ANOMALY SCORE LHC OLYMPICS CHALLENGE

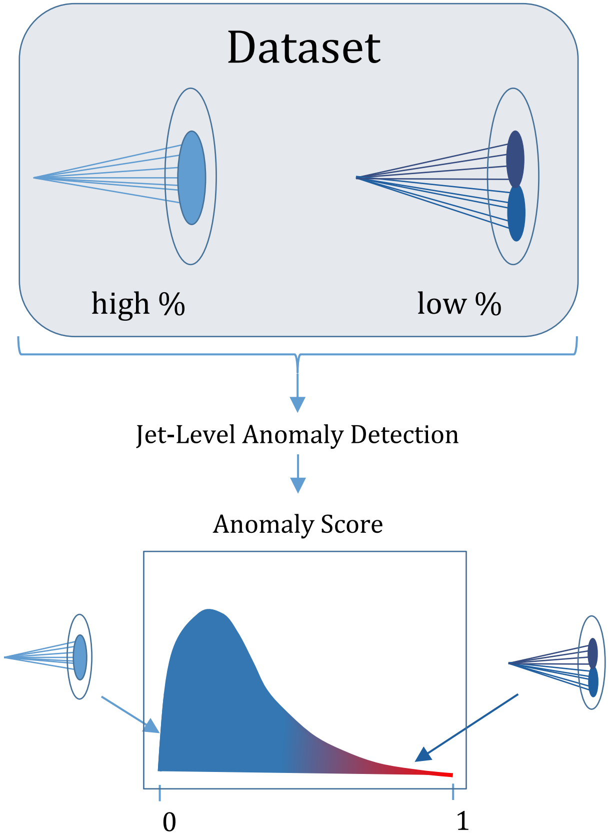
Gustaaf Brooijmans, Julia Gonski, Alan Kahn, Inês Ochoa, Daniel Williams

ML4Jets 2020

January 16, 2020

Intro

- Model: Variational Recurrent Neural Network
  - Sequence-modeling architecture
  - Use case: *Jet-Level Anomaly Detection*
    - Provides one *Anomaly Score* per jet in the training set



Dataset

Jet-Level Anomaly Detection

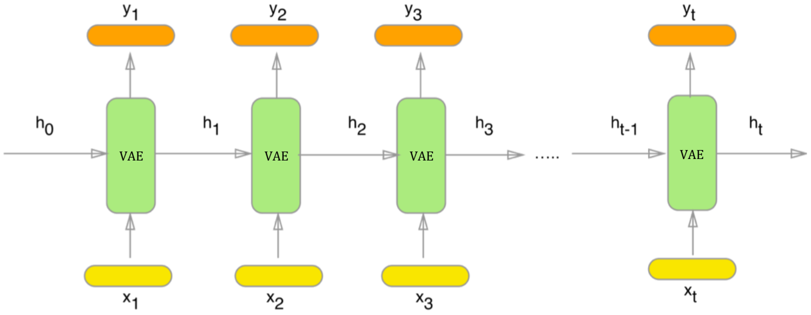
Anomaly Score

- Discussion
  - Model Details
  - Approach to this challenge
    - Data processing/evaluation
  - Results

Variational Recurrent Neural Network (VRNN)

- Based on a *Recurrent Neural Network* (RNN)
  - Architecture which allows for variable amount of inputs
  - A hidden state gets updated after each iteration of the RNN
  - Hidden state can then be decoded to produce outputs
  - Used primarily in sequence modeling
  - Allows us to process a variable number of constituents

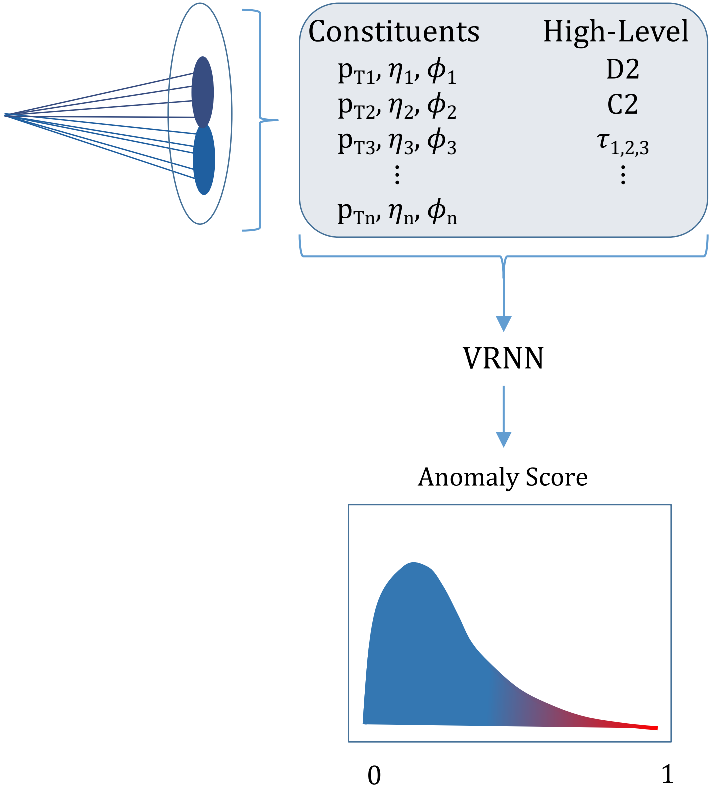
- *Variational Recurrent Neural Network* (paper)
  - Replaces the encoder/decoder step of the RNN with a VAE
  - In addition to constituents, can also input a list of high-level variables



Model Inputs

- Inputs to the model consist of (jet-by-jet)
  - Constituent 4-vectors
    - Up to first 10 constituents (sorted in p_T)
  - High-Level Variables (HLVs)
    - Jet Substructure

- Preprocessing Strategy:
  - Cluster with anti-k_t , R=1, trimmed
  - Calculate Jet Substructure High-Level Variables (HLVs)
    - C2, D2, τ_1, τ_2, τ_3, τ_2/τ_1, τ_3/τ_1, τ_3/τ_2, Split12, Split23
    - Added to and implemented in *pyjet* – python interface to fastjet
  - Boost all jets to reference energy, mass
    - Flat energy spectrum -> alleviates mass, pt correlation
  - Save list of constituent 4-vectors, HLVs for each event
    - Leading and Sub-Leading jets



Constituents	High-Level
p_{T1}, η_1, ϕ_1	D2
p_{T2}, η_2, ϕ_2	C2
p_{T3}, η_3, ϕ_3	$\tau_{1,2,3}$
⋮	⋮
p_{Tn}, η_n, ϕ_n	

VRNN

Anomaly Score

Training and Evaluation

- First, model is *pre-trained* on only constituent information
  - Constituent information tends to get ignored when training on combined constituent+HLV from scratch
  - Weights are saved when performance is at a maximum

- Then, HLV information is added, and training continues
  - Overall, *increases performance*

- Loss Function: $MSE + \lambda D_{KL}$
  - MSE = Mean-Squared Error between input and output constituent components
  - D_{KL} = KL Divergence from latent space prior to the learned latent space distribution
  - λ = constant, ¼ in current tests

- Evaluation metric: *Anomaly Score*
  - Anomaly Score = $1 - e^{-\rho}$
  - $\rho = D_{KL} + \min \sum d_{ij}$
  - $d_{ij} = \min(k_{ti}^2, k_{tj}^2)(\Delta\phi_{ij}^2 + \Delta\eta_{ij}^2)$
    - k_t distance for R=1 jet

- Jet-Level to Event-Level:
  - Determine *Anomaly Score* for leading, sub-leading jets
  - *Event Score* = max(Anomaly Score 1, Anomaly Score 2)
  - Training only on the two leading jets in each event

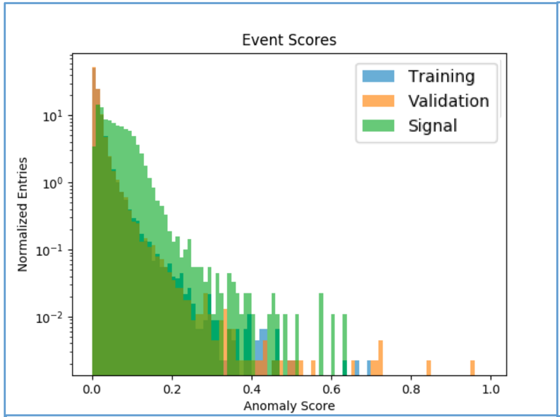
Proof-of-Concept – R&D Dataset

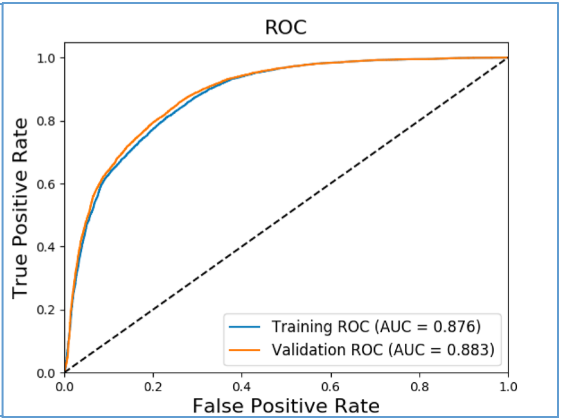
- Split into 3 sets
  - Training (~45k events)
  - Validation (~45k events)
  - Signal (~10k events)
  - Contaminated (100k events)
    - Sum of 3 split datasets

