Learning the latent structure of collider events

hep-ph/2005.12319
hep-ph/1904.04200

Barry M. Dillon
barry.dillon@ijs.si
GitHub.com/barrydillon89

Jozef Stefan Institute

with Darius A. Faroughy, Jernej F. Kamenik, Manuel Szewc

Talk at the IML Machine Learning Working Group
June 2020

Institut
"Jožef Stefan"
Ljubljana, Slovenija

Outline

- Unsupervised algorithm for learning patterns in collider event data
  + classification based on learned patterns

Related work

VAEs	Cerri et al (hep-ex/1811.10276)
CWoLa	Metodiev et al (hep-ph/1708.02949), Collins et al (hep-ph/1902.02634), Amram et al (hep-ph/2002.12376)
AEs	Farina et al (hep-ph/1808.08992), Roy et al (hep-ph/1903.02032), Blance et al (hep-ph/1905.10384)
DeMix	Metodiev et al (hep-ph/1809.01140), Komiske et al (hep-ph/1809.01140), Alzarez et al (hep-ph/1911.09699)
MAFs	Nachman et al (hep-ph/2001.04990)

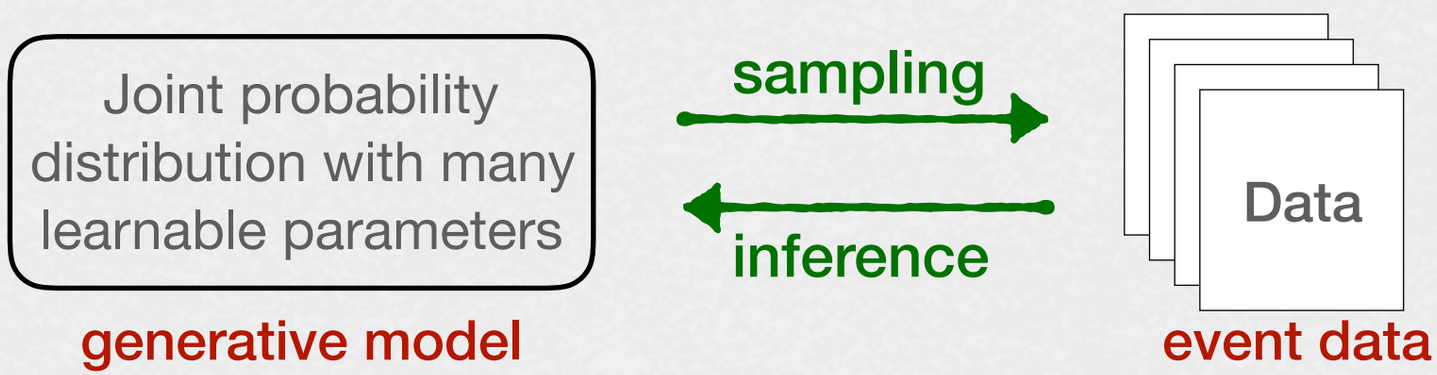
Outline

- Unsupervised algorithm for learning patterns in collider event data
  + classification based on learned patterns

Related work

VAEs	Cerri et al (hep-ex/1811.10276)
CWoLa	Metodiev et al (hep-ph/1708.02949), Collins et al (hep-ph/1902.02634), Amram et al (hep-ph/2002.12376)
AEs	Farina et al (hep-ph/1808.08992), Roy et al (hep-ph/1903.02032), Blance et al (hep-ph/1905.10384)
DeMix	Metodiev et al (hep-ph/1809.01140), Komiske et al (hep-ph/1809.01140), Alzarez et al (hep-ph/1911.09699)
MAFs	Nachman et al (hep-ph/2001.04990)

- Bayesian probabilistic (i.e. generative) modelling



Stochastic variational inference algorithm to learn the parameters

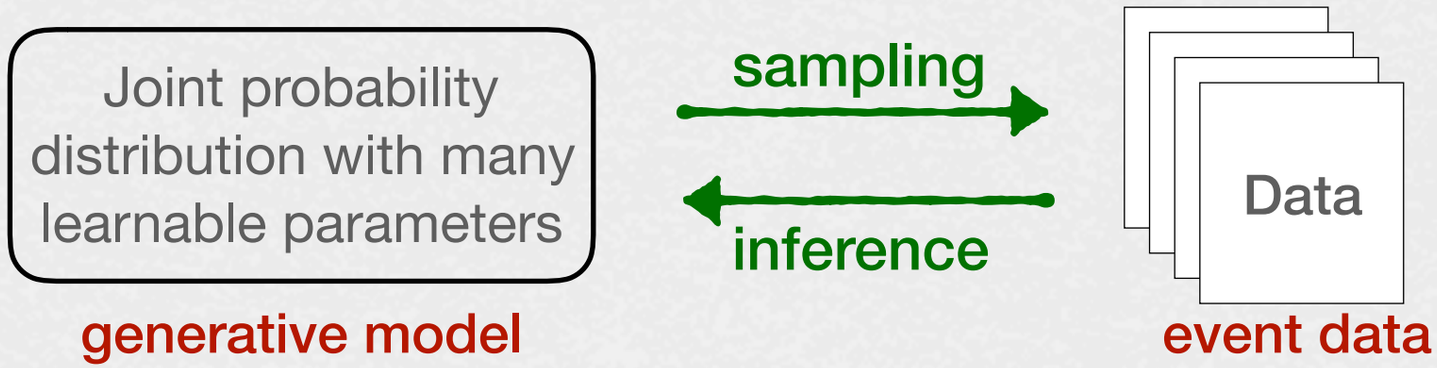
Outline

- Unsupervised algorithm for learning patterns in collider event data
  + classification based on learned patterns

Related work

VAEs	Cerri et al (hep-ex/1811.10276)
CWoLa	Metodiev et al (hep-ph/1708.02949), Collins et al (hep-ph/1902.02634), Amram et al (hep-ph/2002.12376)
AEs	Farina et al (hep-ph/1808.08992), Roy et al (hep-ph/1903.02032), Blance et al (hep-ph/1905.10384)
DeMix	Metodiev et al (hep-ph/1809.01140), Komiske et al (hep-ph/1809.01140), Alzarez et al (hep-ph/1911.09699)
MAFs	Nachman et al (hep-ph/2001.04990)

- Bayesian probabilistic (i.e. generative) modelling



Stochastic variational inference algorithm to learn the parameters

- Example: learning patterns in di-jet data

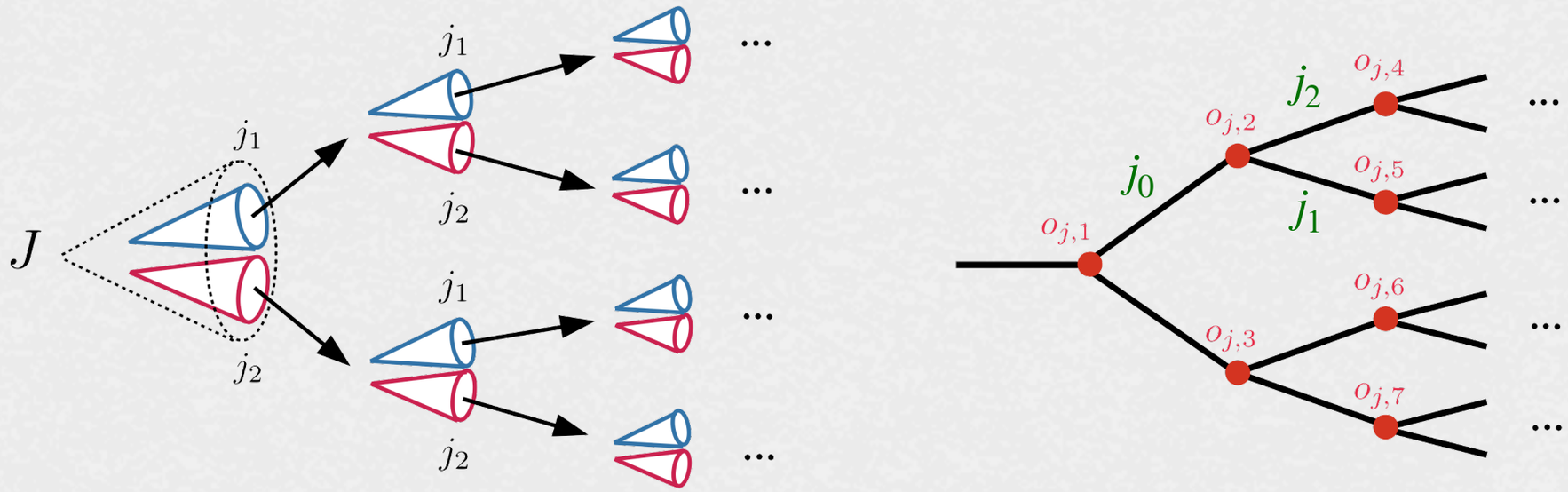
The event data

- Events are represented as a list of binned measurements

The event data

- Events are represented as a list of binned measurements

- Example: jet substructure, where for each splitting $o_{j,i} = [m_0, m_1/m_0]_{j,i}$

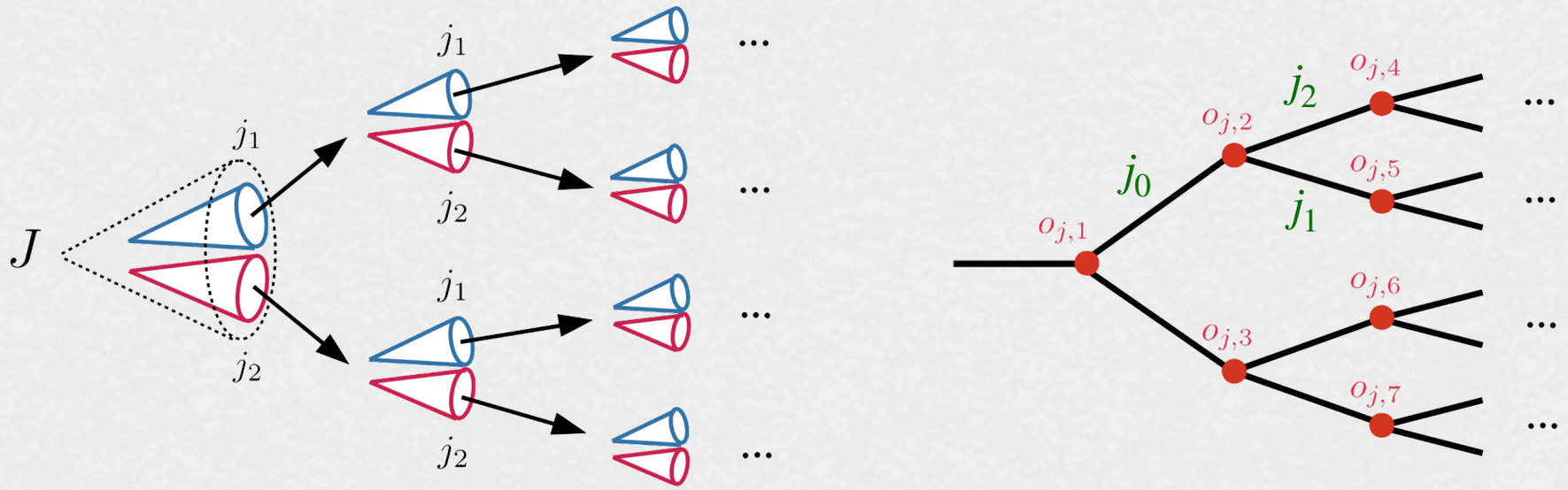


Each $o_{j,i}$ is then a bin in the 2D space spanned by m_0 and m_1/m_0

The event data

- Events are represented as a list of binned measurements

- Example: jet substructure, where for each splitting $o_{j,i} = [m_0, m_1/m_0]_{j,i}$



Each $o_{j,i}$ is then a bin in the 2D space spanned by m_0 and m_1/m_0

- Di-jet events: jet labels (J_1 or J_2, $m_{J_1} > m_{J_2}$) specify which jet a splitting belongs to

  an event: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \ldots\}$

The probabilistic model

- We want to quantify the probability that the events were generated

The probabilistic model

- We want to quantify the probability that the events were generated

- Generative assumptions:

  - the $o_{j,i}$ are sampled from distributions representing physical processes

    We call these distributions themes

The probabilistic model

- We want to quantify the probability that the events were generated

- Generative assumptions:

  - the $o_{j,i}$ are sampled from distributions representing physical processes

    We call these distributions themes

  - a single event is generated by a mixture of physical processes

    The prevalence of different themes in the event sample is determined by the Dirichlet prior:

$$p(\omega \mid \alpha) = \text{Dirichlet}(\alpha)$$

Latent Dirichlet Allocation (LDA)
Blei et al (2003)

The probabilistic model

Challenge: small S/B

The **Dirichlet prior** allows the algorithm to learn rare signals.
Asymmetries in the distribution can produce rare events.

