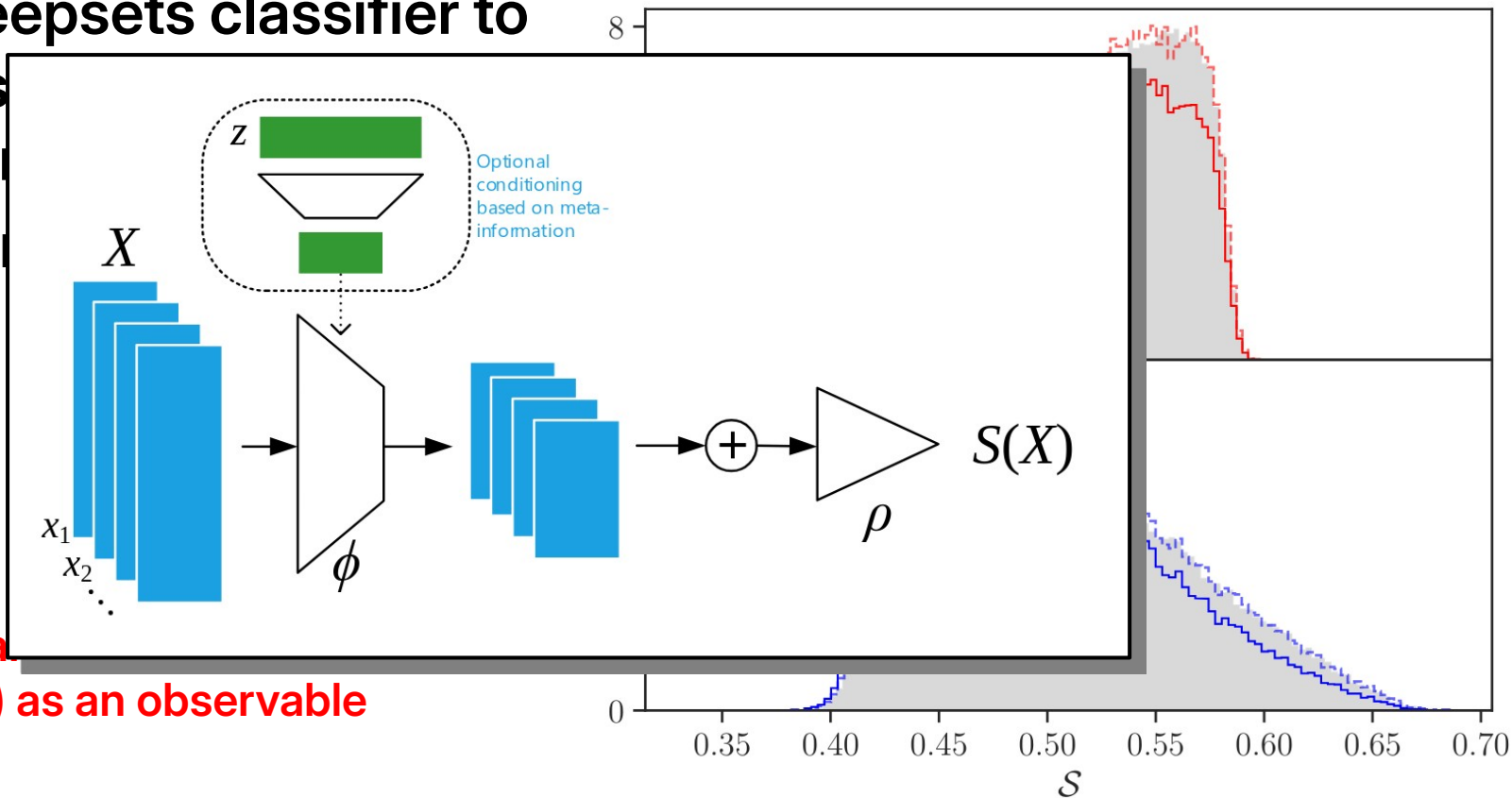


ML-based observables

- Train a deepsets classifier to distinguish experimental conditions

– Takes full input

Use training data (score) as an observable



INN learned mapping

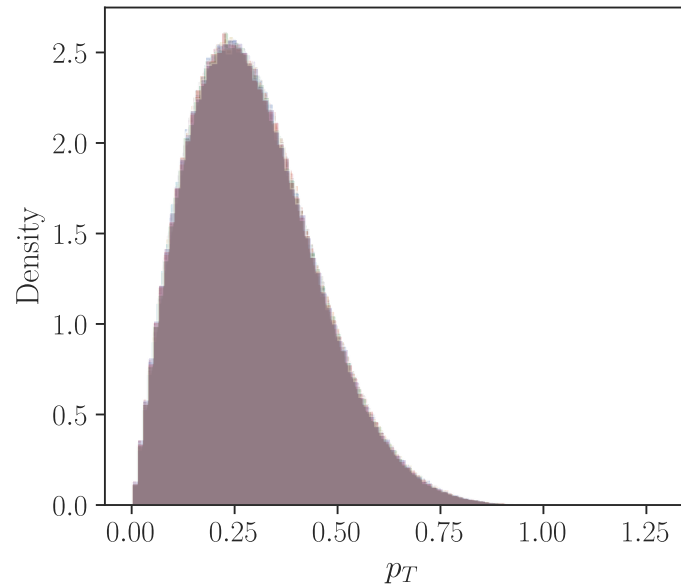
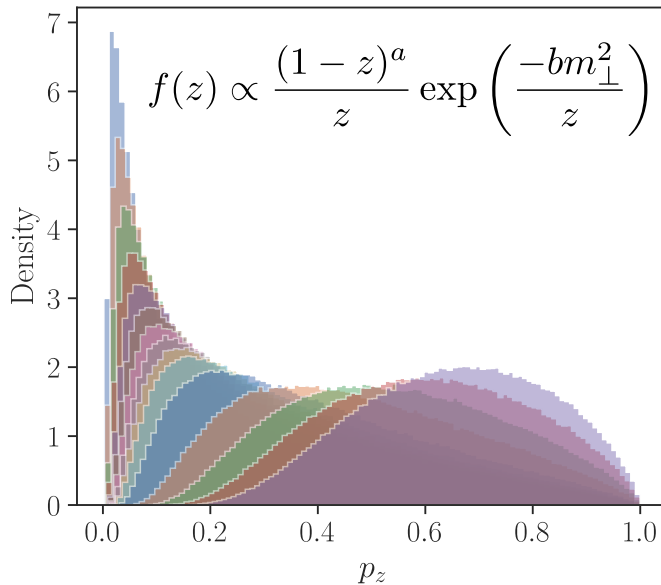
■

INN learned mapping

■

Training data

The implementation of full hadronization event using single emission kinematics requires an independent p_T sampling followed by a p_T -dependent sampling of p_z sampling (due to the dependence of $f(z)$ of transverse mass).



Generate Pythia $q\bar{q} \rightarrow h$ events, first with no p_T kicks at different values of the hadron mass, record p_z . Generate events again, with kicks turned on, record p_T .

MAGIC: $q\bar{q}$

Use hadron multiplicity N as global event-level observable