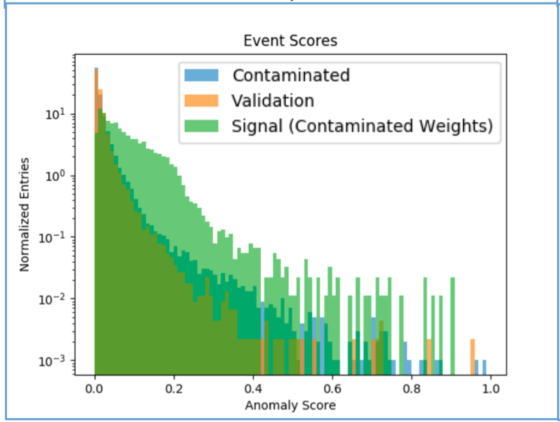
- Two training prescriptions
  - Train on Training set
  - Train on Contaminated set
    - True Anomaly Detection setup

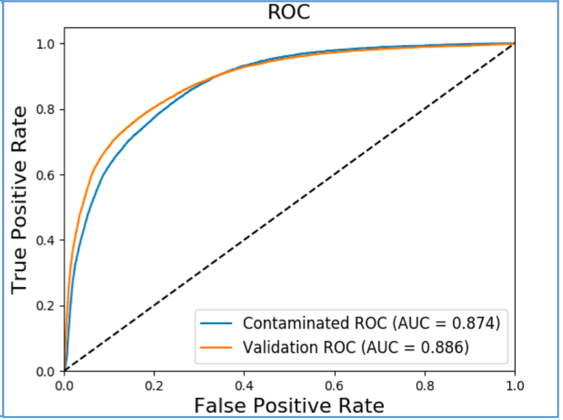
- Both scenarios show consistent performance

- Training on contaminated dataset produces longer tails in anomaly score

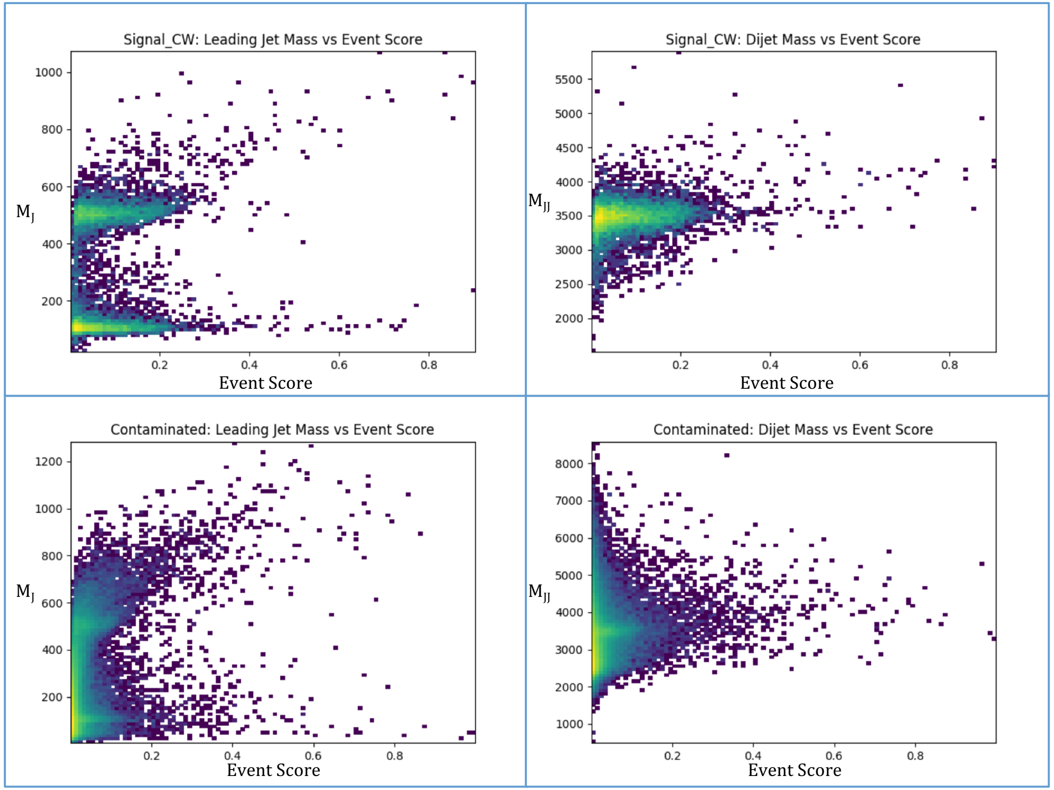








R&D: Jet Mass vs Score

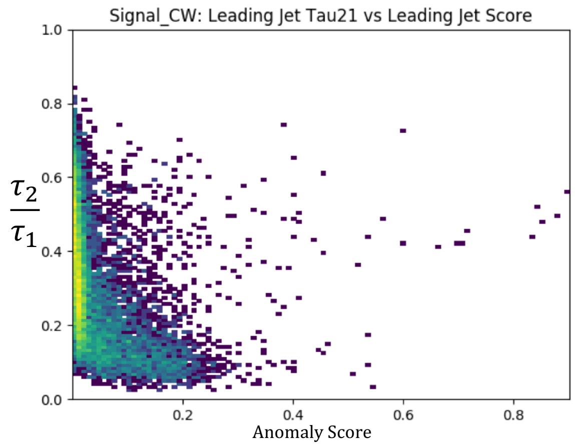


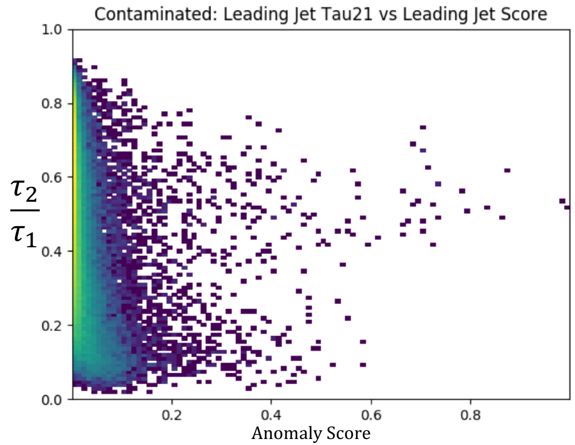
- Signal populates region of higher Anomaly Score
- Signal dijet peak also correlated in contaminated set
  - Dijet computed as sum of leading, sub-leading jets

R&D: Substructure Variables vs Score

- Anomaly score correlates with signal-indicating substructure variables
  - Example: τ_2/τ_1
    - Low τ_2/τ_1 -> more two-prong like

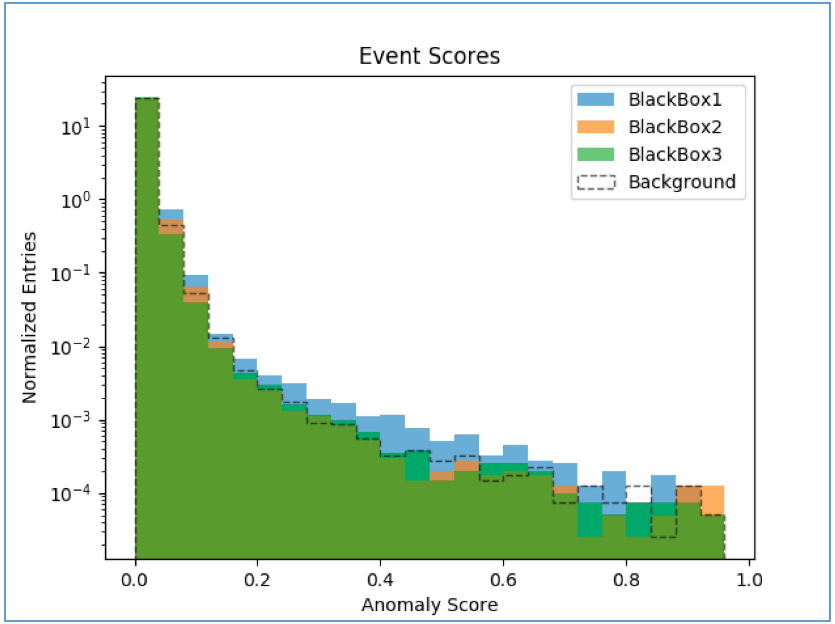
- Correlation still present in contaminated dataset





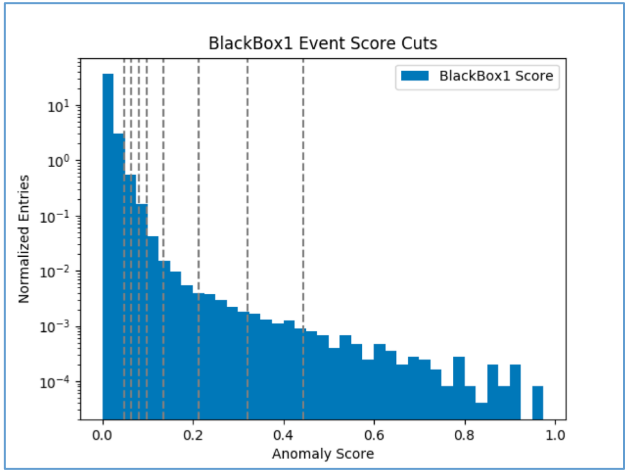
Black Box Analyses

- Procedure
  - Train on each dataset (Background, Black Box 1,2,3)
  - Study the dijet mass in bins of event score acceptance
- Idea:
  - Rely on acceptance due to unique network weights for each dataset
  - Datasets with signal contamination will contain high S/B in the same acceptance bin w.r.t. pure-background
    - Searching for M_{JJ} type resonance in high anomaly score regions
  - Can use lower score, higher acceptance bins for background estimation
    - Acts as background-enriched control region