- the $o_{j,i}$ are sampled from distributions representing physical processes

  We call these distributions themes

- a single event is generated by a mixture of physical processes

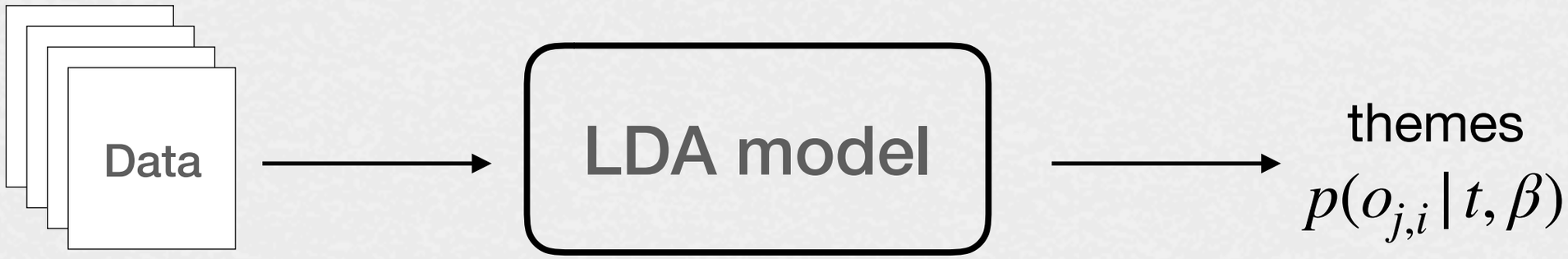
  The prevalence of different themes in the event sample is determined by the Dirichlet prior:

$$p(\omega \mid \alpha) = \text{Dirichlet}(\alpha)$$

Latent Dirichlet Allocation (LDA)
Blei et al (2003)

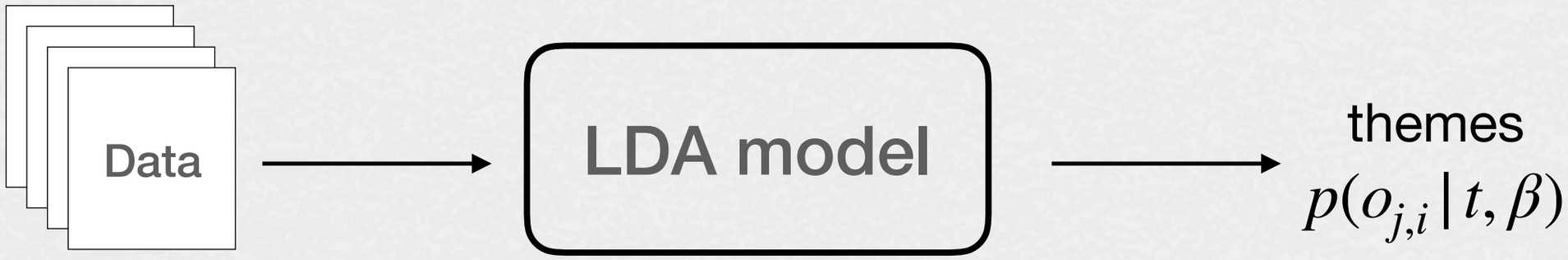
Inference - how we learn

- We want to estimate, or ‘learn’, the parameters of the model



Inference - how we learn

- We want to estimate, or ‘learn’, the parameters of the model



- Stochastic variational inference: estimating the posterior

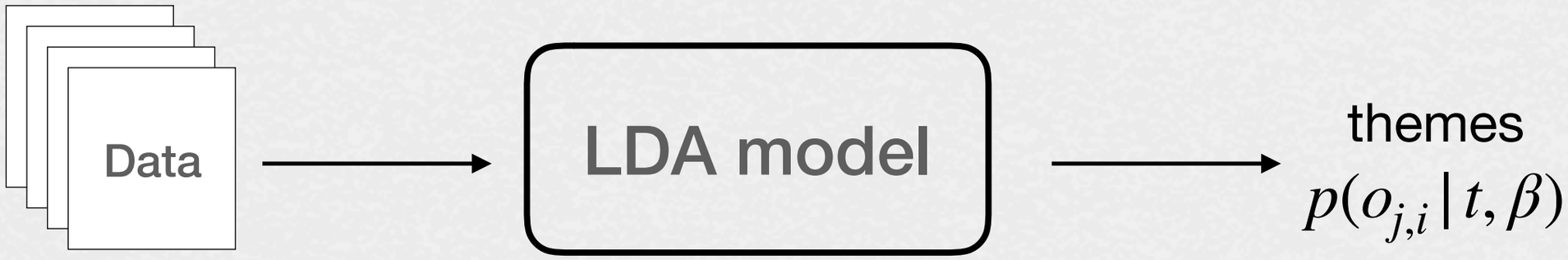
Hoffman et al (2010), Gensim: Rehurek et al (2010)

$$p(\beta, t, \omega \mid e) \simeq q(\beta)q(t)q(\omega)$$

LDA is special: belongs to the Conjugate Exponential Family of models

Inference - how we learn

- We want to estimate, or ‘learn’, the parameters of the model



- Stochastic variational inference: estimating the posterior

Hoffman et al (2010), Gensim: Rehurek et al (2010)

$$p(\beta, t, \omega \mid e) \simeq q(\beta)q(t)q(\omega)$$

LDA is special: belongs to the Conjugate Exponential Family of models

$\Rightarrow$ closed form updates for $q(\beta)$, $q(t)$, and $q(\omega)$ that minimise the divergence between the the true and approximate posterior.

$\Rightarrow$ we can learn the posterior, and therefore the themes (β), iteratively

What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\},\quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

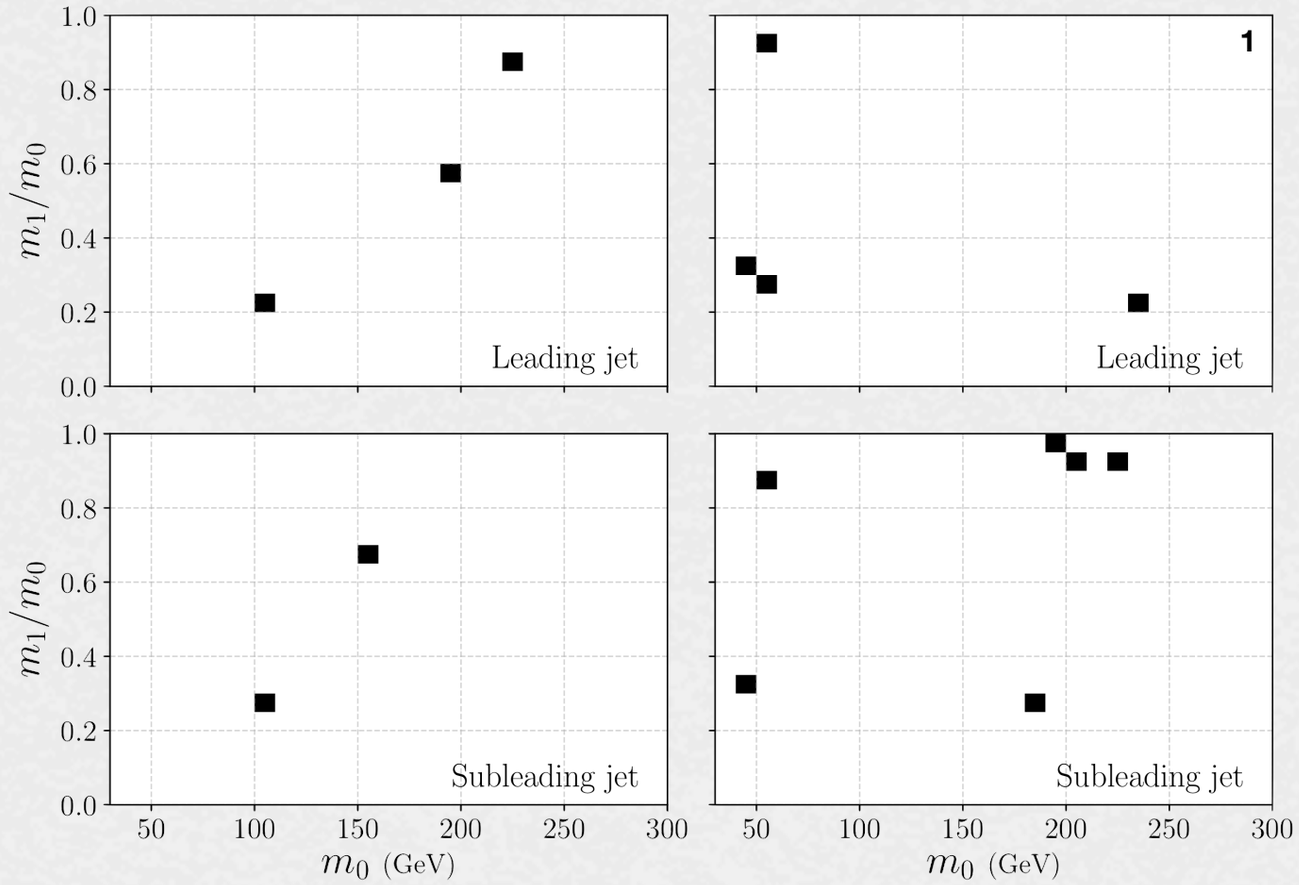
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\}, \quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Example events:



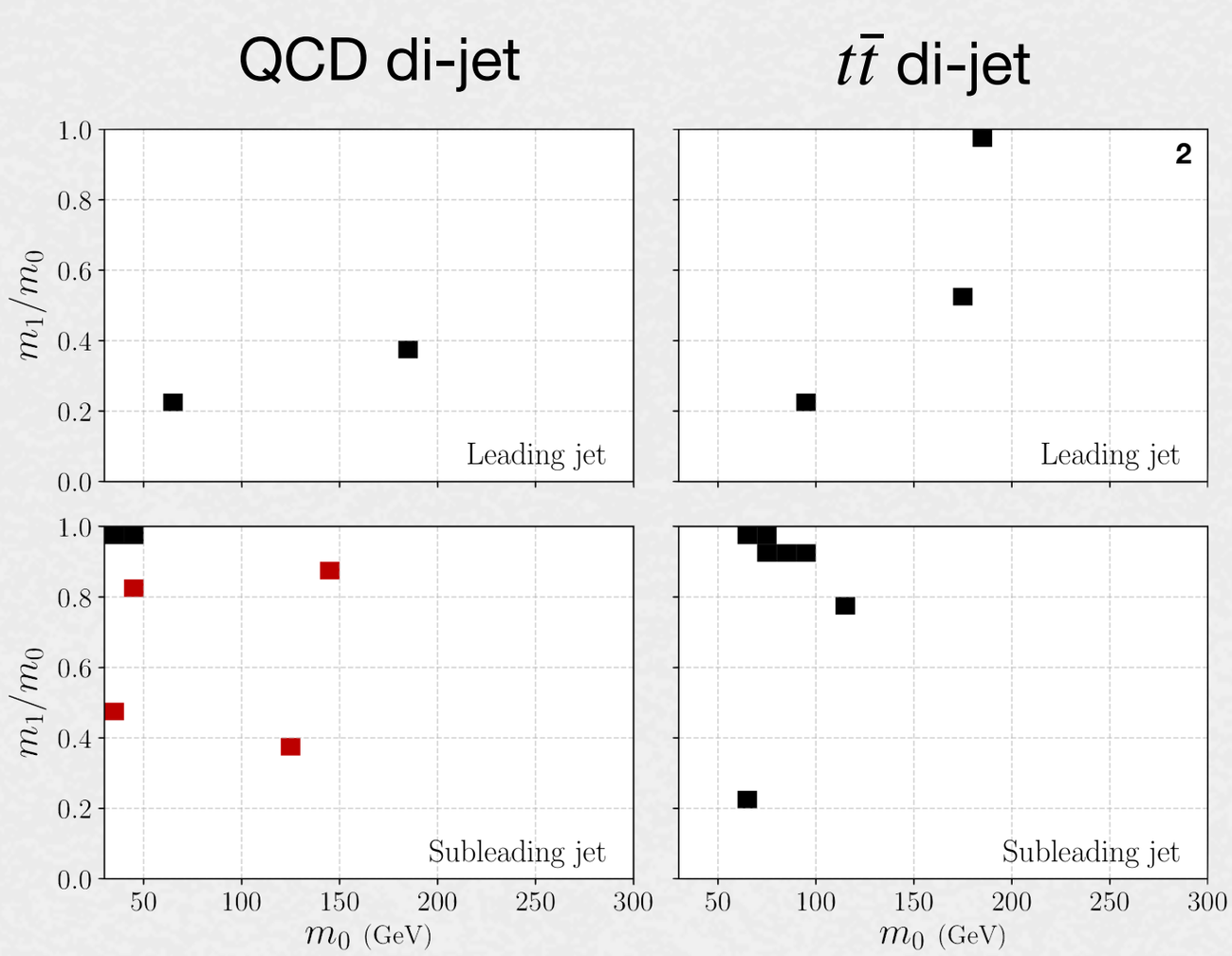
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\}, \quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Example events:



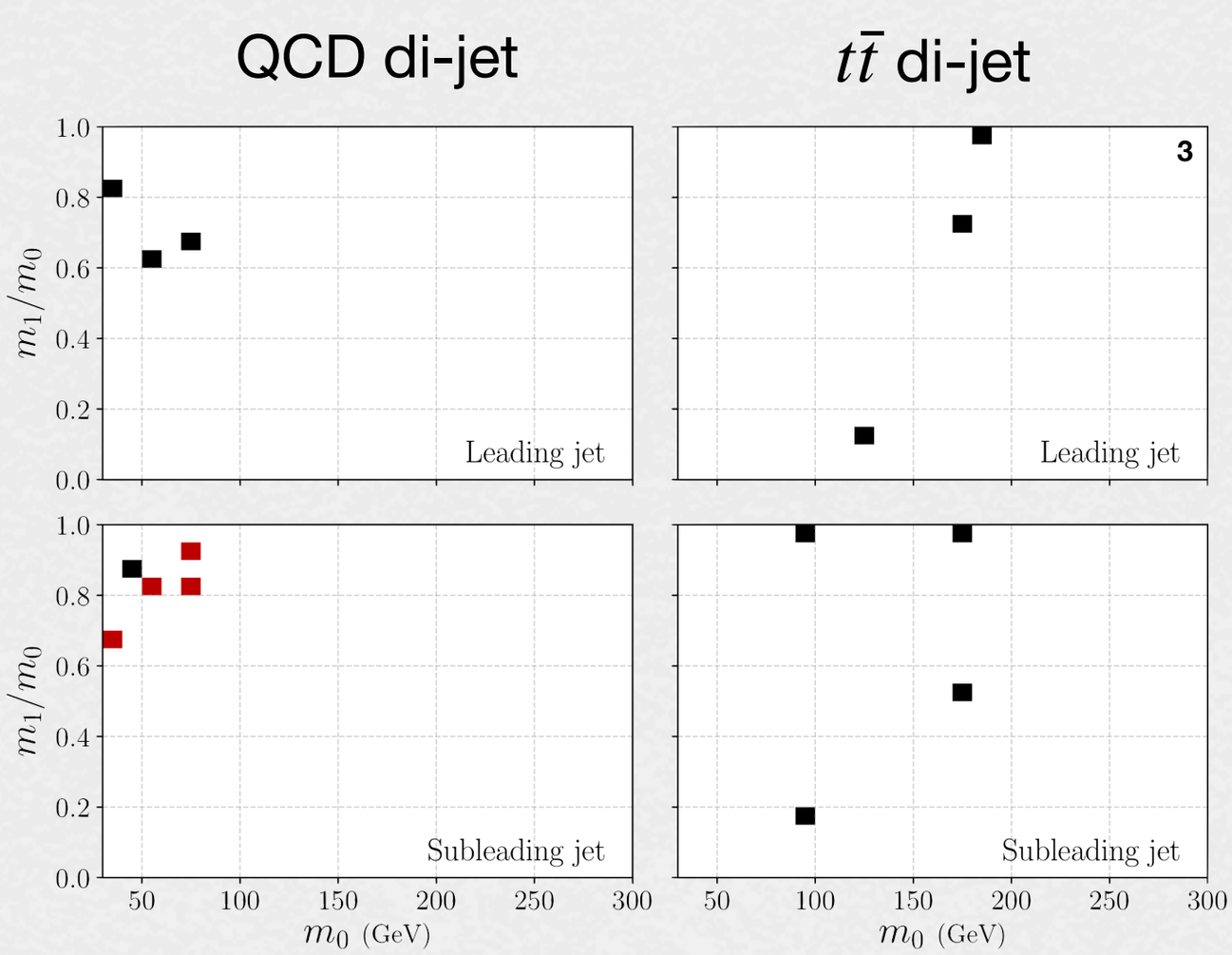
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\},\quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Example events:



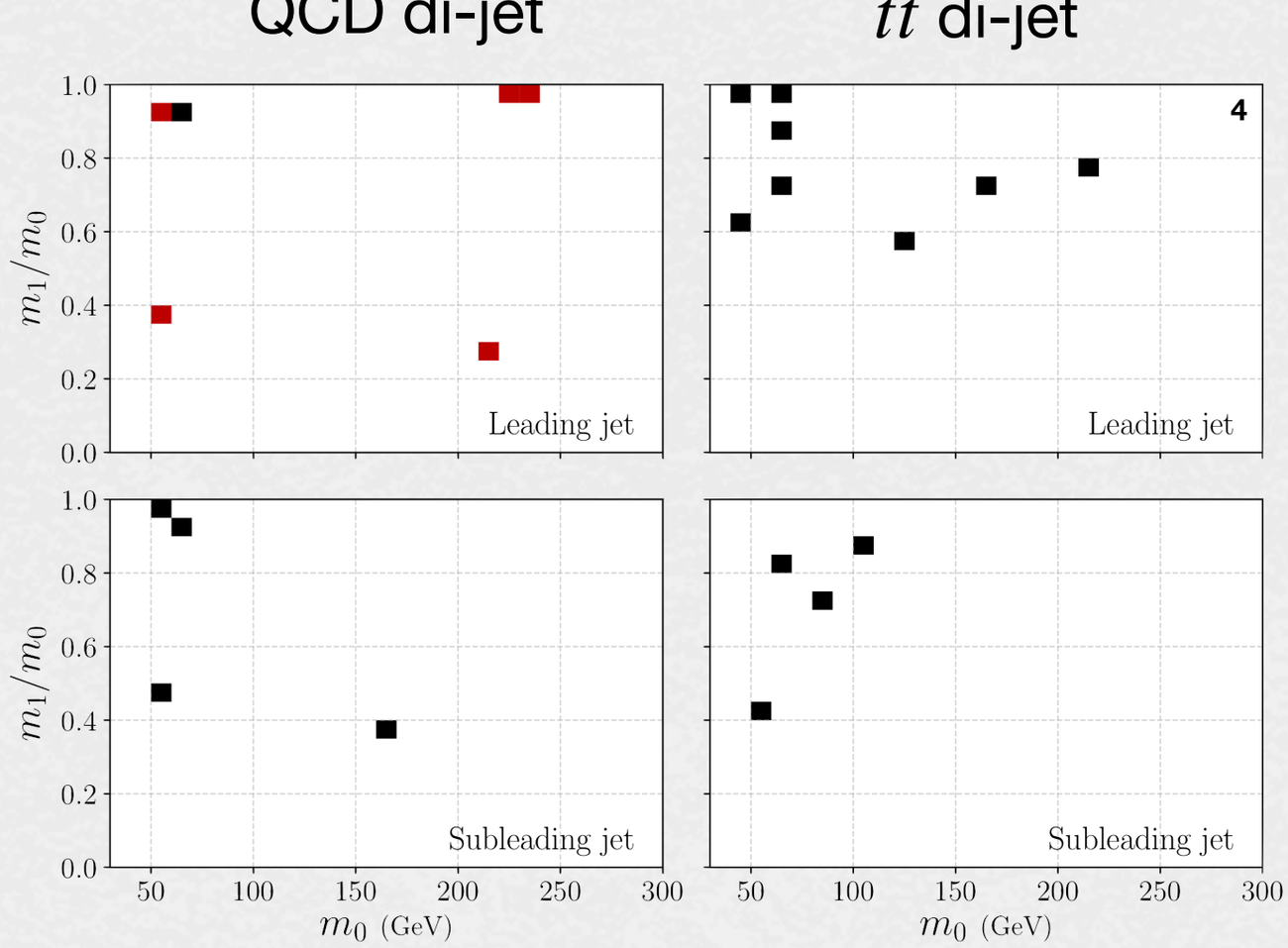
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\}, \quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Example events:



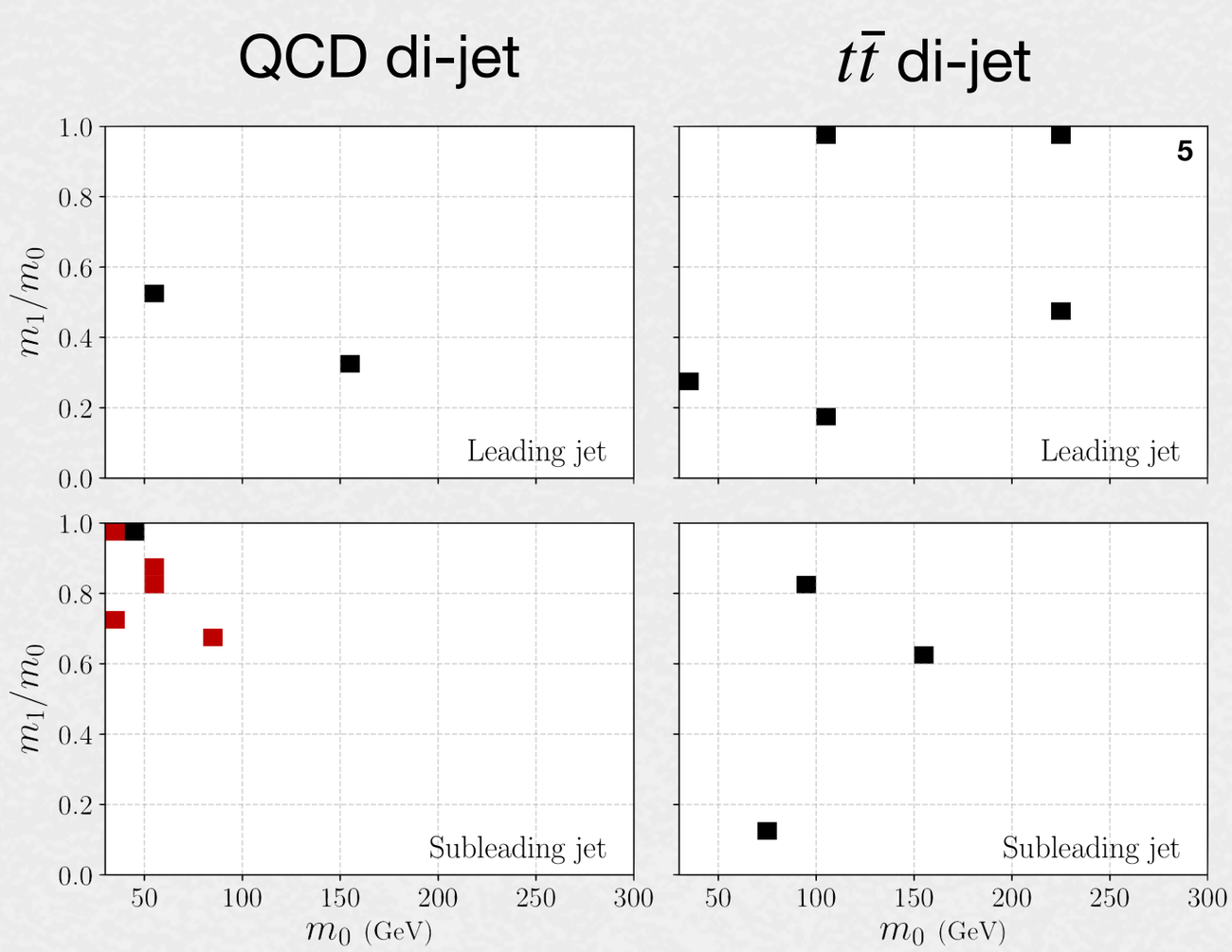
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\}, \quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Example events:

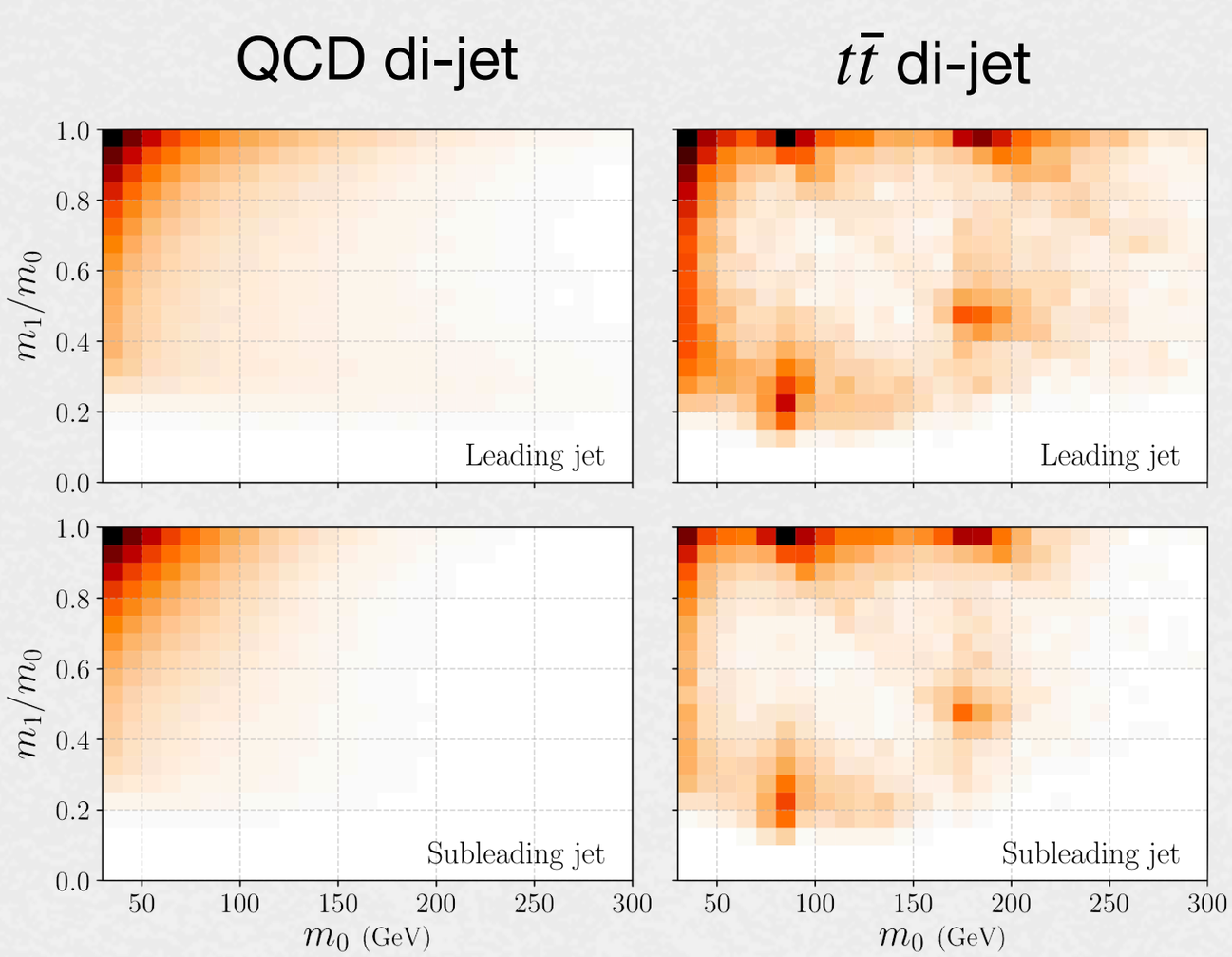


What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM
  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\},\quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

- Truth distributions:

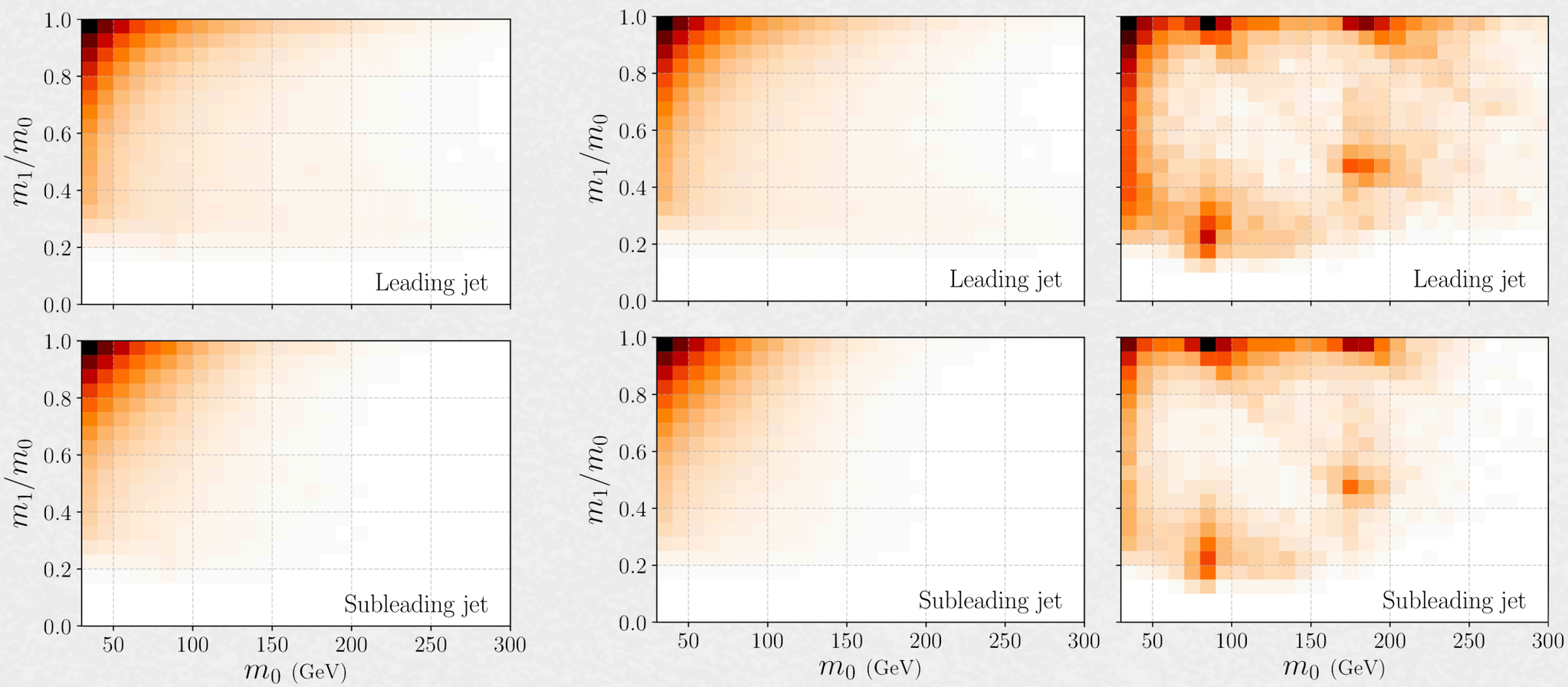


What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\}, \quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$



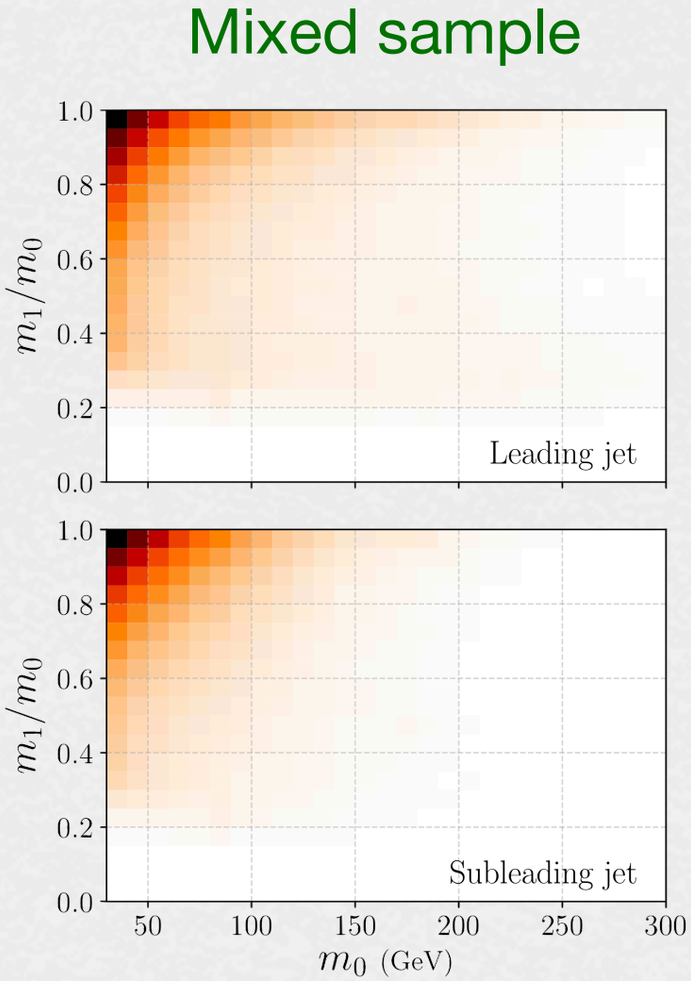
What does the algorithm learn?

- Example: boosted $t\bar{t} \to jj$ production in the SM

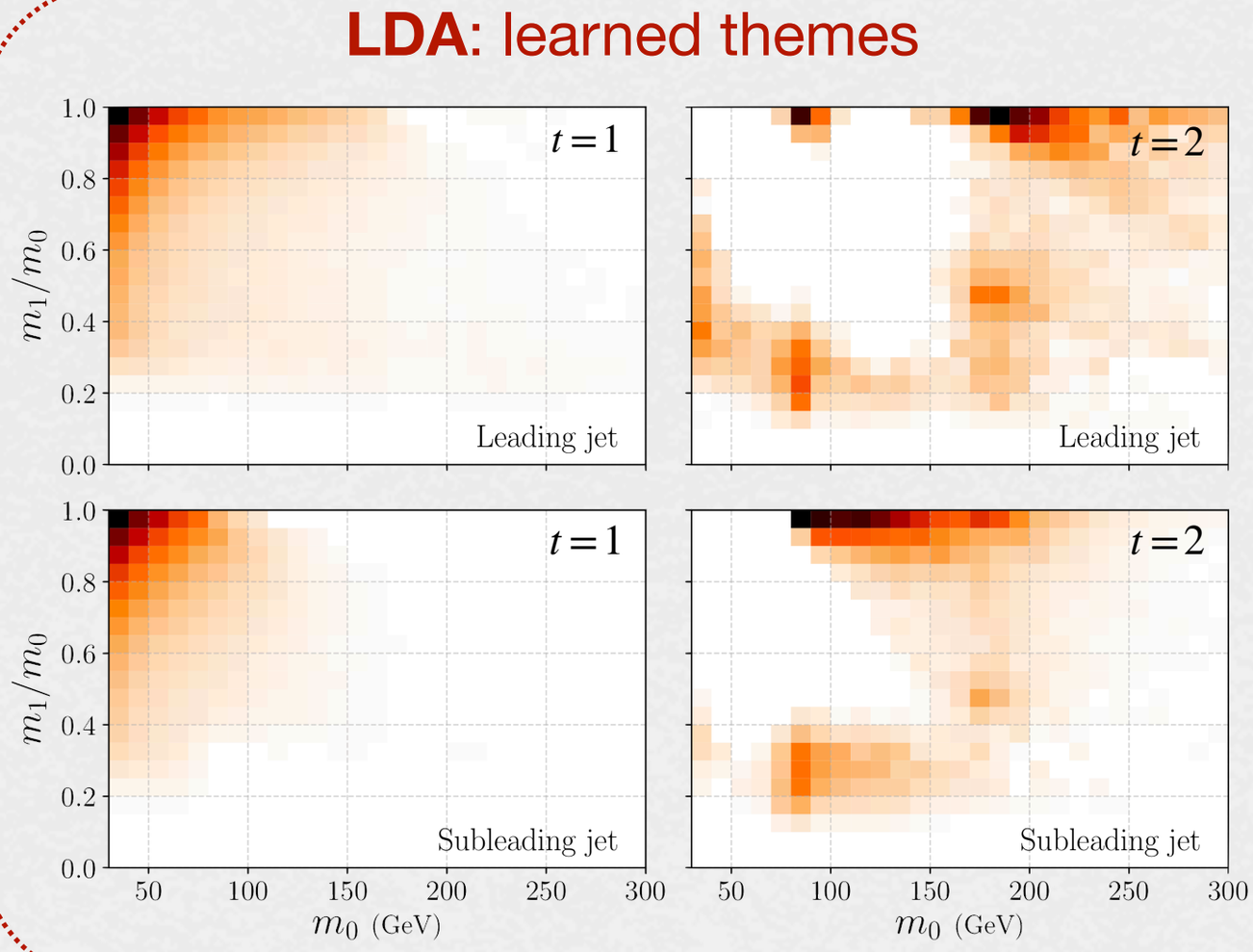
  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\},\quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

Mixed sample



LDA: learned themes



What does the algorithm learn?