$$\mathbf{X} = \begin{pmatrix} [\{p_z^{h_1}, p_T^{h_1}\}, \{p_z^{h_2}, p_T^{h_2}\}, \{p_z^{h_3}, p_T^{h_3}\}]_1 \\ [\{p_z^{h_1}, p_T^{h_1}\}, \{p_z^{h_2}, p_T^{h_2}\}, \{p_z^{h_3}, p_T^{h_3}\}, \{p_z^{h_4}, p_T^{h_4}\}, \{p_z^{h_5}, p_T^{h_5}\}]_2 \\ \vdots \\ [\{p_z^{h_1}, p_T^{h_1}\}, \{p_z^{h_2}, p_T^{h_2}\}]_n \end{pmatrix}, \quad \mathbf{Y}^{\text{sim}} = \begin{pmatrix} N_1 = 3 \\ N_2 = 5 \\ \vdots \\ N_n = 2 \end{pmatrix}, \quad \mathbf{Y}^{\text{exp}} = \begin{pmatrix} N_1 = 7 \\ N_2 = 3 \\ \vdots \\ N_n = 2 \end{pmatrix}$$

Event weights: $w = \begin{pmatrix} \prod_{i=1}^{N_1} w_i \\ \prod_{j=1}^{N_2} w_j \\ \vdots \\ \prod_{k=1}^{N_n} w_k \end{pmatrix}$ where $w_i = \frac{p_X^{F'}(p_z^{h_i}, p_T^{h_i})}{p_X^F(p_z^{h_i}, p_T^{h_i})}$,

Loss (Earth mover's distance): $\mathcal{L}_{\text{EMD}}(\mathbf{Y}^{\text{sim}}, \mathbf{w}, \mathbf{Y}^{\text{exp}})$

SHAP values