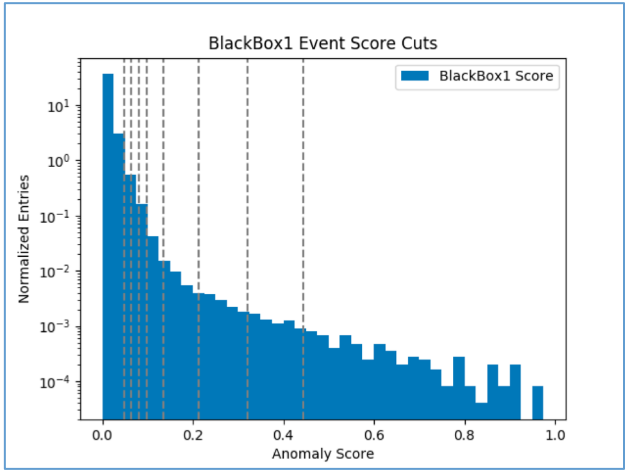
Example – Black Box 1

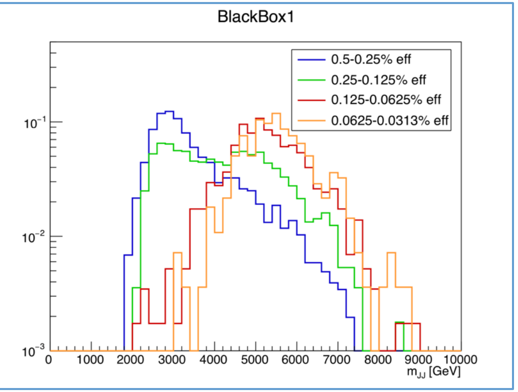
- Cut in exclusive bins of anomaly score acceptance
  - $2\%, 1\%, \frac{1}{2}\%, \frac{1}{4}\%, \frac{1}{8}\%, \frac{1}{16}\%, \frac{1}{32}\%, \frac{1}{64}\%$



Example – Black Box 1

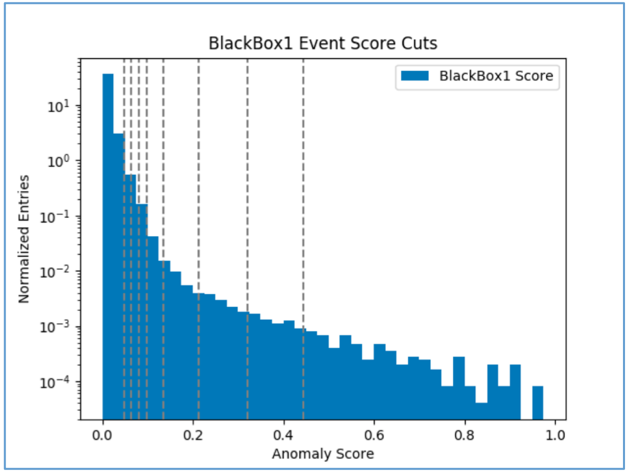
- Cut in exclusive bins of anomaly score acceptance
  - $2\%, 1\%, \frac{1}{2}\%, \frac{1}{4}\%, \frac{1}{8}\%, \frac{1}{16}\%, \frac{1}{32}\%, \frac{1}{64}\%$
- Plot M_{JJ}

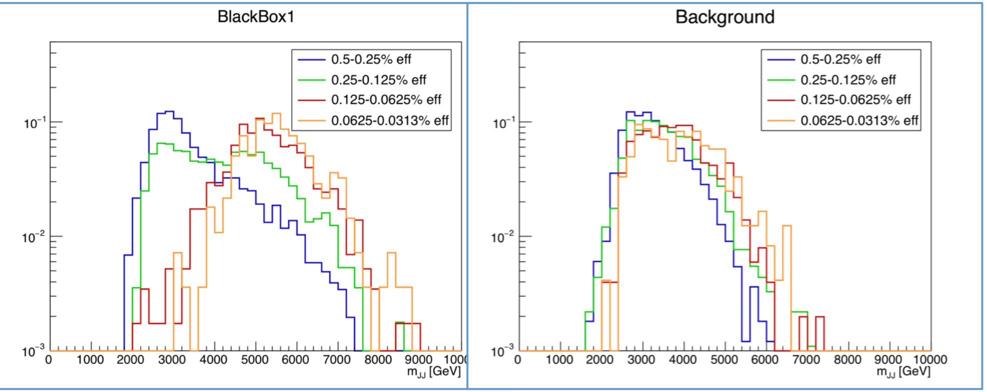




Example – Black Box 1

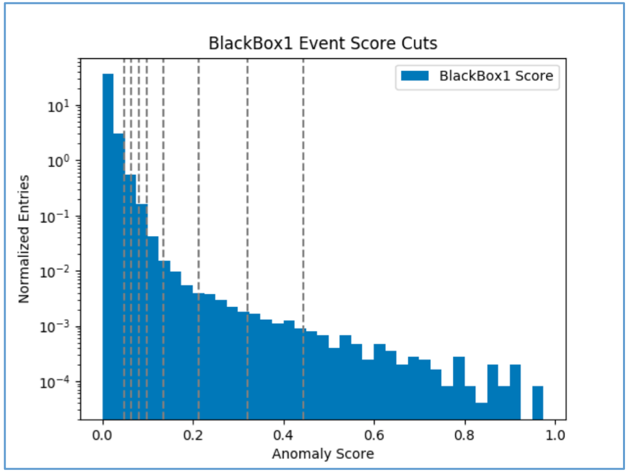
- Cut in exclusive bins of anomaly score acceptance
  - $2\%, 1\%, \frac{1}{2}\%, \frac{1}{4}\%, \frac{1}{8}\%, \frac{1}{16}\%, \frac{1}{32}\%, \frac{1}{64}\%$
- Plot M_{JJ}
- Compare to background-only sample/background-enriched control region

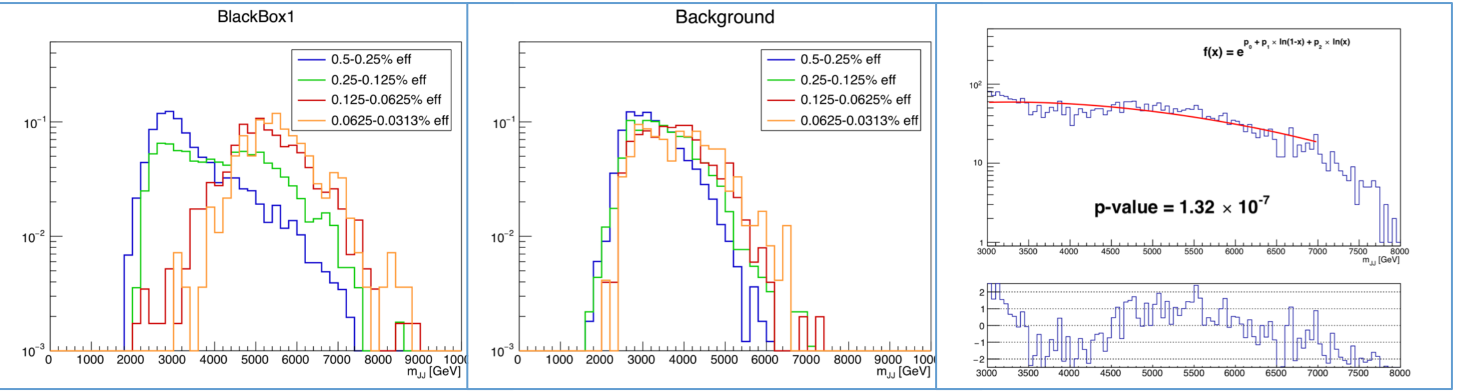




Example – Black Box 1

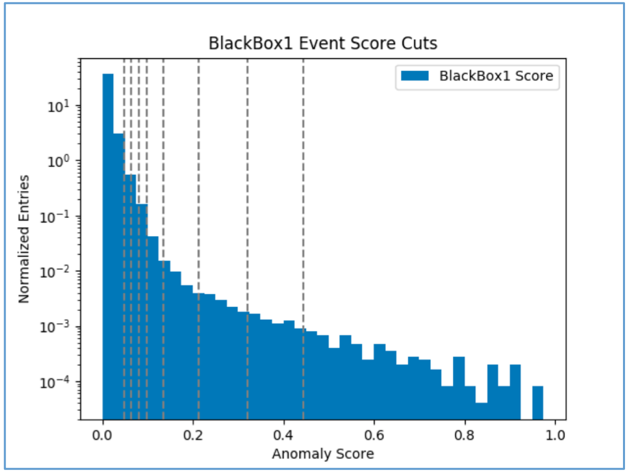
- Cut in exclusive bins of anomaly score acceptance
  - $2\%, 1\%, \frac{1}{2}\%, \frac{1}{4}\%, \frac{1}{8}\%, \frac{1}{16}\%, \frac{1}{32}\%, \frac{1}{64}\%$
- Plot M_{JJ}
- Compare to background-only sample/background-enriched control region
- Fit distribution with ad-hoc "dijet" function to determine p-value
  - $p_1(1-x)^{p_2}x^{-p_3}$