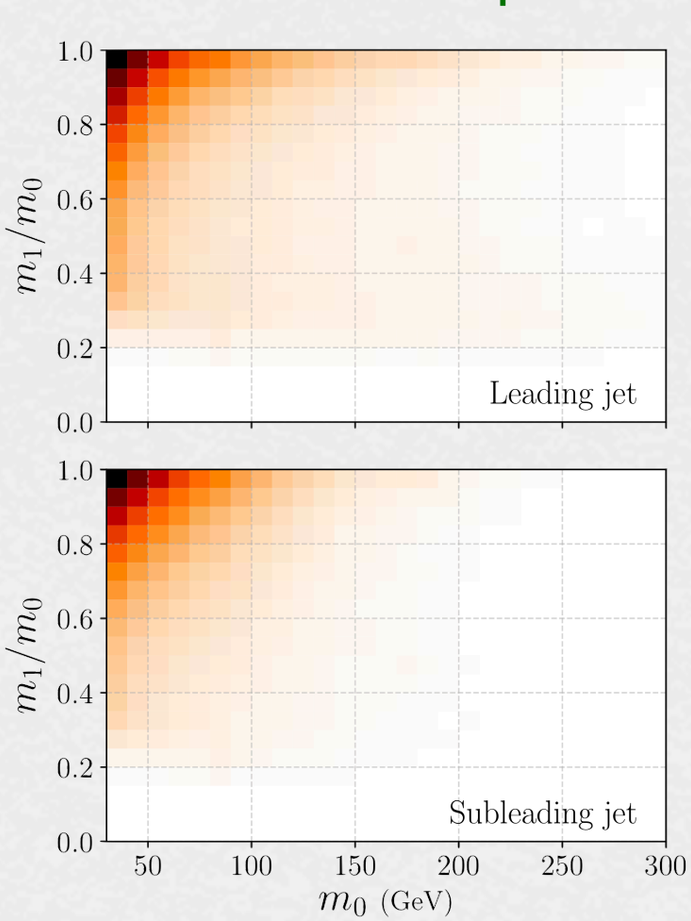
Why

- Example: boosted $t\bar{t} \to jj$ production in the SM

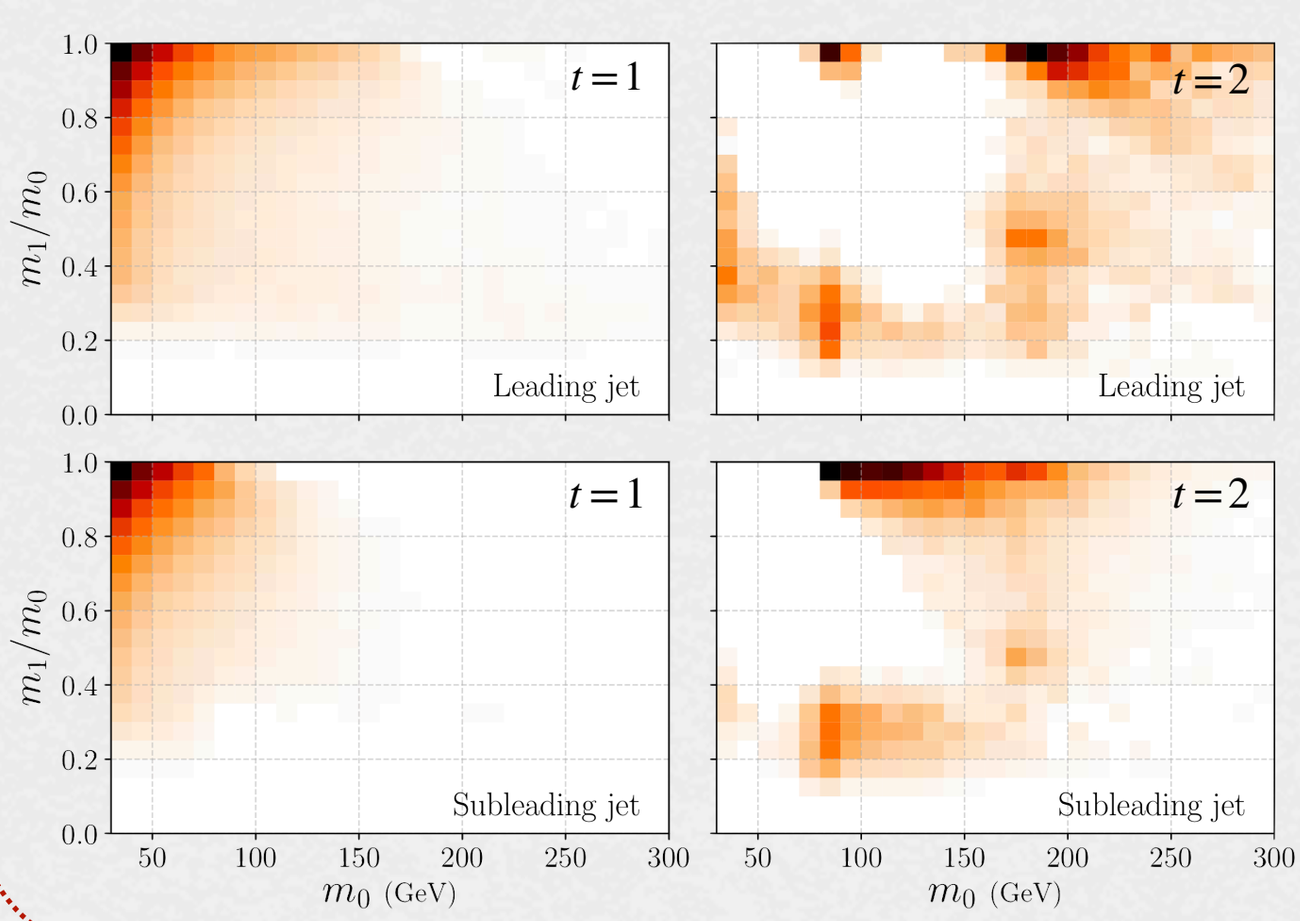
  $S/B = 5\,\%\,,\ N_e = 10^5$

- Data representation: $e_j = \{o_{j,1}, o_{j,2}, o_{j,3}, \dots\},\quad o_{j,i} = [J, m_0, m_1/m_0]_{j,i}$

Mixed sample



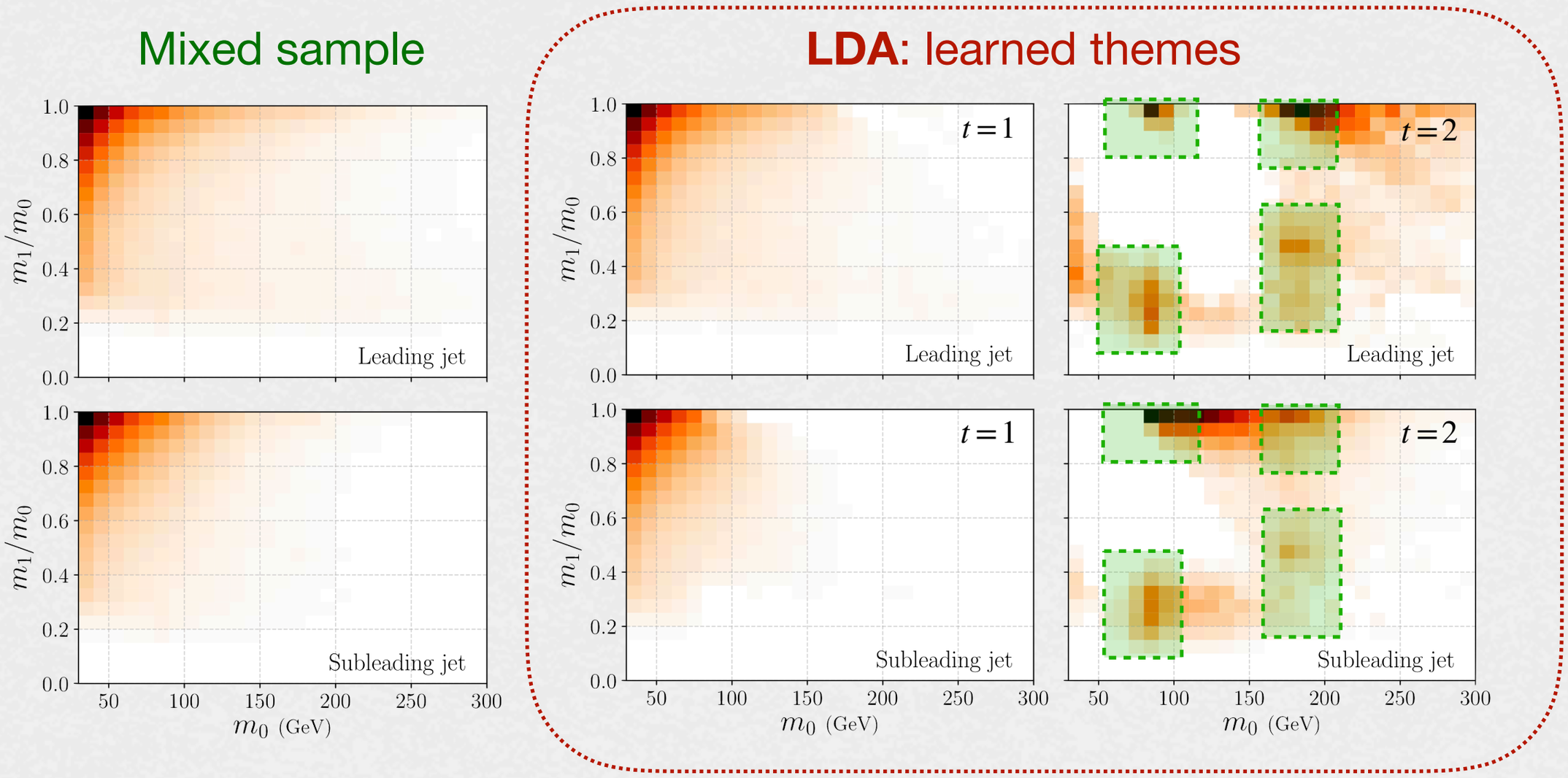
LDA: learned themes



What does the algorithm learn?