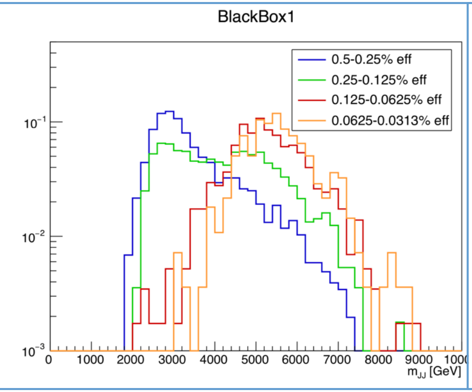


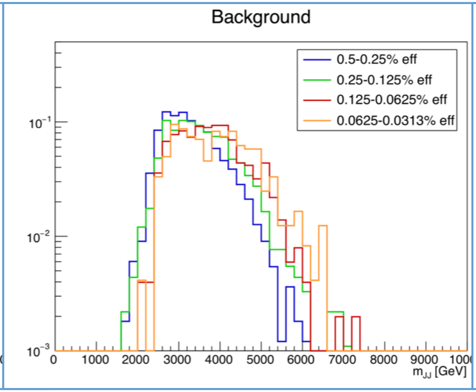


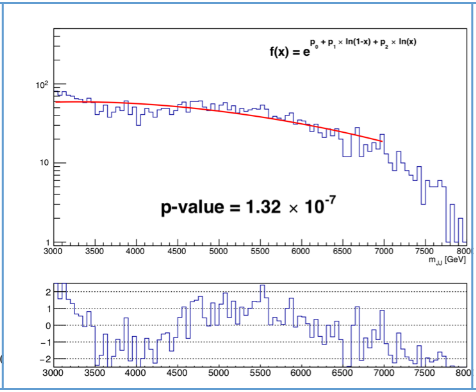
Example – Black Box 1

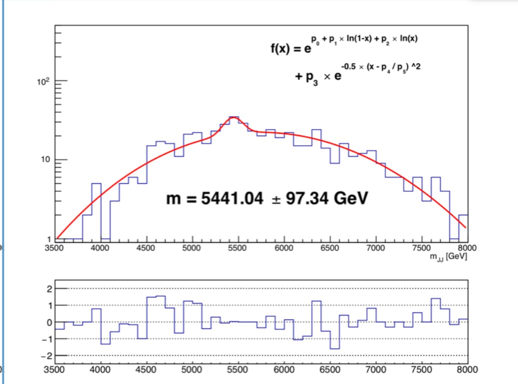
- Cut in exclusive bins of anomaly score acceptance
  - $2\%, 1\%, \frac{1}{2}\%, \frac{1}{4}\%, \frac{1}{8}\%, \frac{1}{16}\%, \frac{1}{32}\%, \frac{1}{64}\%$
- Plot M_{JJ}
- Compare to background-only sample/background-enriched control region
- Fit distribution with ad-hoc “dijet” function to determine p-value
  - $p_1(1-x)^{p_2}x^{-p_3}$
- Fit distribution with dijet+gaussian to determine mass+width of resonance



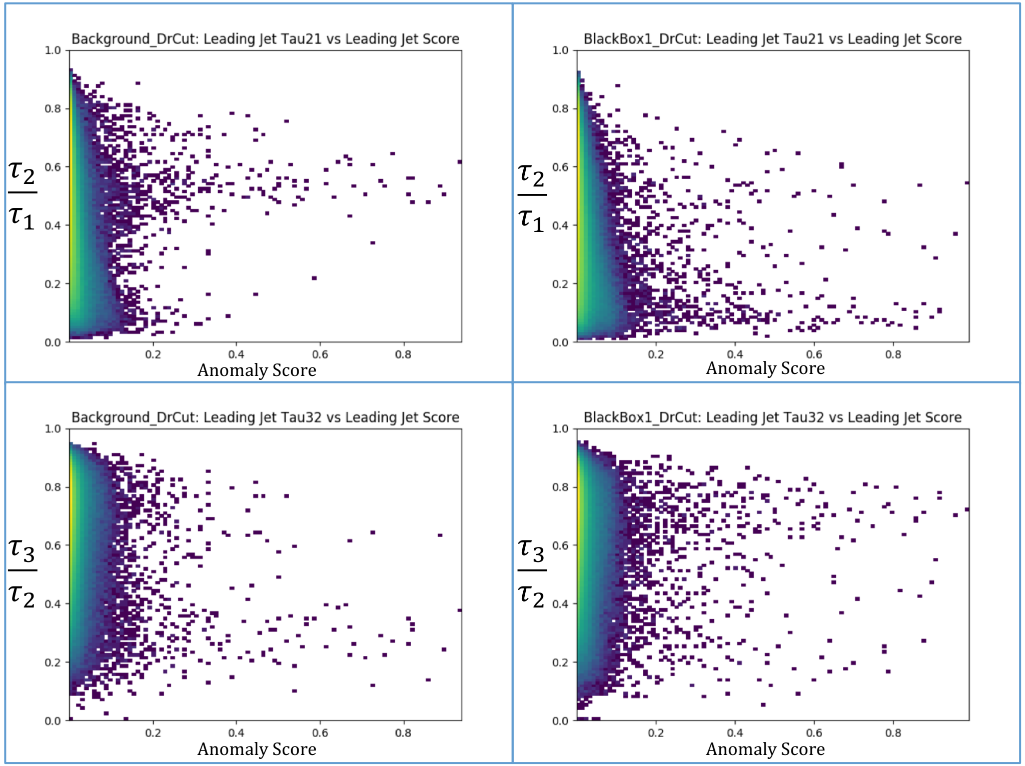








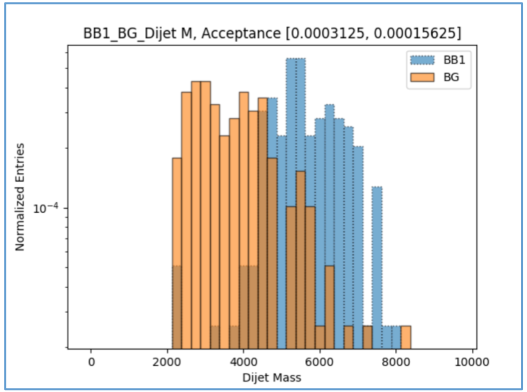
Black Box 1: Substructure vs Anomaly Score

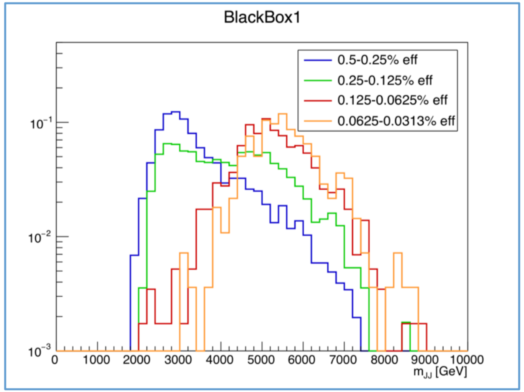


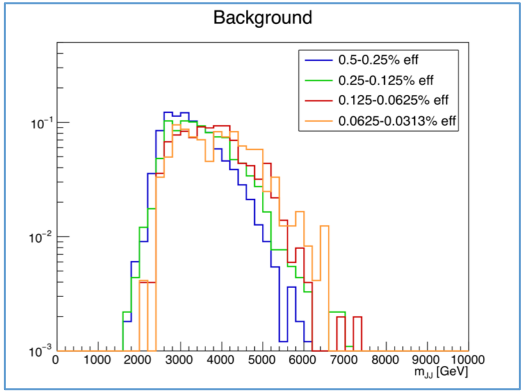
- Black Box 1 anomaly score correlated with low τ_2/τ_1 and high τ_3/τ_2
- Consistent with two-pronged jet substructure

Results – Black Box 1

- Signal visible with mass 5441 ± 97 GeV
  - Decreasing acceptance -> Higher S/B -> Convergence on signal peak
- Physics description: A->BC->JJ
  - New heavy resonance to two new particles, decaying to (boosted) $q\bar{q}$

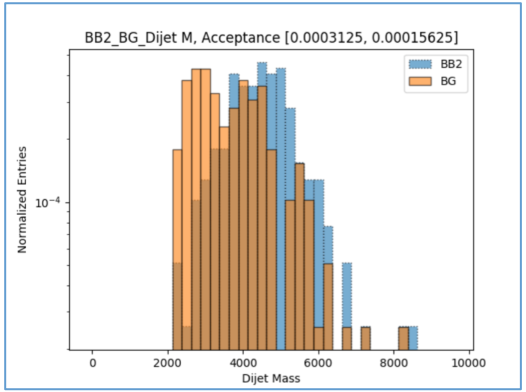


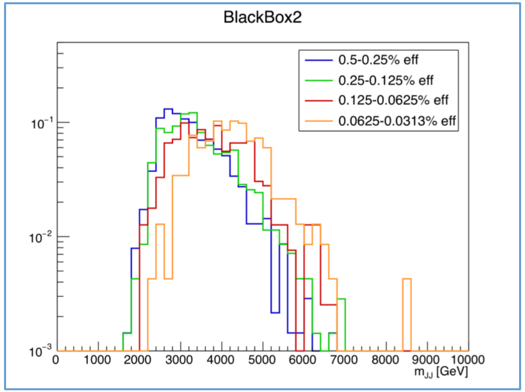


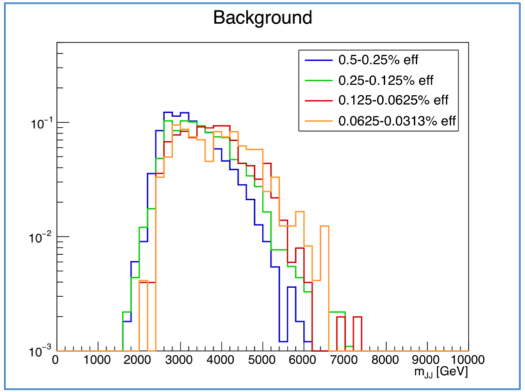


Results – Black Box 2

- Evidence of signal at ~4500GeV
  - Much less prevalent than Black Box 1
- Difficult to fit background + signal bump
  - Smoothly falling M_{JJ} background sculpted by event score cut