Why

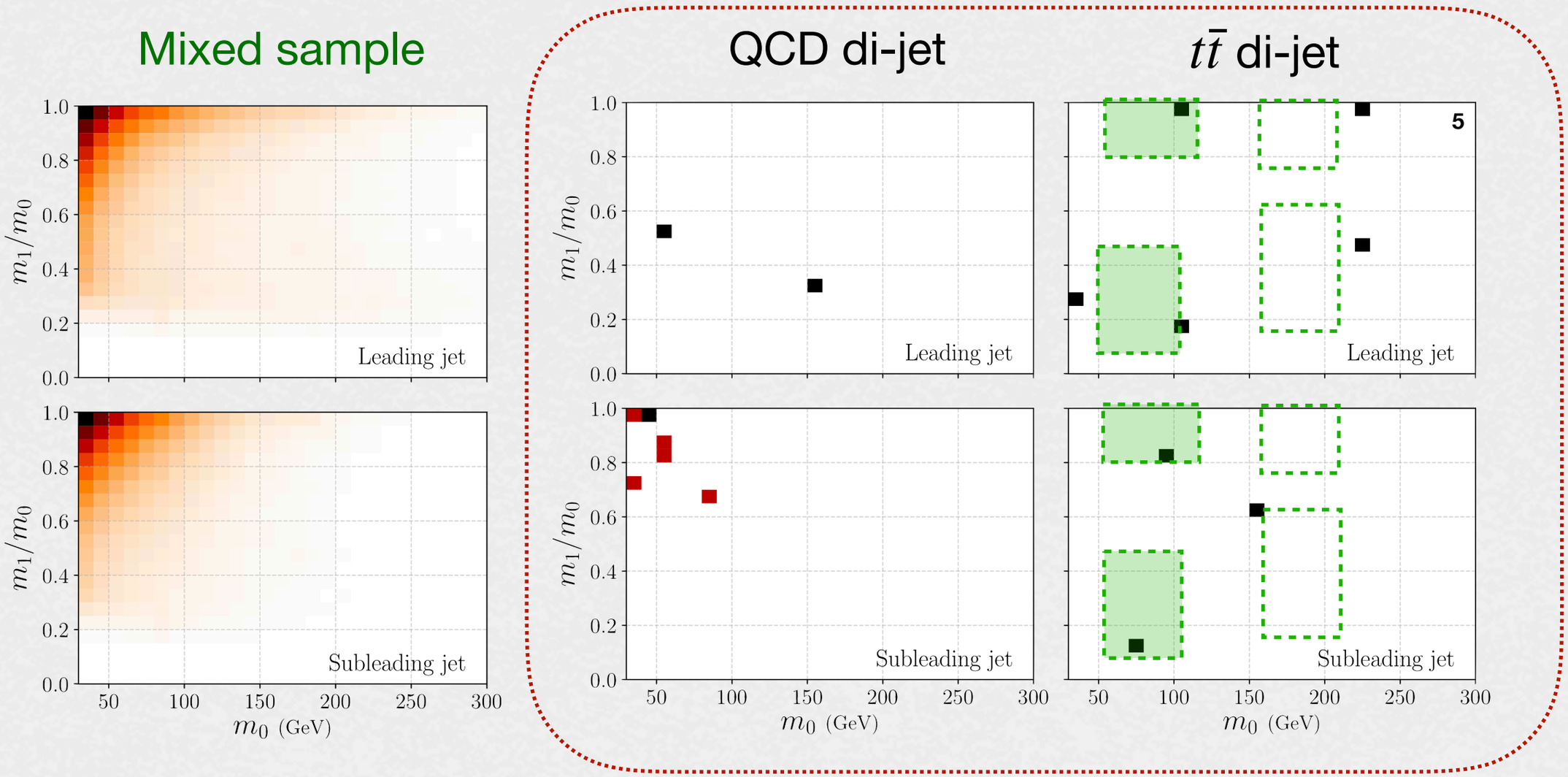
- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



What does the algorithm learn?

Why

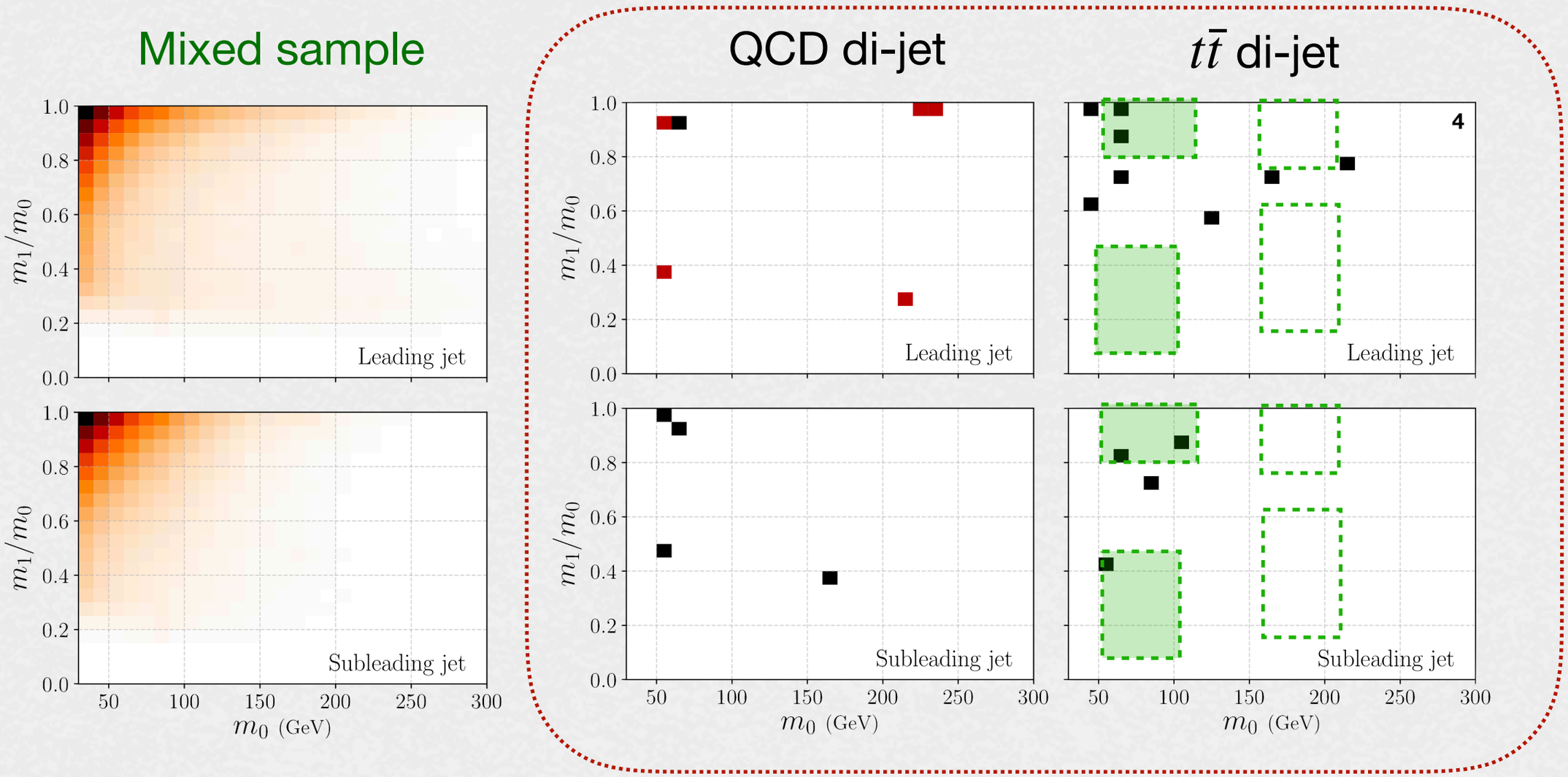
- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



What does the algorithm learn?

Why

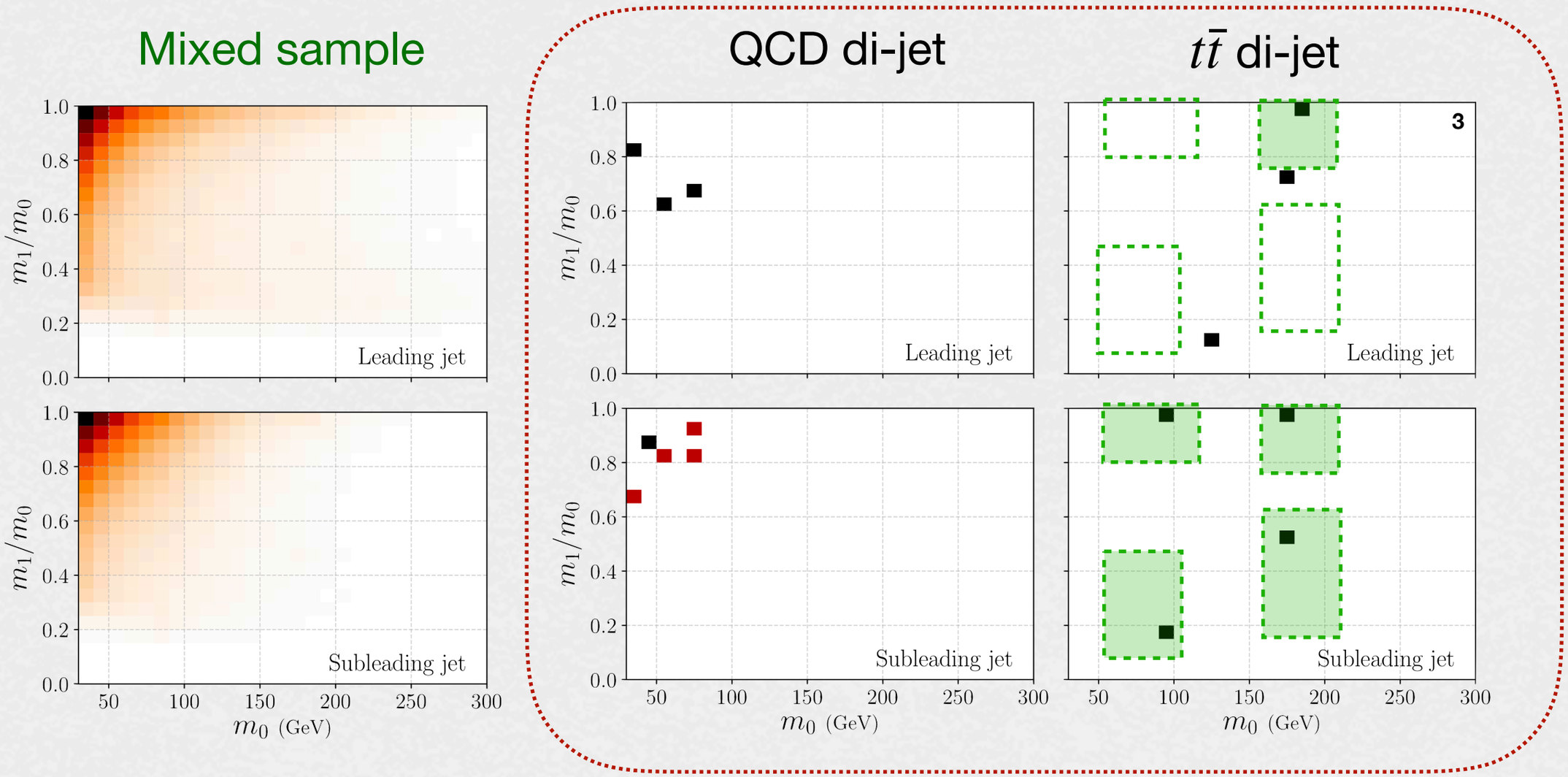
- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



What does the algorithm learn?