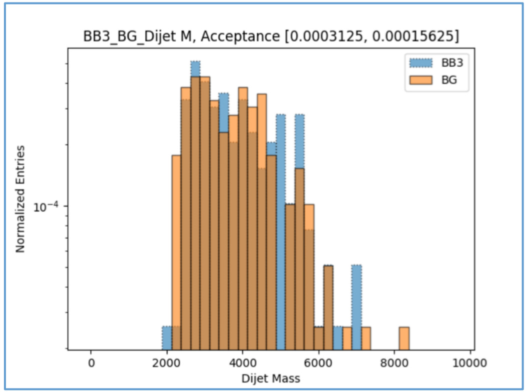


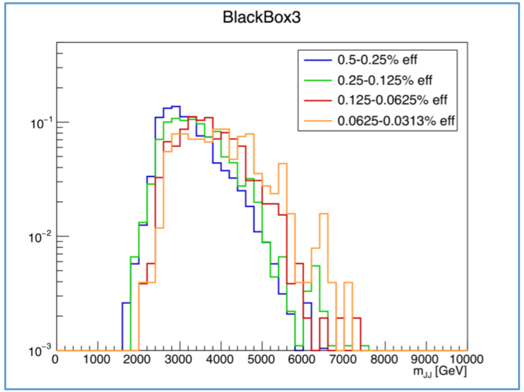


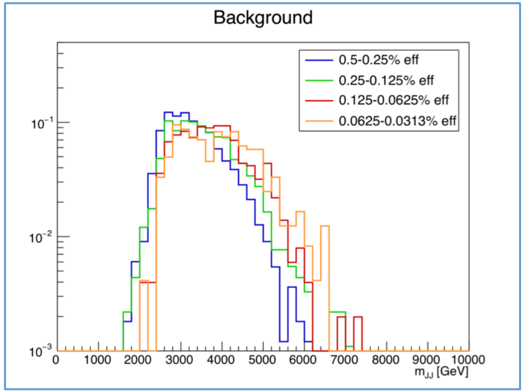


Results – Black Box 3

- No clear signal visible
  - Potential reasons
    - Not present
    - Mass ~3TeV
    - Signal we’re not sensitive to







Conclusion

- Our VRNN model provides a very promising way of performing jet-level anomaly detection

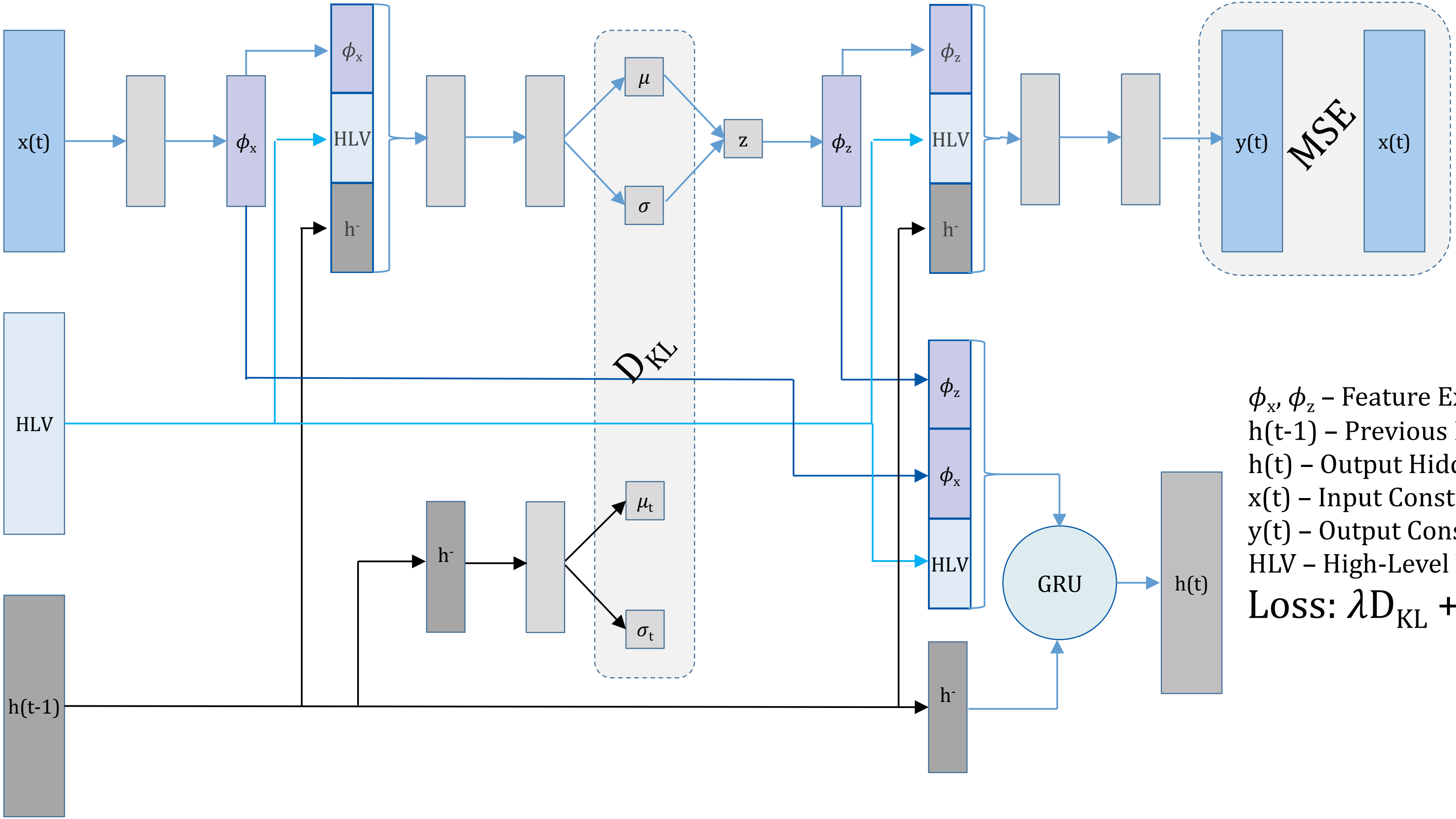
- What we learned
  - Training on contaminated dataset allows for signal sensitivity
  - Built full analysis workflow for future studies
    - Added substructure variables to pyjet
  - Definitions signal regions using Anomaly Score
  - First application of this model to toy analysis

- Future studies
  - Understand Anomaly Score behavior better
    - Undesired correlations? (Background mass sculpting)
  - Background estimation
    - Defining control regions
  - Jet-level vs. event-level Anomaly Score
  - Benchmark performance in well-defined signals

- Thank you to the organizers of ML4Jets!

BACKUP

VRNN Cell + High-Level Variables

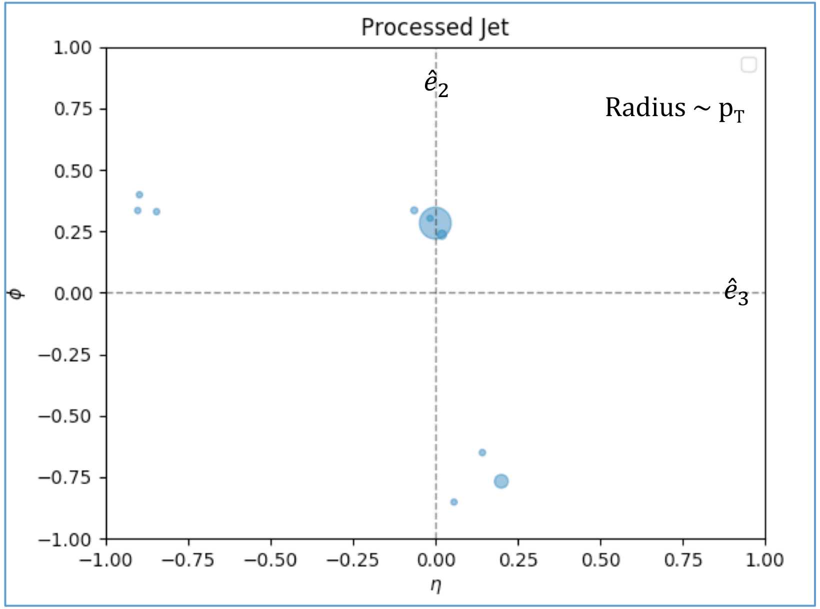


ϕ_x, ϕ_z – Feature Extractors
h(t-1) – Previous Hidden State
h(t) – Output Hidden State
x(t) – Input Constituent
y(t) – Output Constituent
HLV – High-Level Variables