Why

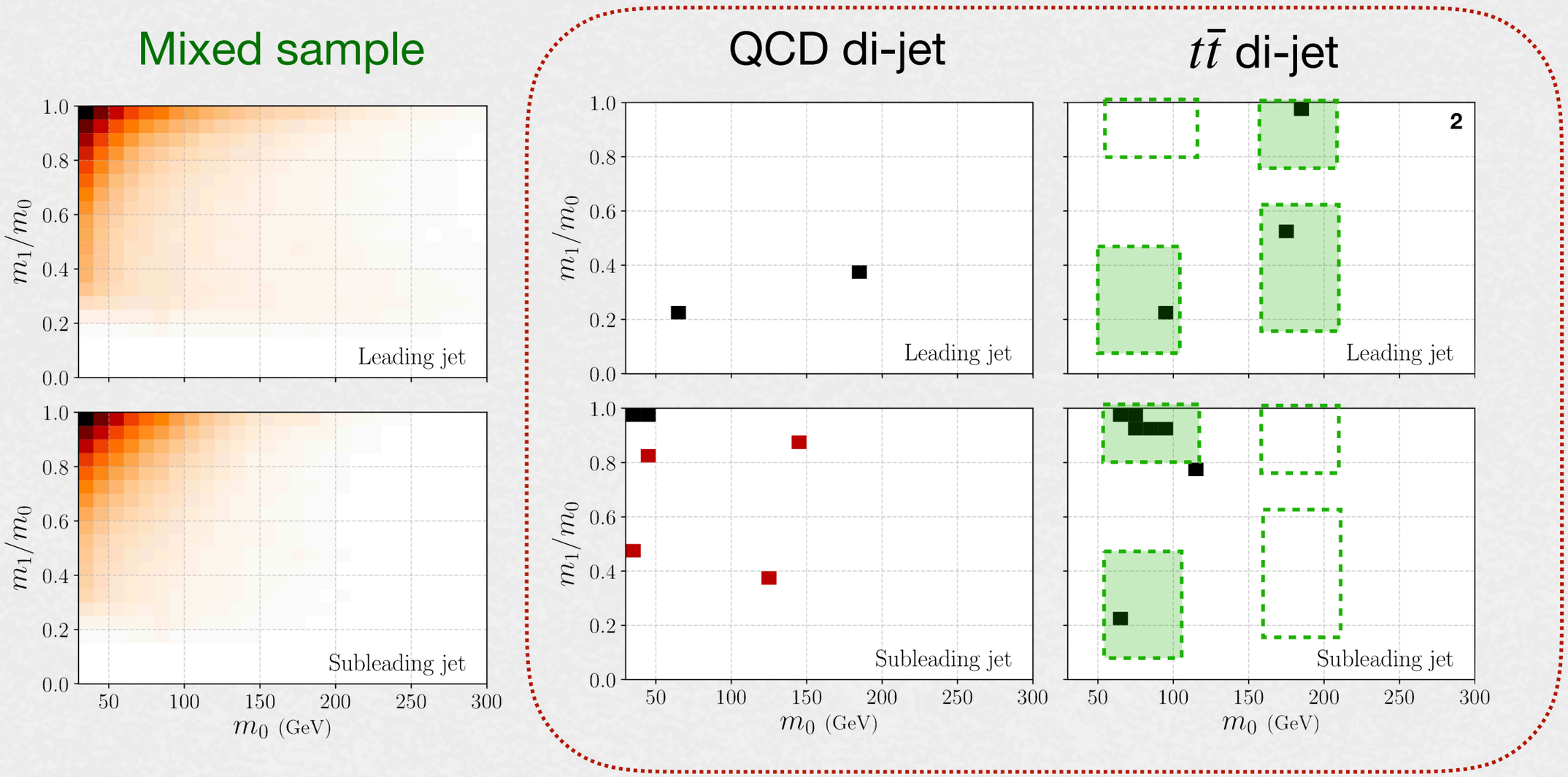
- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



What does the algorithm learn?

Why

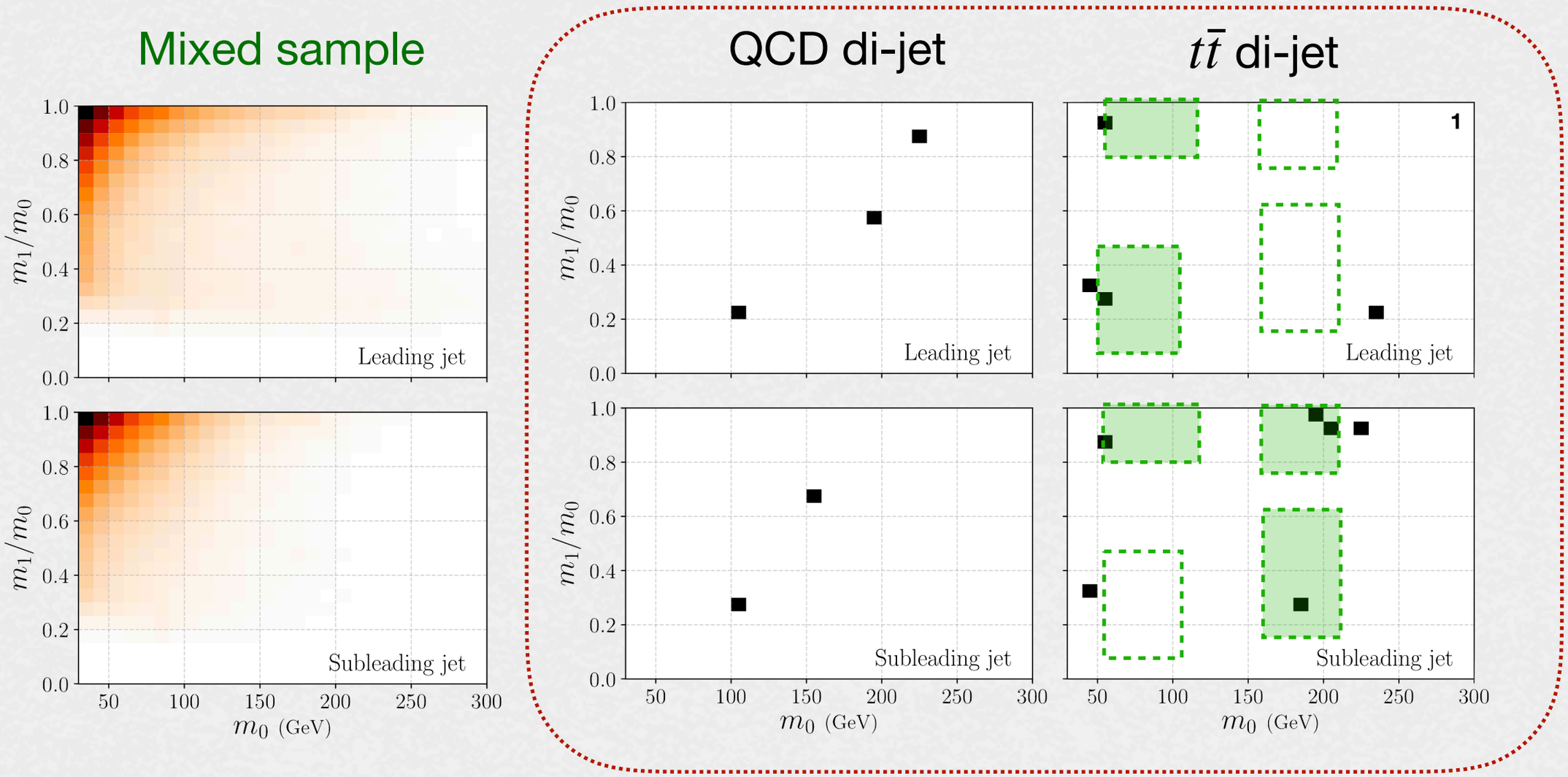
- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



What does the algorithm learn?

Why

- The answer: co-occurrences in $o_{j,i}$
- The VI algorithm assigns co-occurring $o_{j,i}$ large weights in the same theme



Learning new physics

- Signal: $pp \to W' \to (\phi \to WW)W$

  $m_{W'} = 3\ TeV,\ \ m_{\phi} = 400\ GeV, S/B = 5\,\%$

  Collins et al (hep-ph/1805.02664)

- Observables: primary Lund plane, $o_{j,i} = [J, \log k_T, \log R/\Delta]$

  Dreyer et al (hep-ph/1807.04758)

Learning new physics

- Signal: $pp \to W' \to (\phi \to WW)W$

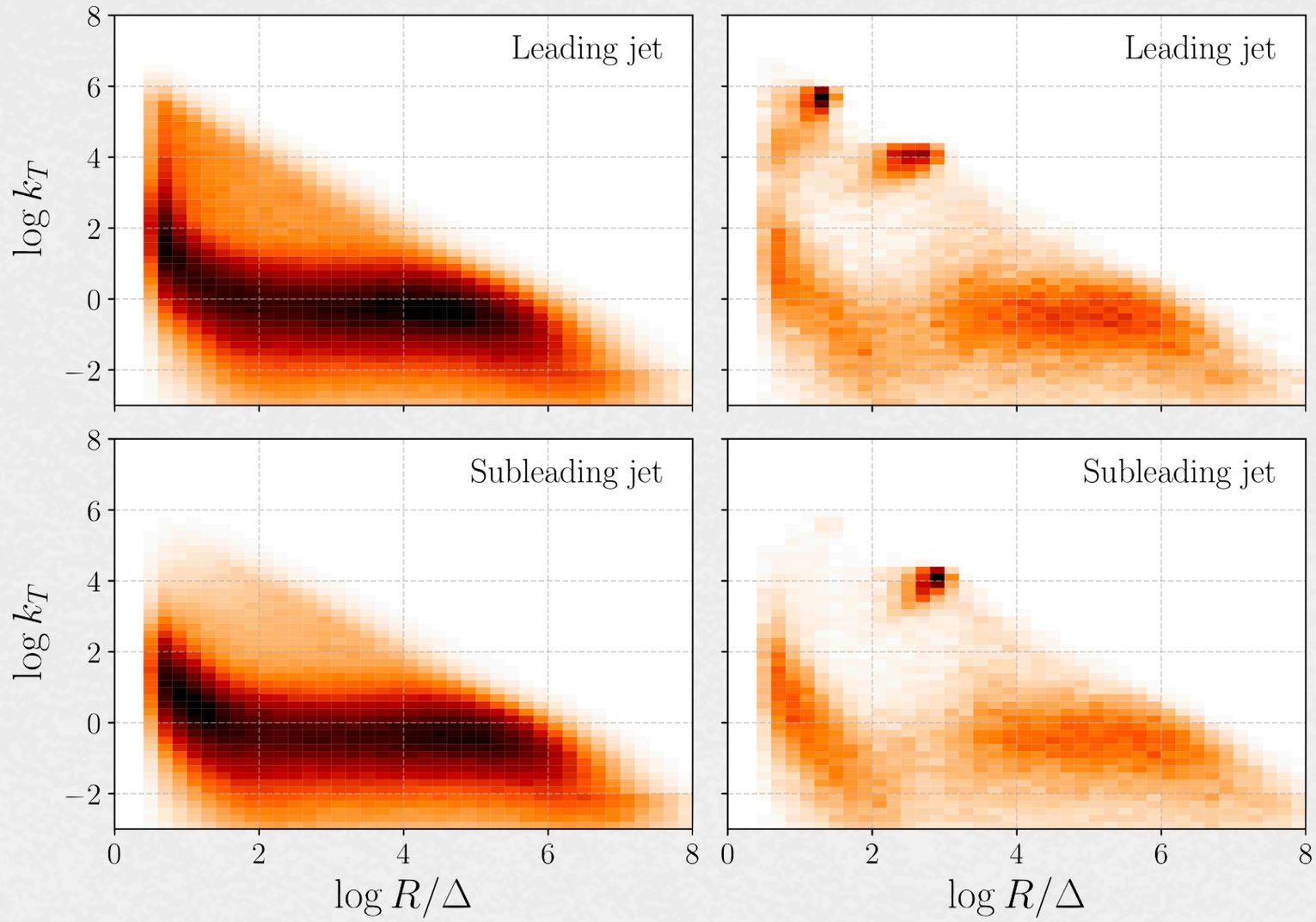
  $m_{W'} = 3\ TeV,\ \ m_{\phi} = 400\ GeV, S/B = 5\,\%$

  Collins et al (hep-ph/1805.02664)

- Observables: primary Lund plane, $o_{j,i} = [J, \log k_T, \log R/\Delta]$

  Dreyer et al (hep-ph/1807.04758)