Loss: λD_{KL} + MSE

Jet Pre-Processing

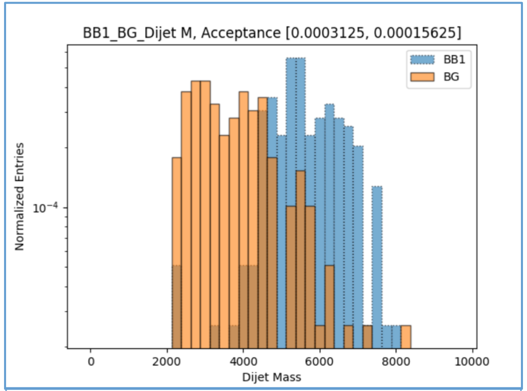
- Mostly follows this paper
- p_T Cut: 150GeV
- Rescale to 50GeV mass: $p^\mu \to \frac{50GeV}{M} p^\mu$
- Boost to 100GeV energy: $p^\mu \to \Lambda^\mu_\nu p^\nu$
- Define new coordinates:
  - $\hat{e}_1$ along jet axis
  - $\hat{e}_2$ pointing to hardest constituent
  - $\hat{e}_3$ pointing to second hardest constituent
    - Hardest constituent will have $e_3 = 0$
    - Second hardest constituent will have $e_3 > 0$
- Define new p_T, η, ϕ
  - p_T along $\hat{e}_1$
  - η, ϕ as $\Delta\eta, \Delta\phi$ wrt $\hat{e}_1$
- Remove constituents with $\Delta R = \sqrt{\eta^2 + \phi^2} > 1$
- Remove constituents with p_T fraction < 1%
- Normalize p_T
  - $p_{Ti} \to \frac{p_{Ti}}{\sum_j p_{Tj}}$
- Save processed p_T, η, ϕ for each constituent

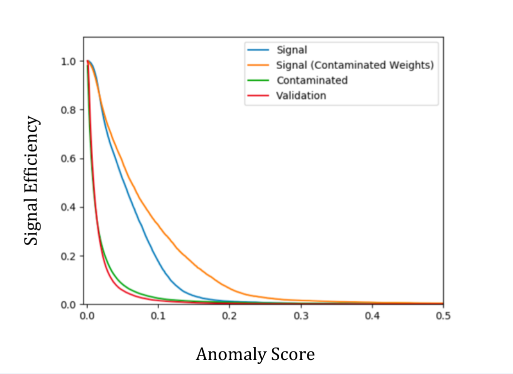


Estimating Number of Signal Events

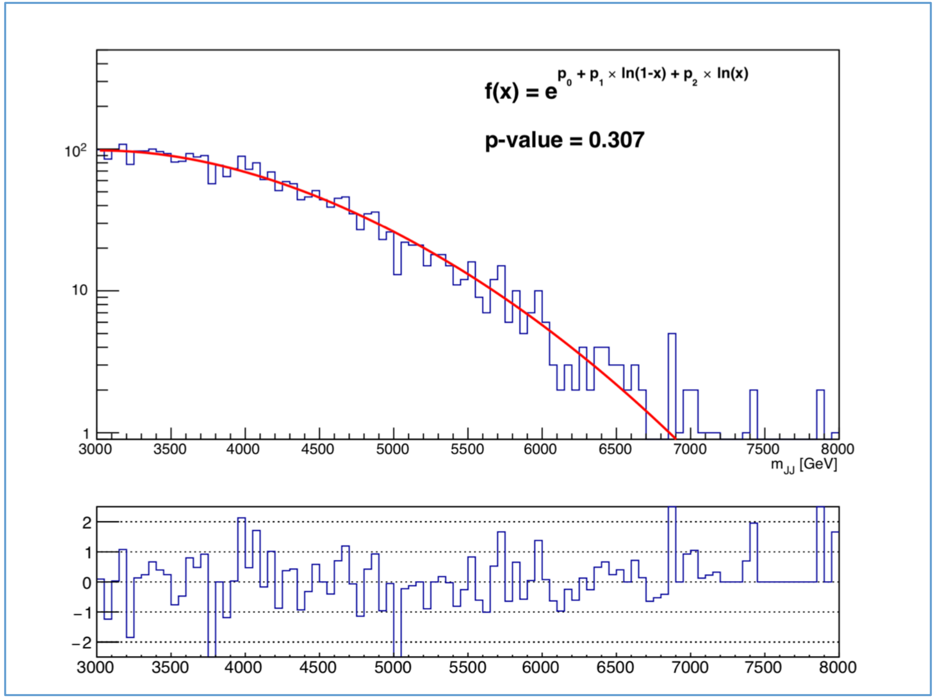
- Using anomaly score cut value from Black Box…
    - Cut on both Black Box and Background
    - Compare stats
        - Difference in stats -> number of signal events
    - Divide by signal efficiency
        - Determined by study on labeled dataset
    - Result is a (conservative) upper limit on the number of signal events

- Results:
    - Black Box 1: 86270
    - Black Box 2: 11926
    - Black Box 3: 7455



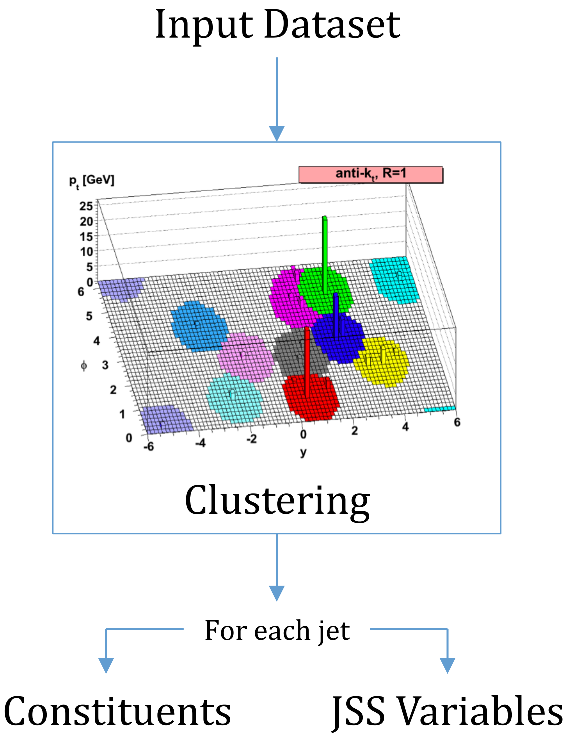


Background dijet fit



Approach

- Preprocess data
  - Cluster with anti-k_t , R=1
  - Calculate Jet Substructure High-Level Variables (HLVs)
    - C2, D2, τ_1, τ_2, τ_3, τ_2/τ_1, τ_3/τ_1, τ_3/τ_2, Split12, Split23
  - Save list of constituent 4-vectors, HLVs for each event
    - Leading and Sub-Leading jets



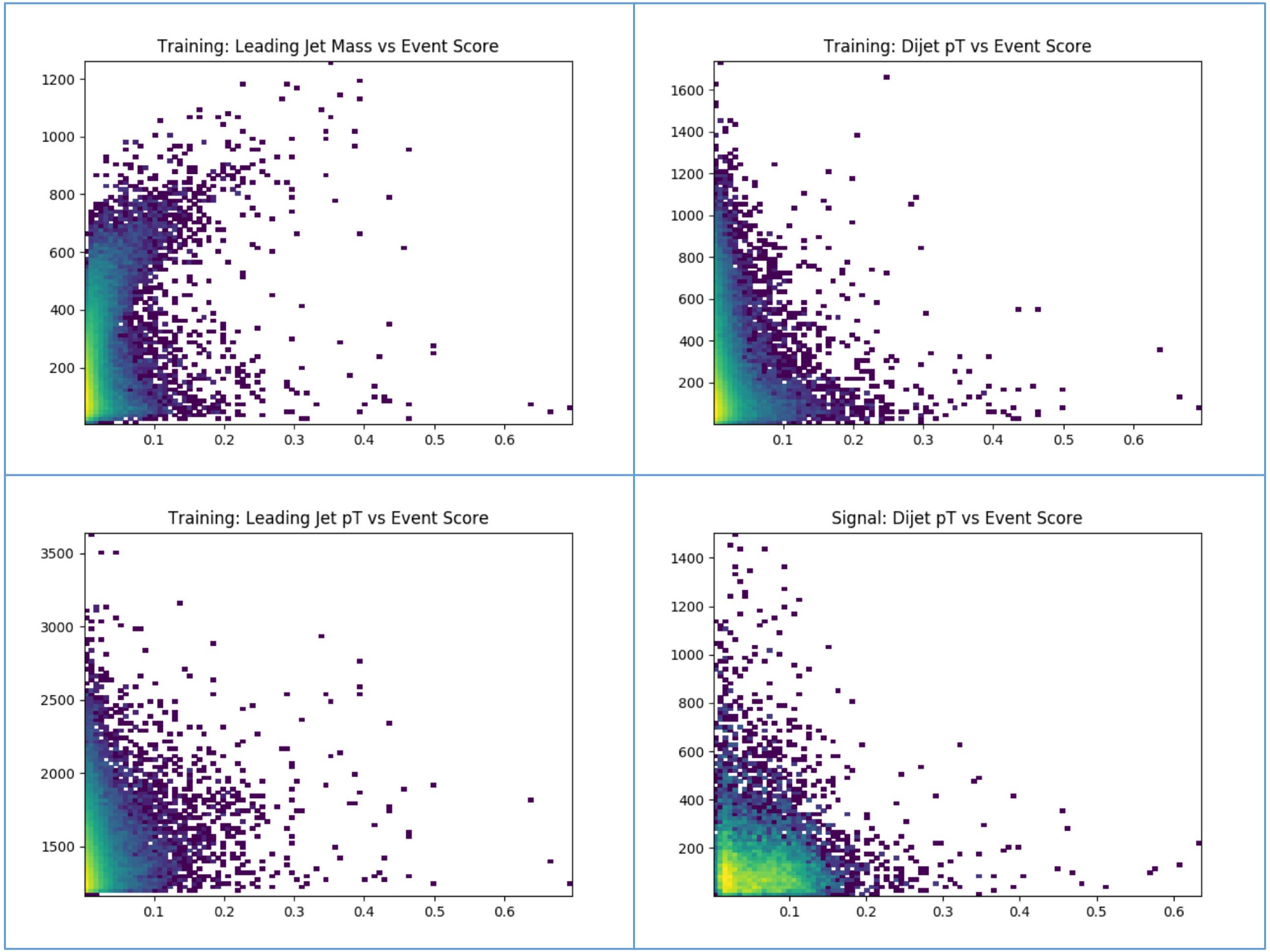
Input Dataset

Clustering

For each jet

Constituents

JSS Variables



Training: Leading Jet Mass vs Event Score

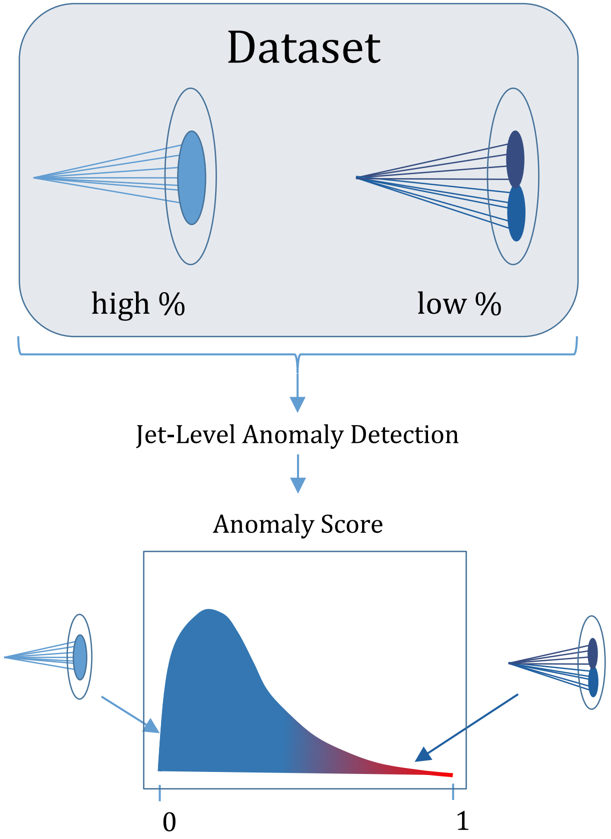
Training: Dijet pT vs Event Score

Training: Leading Jet pT vs Event Score

Signal: Dijet pT vs Event Score

Motivation

- *Model Independent* signal jet identification
  - “Anti-QCD Tagging”
  - Applicable to:
    - General searches
    - Analyses supported by multiple physics models
    - Unpredicted new signals
- *Unsupervised training*
  - No labels
  - Ability to train on data
- *Anomaly Detection*
  - Discovering elements within a dataset which are produced by a different, unusual mechanism
  - A sample of jets with loose cuts will be comprised of mostly light-jet QCD, with signal (t, W/Z, b, BSM, etc..) acting as some small contamination
- **ML Architecture: Variational Recurrent Neural Network (VRNN)**



Dataset

high %

low %

Jet-Level Anomaly Detection

Anomaly Score

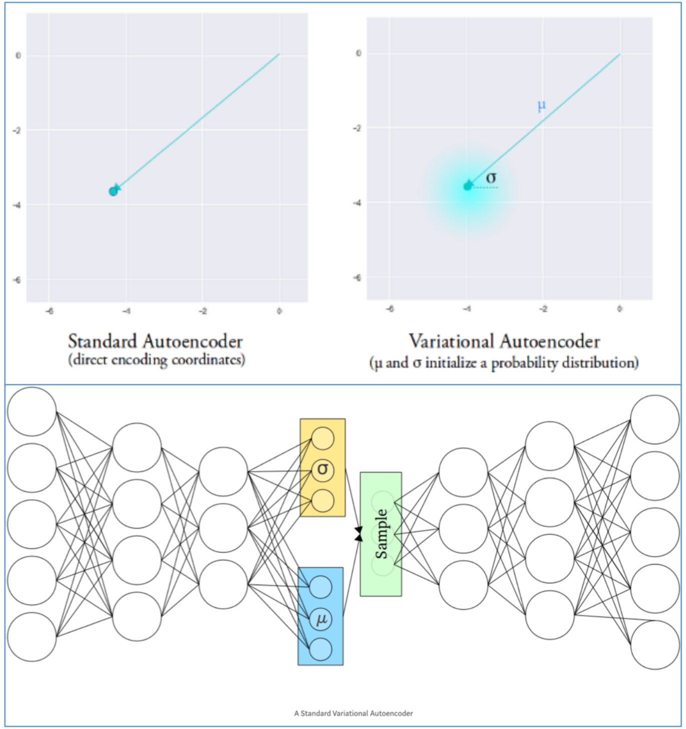
Variational Autoencoders (VAEs)

- Observed data x is produced by a random latent variable z

- Variational encoders perform Bayesian Inference to approximate the posterior distribution of z
  - Encoder: $q(z|x)$ *Approximate Posterior*
  - Decoder: $p(x|z)$ *Data Reconstruction*

- How: Maximizing the Evidence Lower Bound
  - $$\mathcal{L} = \underbrace{\mathbb{E}_z\,[\log p(x|z)]}_{\substack{\text{Reconstruction Accuracy} \\ \text{(Mean-Squared-Error)}}} - \underbrace{D_{KL}(q(z|x)||p(z))}_{\substack{\text{Kullback-Leibler Divergence} \\ \text{From (choosable) prior} \\ \text{to approximate posterior}}}$$
  - Choice of Prior:
    - $\mathcal{N}(0,1)$ – Unit Gaussian centered at the origin

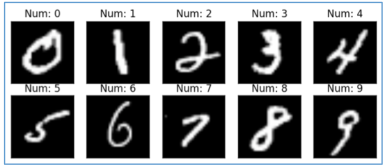
- Anomalous data has both *higher KL-Divergence* and *poorer reconstruction* compared to normal data


0
-2
-4
-6
-6
-4
-2
0
Standard Autoencoder
(direct encoding coordinates)
0
-2
-4
-6
-6
-4
-2
0
μ
σ
Variational Autoencoder
(μ and σ initialize a probability distribution)
σ
μ
Sample
A Standard Variational Autoencoder

Source

Example: MNIST

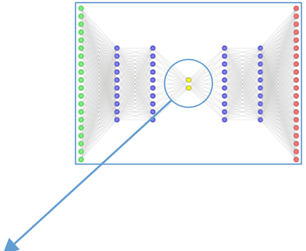
- MNIST – Database of 70,000 images of hand-drawn digits (28x28 pixels)

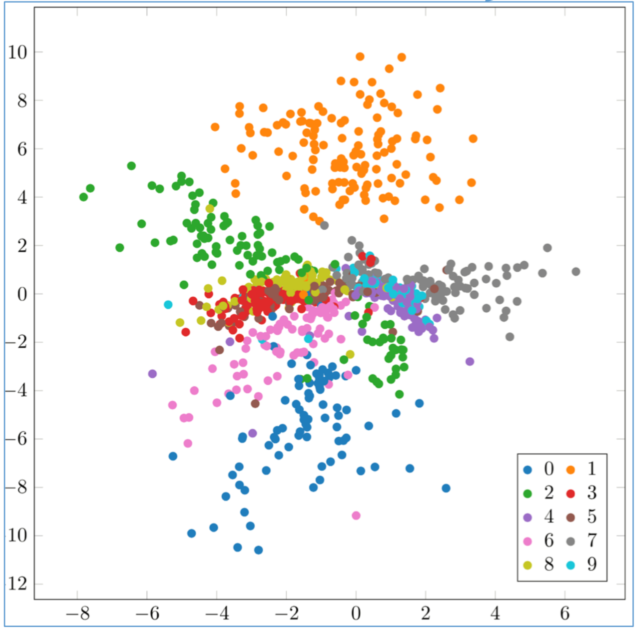


MNIST Examples

- Autoencoder trained by minimizing the *mean-squared-error* between the input/output pixel values
  - Loss Function (MSE): $L(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^{n} |x_i - \hat{x}_i|^2$

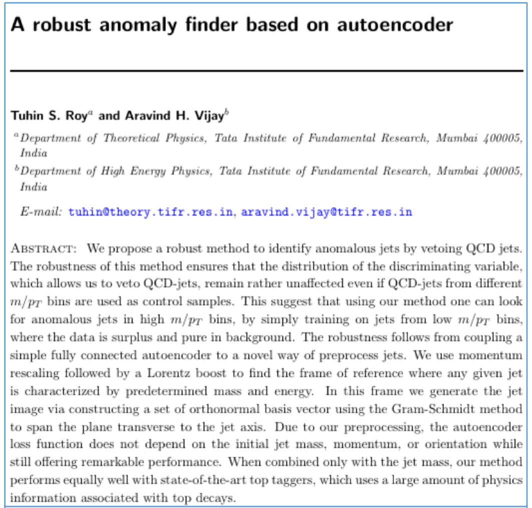
- Result: Different digits are represented by distant vectors in the latent space
  - *Features* have been realized, and separated in the latent space
  - Did not explicitly tell the autoencoder to do this