- **Truth-level distributions**



Learning new physics

- Signal: $pp \to W' \to (\phi \to WW)W$

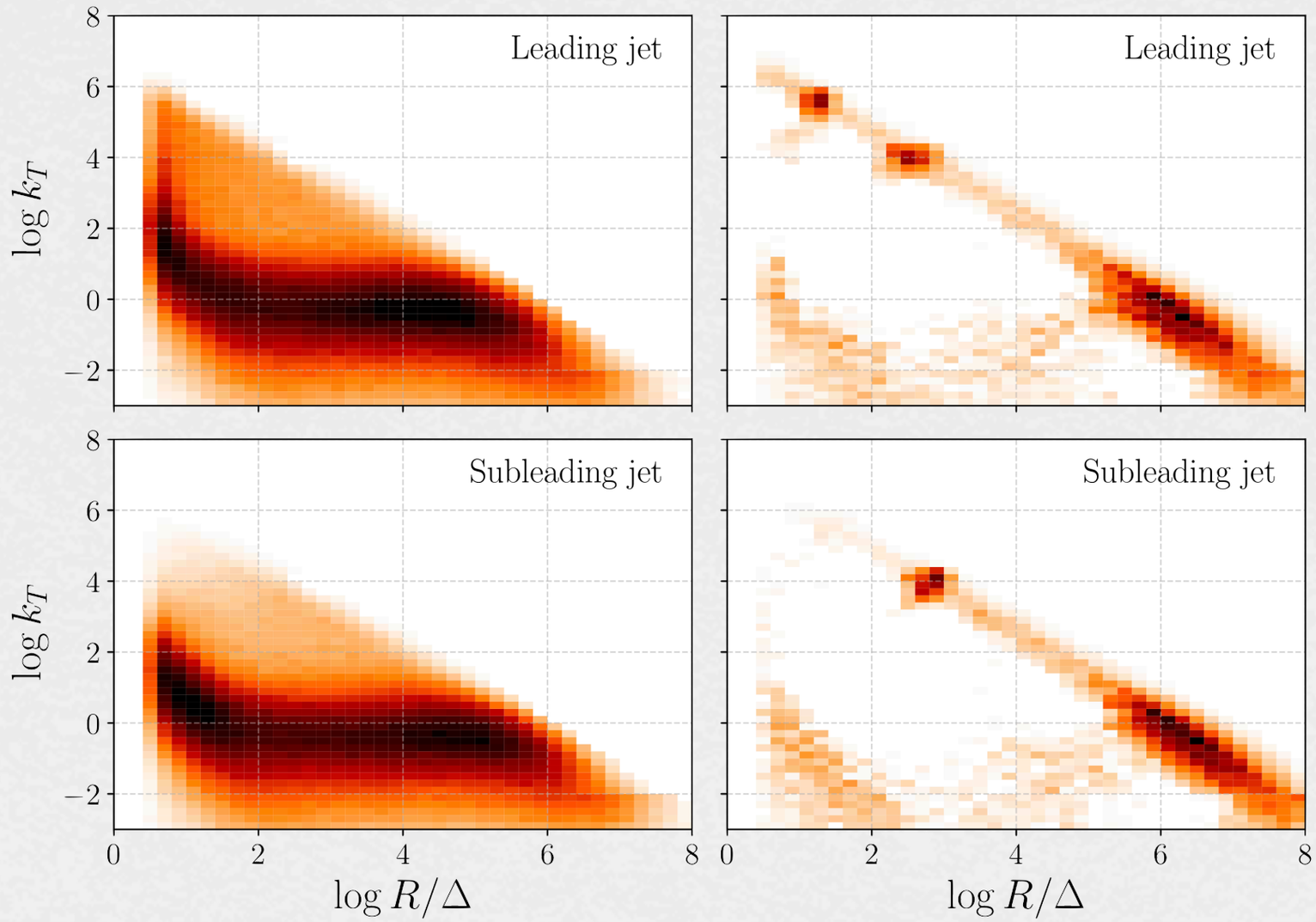
  $m_{W'} = 3\ TeV,\ \ m_{\phi} = 400\ GeV, S/B = 5\,\%$

  Collins et al (hep-ph/1805.02664)

- Observables: primary Lund plane, $o_{j,i} = [J, \log k_T, \log R/\Delta]$

  Dreyer et al (hep-ph/1807.04758)

- **Learned LDA themes**



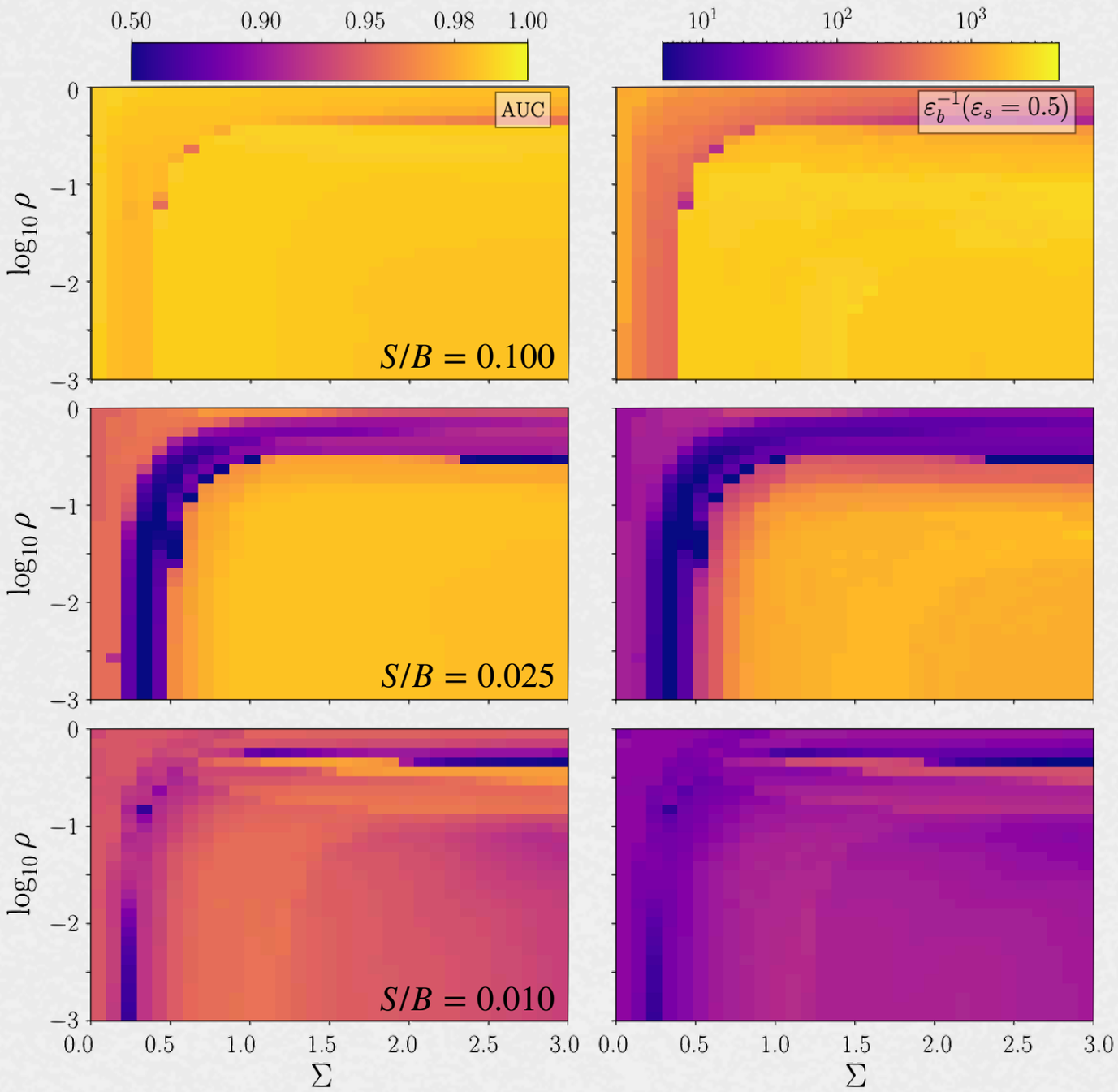
Classification

- Same signal, but now with $o_{j,i} = [J, m_0, m_1/m_0]$

- Likelihood ratio:

$$L(e_j) = \prod_{i=1}^{N_j} \frac{p(o_{j,i}|\beta_2)}{p(o_{j,i}|\beta_2)}$$

- We choose optimal ρ and Σ using the model and data only.



Conclusions

- Unsupervised generative model to learn rare signals in collider data
- Variational inference technique relies on co-occurrences in the event data
- **Co-occurrences are crucial to describe signals with unsupervised learning**
- In our most recent paper:
  - more detailed intro to probabilistic modelling for collider physics
  - study of co-occurrences vs bin sizes and #observables
  - study of perplexity ~ performance
  - study of variational inference technique
- This LDA technique should in principle work for other event types
  → work on 4-top study in progress
- The technique is ready-to-go for use in di-jet new physics searches
  → another work in progress (LHC Olympics)

Additional slides

The generative model

- We want to quantify the probability that the events were generated

- Generative process:

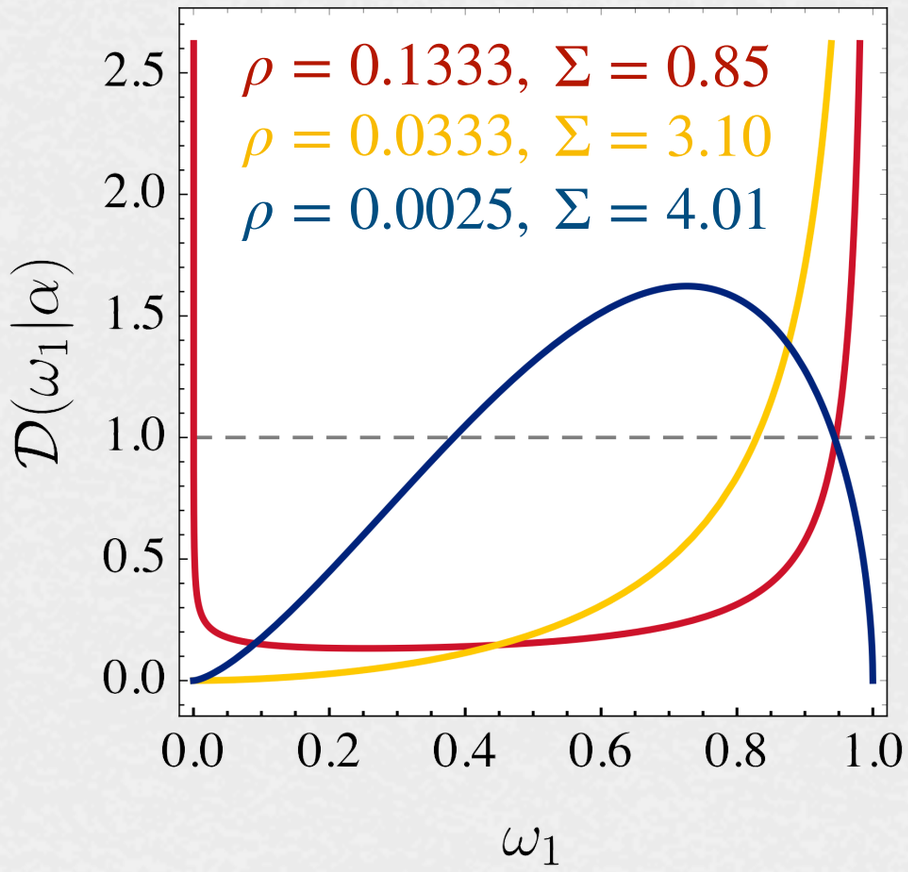
  1. Sample theme weights from the prior $\omega \sim p(\omega \mid \alpha)$
  2. Sample a theme using the theme weights $t \sim p(t \mid \omega)$
  3. Sample a splitting from the relevant theme $o_{j,i} \sim p(o_{j,i} \mid t, \beta)$
  4. Repeat 2 & 3 for each $o_{j,i}$ in the event
  5. Repeat 1-4 for each event e_j in the sample

- Latent Dirichlet Allocation (LDA):
  Blei et al (2003)

$$p(e_j \mid \alpha, \beta) = \int d\omega \; p(\omega \mid \alpha) \prod_{i=1}^{N_j} \sum_{t=1}^{T} p(o_{j,i} \mid t, \beta) \; p(t \mid \omega)$$

The prior and rare signals

- The Dirichlet prior with **2 themes**



- Reminder: theme weights per event are sampled from the Dirichlet prior

- $\rho \sim$ ratio of themes present in sample
  $\rho \ll 1 \Rightarrow$ one theme representing **rare** co-occurring features in the sample

- We can estimate the optimal ρ and Σ from the model and the data alone.

  Perplexity $\sim$ lower bound on the model evidence, $\prod_{j=1}^{N_e} p(e_j)$.