Representations of classified MNIST digits in a 2-D latent space. Source

Previous studies at LHC (2): https://arxiv.org/pdf/1903.02032.pdf



A robust anomaly finder based on autoencoder

Tuhin S. Roy[a] and Aravind H. Vijay[b]

[a] *Department of Theoretical Physics, Tata Institute of Fundamental Research, Mumbai 400005, India*

[b] *Department of High Energy Physics, Tata Institute of Fundamental Research, Mumbai 400005, India*

E-mail: tuhin@theory.tifr.res.in, aravind.vijay@tifr.res.in

Abstract: We propose a robust method to identify anomalous jets by vetoing QCD jets. The robustness of this method ensures that the distribution of the discriminating variable, which allows us to veto QCD-jets, remain rather unaffected even if QCD-jets from different m/p_T bins are used as control samples. This suggest that using our method one can look for anomalous jets in high m/p_T bins, by simply training on jets from low m/p_T bins, where the data is surplus and pure in background. The robustness follows from coupling a simple fully connected autoencoder to a novel way of preprocess jets. We use momentum rescaling followed by a Lorentz boost to find the frame of reference where any given jet is characterized by predetermined mass and energy. In this frame we generate the jet image via constructing a set of orthonormal basis vector using the Gram-Schmidt method to span the plane transverse to the jet axis. Due to our preprocessing, the autoencoder loss function does not depend on the initial jet mass, momentum, or orientation while still offering remarkable performance. When combined only with the jet mass, our method performs equally well with state-of-the-art top taggers, which uses a large amount of physics information associated with top decays.

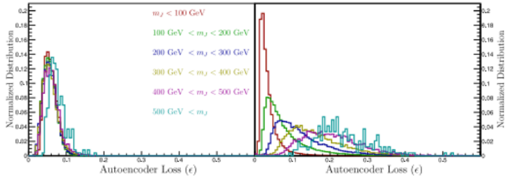
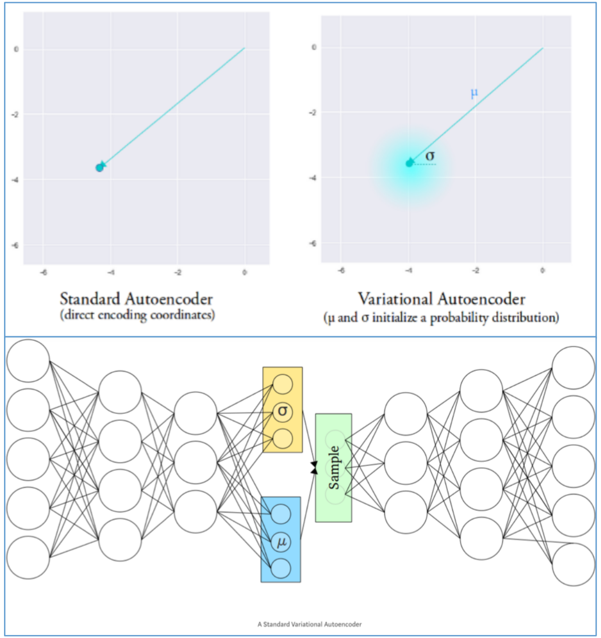


Figure 5. The effect of mass cuts on the autoencoder response (for QCD jets) using our method **(LEFT)** and using dense autoencoder from Ref. [31] **(RIGHT)**.

- Input: Softmax Jet images
  - Jets boosted to predefined E_0, m_0, image made of pts in transverse jet plane
- Architecture: Fully-Connected Layers
- Loss function used: L2 Norm $\epsilon \equiv \sqrt{\sum_i (O_i^K - I_i^1)^2}$
- Main feature: Robustness
  - Softmax jet images produce a loss function which *doesn't vary with jet mass*
  - Can train on low p_T and look for anomalous jets in high p_T

Variational Autoencoders in Practice

- Now, express each latent vector as a *distribution* in the latent space
  - $\boldsymbol{z} \rightarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ – Multivariate Gaussian w/ Diagonal Covariance
- Sample from the latent distribution using the *reparameterization trick*
  - $\hat{\boldsymbol{z}} = \boldsymbol{\mu} + \boldsymbol{\sigma}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} = \mathcal{N}(\mathbf{0}, \mathbf{1})$ is a multivariate unit Gaussian
- The latent distribution is the *approximate posterior* $q(z|x)$
  - Approximating the *true posterior* $p(z|x)$
- The conditional distribution $p(x|z)$ can be inferred from the *reconstruction accuracy*
- For the *prior* $p(z)$, assume a unit Gaussian centered at the origin
  - $p(z) = \mathcal{N}(0, 1)$


0
-2
-4
-6
-6
-4
-2
0
Standard Autoencoder
(direct encoding coordinates)
μ
σ
Variational Autoencoder
(μ and σ initialize a probability distribution)
σ
μ
Sample
A Standard Variational Autoencoder

Source

Bayesian Inference in Variational Autoencoders

- $p(x) = \int p(z)p(x|z)dz$ - Intractable
- Approach: Investigate log likelihood of x
- $\log p(x) = \mathbb{E}_z[\log p(x)]$

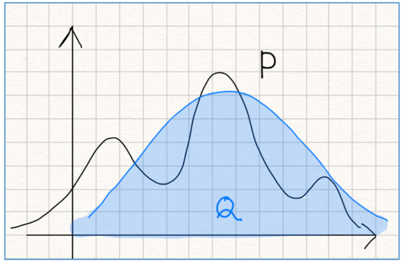
$$= \mathbb{E}_z\left[\log \frac{p(X|Z)p(z)}{p(z|x)}\right]$$ Bayes Theorem

$$= \mathbb{E}_z\left[\log \frac{p(X|Z)p(z)}{p(z|x)} \frac{q(z|x)}{q(z|x)}\right]$$ Multiply by $\frac{q(z|x)}{q(z|x)}$

$$= \mathbb{E}_z[\log p(x|z)] - \mathbb{E}_z\left[\log \frac{q(z|x)}{p(z)}\right] + \mathbb{E}_z\left[\log \frac{q(z|x)}{p(z|x)}\right]$$

Kullback-Leibler Divergence

$$D_{KL}(q(x)||p(x)) = \int q(x) \log\left(\frac{q(x)}{p(x)}\right) dx$$



"The KL divergence is the measure of inefficiency in using the probability distribution *Q* to approximate the true probability distribution *P*." Source

$$\log p(x) = \mathbb{E}_z[\log p(x|z)] - D_{KL}(q(z|x)||p(z)) + \cancel{D_{KL}(q(z|x)||p(z|x))}$$

KL Divergence ≥ 0, so remove this term and define a *lower bound* on the log likelihood

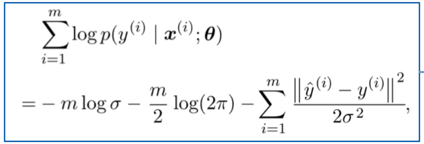
Reconstruction Accuracy

KL Divergence of approximate posterior from gaussian prior

Intractable, since p(z|x) is intractable

Bayesian Inference in Variational Autoencoders

- *Variational lower bound* on the data log-likelihood
  - $\log p(x) \geq \mathcal{L} = \mathbb{E}_z\left[\log p(x|z)\right] - D_{KL}(q(z|x)||p(z))$ ~~$+ D_{KL}(q(z|x)||p(z|x))$~~
  - Goal: Maximize it!
    - $D_{KL}(q(z|x)||p(z|x))$ will approach 0, i.e. $q(z|x) \sim p(z|x)$
    - Performed by using $-\mathcal{L}$ as the *loss function* of the autoencoder
    - Treat first term via maximum likelihood estimation:

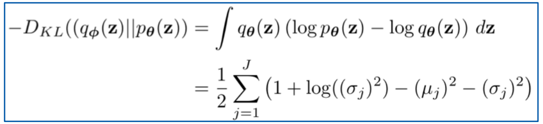


$$\sum_{i=1}^{m} \log p(y^{(i)} \mid \boldsymbol{x}^{(i)}; \boldsymbol{\theta})$$
$$= -m \log \sigma - \frac{m}{2}\log(2\pi) - \sum_{i=1}^{m} \frac{\left\|\hat{y}^{(i)} - y^{(i)}\right\|^2}{2\sigma^2},$$

deeplearningbook.org

Equivalent to minimizing *mean-squared-error* between input and output

- Second term has a nice closed-form solution when q(z|x) and p(z) are Gaussian!



$$-D_{KL}((q_{\boldsymbol{\phi}}(\mathbf{z})||p_{\boldsymbol{\theta}}(\mathbf{z})) = \int q_{\boldsymbol{\theta}}(\mathbf{z})\left(\log p_{\boldsymbol{\theta}}(\mathbf{z}) - \log q_{\boldsymbol{\theta}}(\mathbf{z})\right) d\mathbf{z}$$
$$= \frac{1}{2}\sum_{j=1}^{J}\left(1 + \log((\sigma_j)^2) - (\mu_j)^2 - (\sigma_j)^2\right)$$

Auto-Encoding Variational Bayes