Thesis Presentation

Latent Variable Machine Learning Algorithms: Applications in a Nuclear Physics Experiment

Robert Solli

University of Oslo, Expert Analytics AS

December 15, 2019

Latent Variable Machine Learning Algorithms: Applications in a Nuclear Physics Experiment

2019-12-15

└─Thesis Presentation

Thesis Presentation

Latent Variable Machine Learning Algorithms: Applications in a Nuclear Physics Experiment

Robert Solli

University of Oslo, Expert Analytics AS

December 15, 2019

Outline

1. Introducing the Active Target Time Projection Chamber (AT-TPC)
2. Challenges with traditional analysis of AT-TPC data - and the evangilization of Machine Learning
   (i) Recap of central literature
   (ii) Introducing thesis problem statements
3. An introduction to central Machine Learning concepts
4. The Auto-Encoder neural network
5. Results
6. Summary, Conclusion and Outlook

Outline

1. Introducing the Active Target Time Projection Chamber (AT-TPC)
2. Challenges with traditional analysis of AT-TPC data - and the evangilization of Machine Learning
   (i) Recap of central literature
   (ii) Introducing thesis problem statements
3. An introduction to central Machine Learning concepts
4. The Auto-Encoder neural network
5. Results
6. Summary, Conclusion and Outlook

AT-TPC

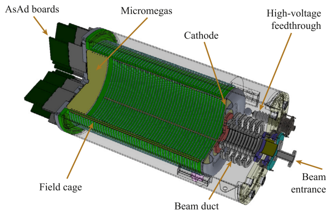
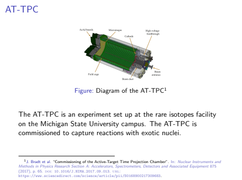


Figure: Diagram of the AT-TPC[1]

The AT-TPC is an experiment set up at the rare isotopes facility on the Michigan State University campus. The AT-TPC is commissioned to capture reactions with exotic nuclei.

[1] J. Bradt et al. "Commissioning of the Active-Target Time Projection Chamber". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 875 (2017), p. 65. DOI: 10.1016/J.NIMA.2017.09.013. URL: https://www.sciencedirect.com/science/article/pii/S0168900217309683.



$1e4$ pads, $1e2$ timebuckets, $1e5$ events/hr: terrabytes of data.

AT-TPC Data

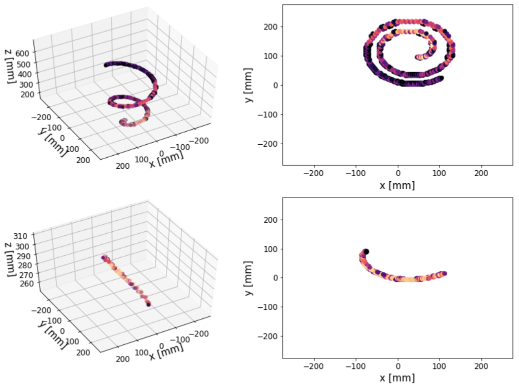
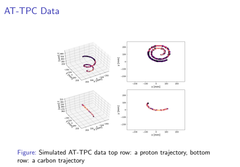


Figure: Simulated AT-TPC data top row: a proton trajectory, bottom row: a carbon trajectory

Latent Variable Machine Learning Algorithms: Applications in a Nuclear Physics Experiment

2019-12-15

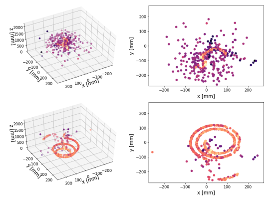
└AT-TPC Data



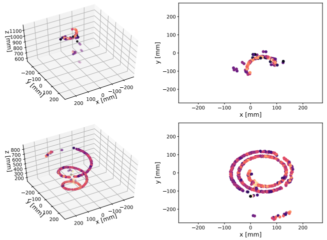
Remember: talk about the plots in detail. Proton on top, carbon on bottom.

Reactions in the whole volume (development of spirals), spirals from magnet.

AT-TPC Data

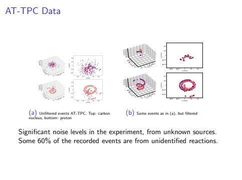


(a) Unfiltered events AT-TPC. Top: carbon nucleus, bottom: proton



(b) Same events as in (a), but filtered

Significant noise levels in the experiment, from unknown sources.
Some 60% of the recorded events are from unidentified reactions.



Remember - talk about event plots in detail.
Not pictured: pure noise events.
More noise in carbon - more ionizing: not necessarily uncorrelated noise

I Expensive integration - for each event a fit is computed.

II Assumptions of the integration technique:

(i) Each event is fit against parameters of the event of interest,
(ii) The integration is sensitive to Noise and Breaks in the tracks.

III In some experiments researchers are unable to identify samples of the positive class of events.

Challenges with the AT-TPC

- I Expensive integration - for each event a fit is computed.
- II Assumptions of the integration technique:
  - (i) Each event is fit against parameters of the event of interest,
  - (ii) The integration is sensitive to Noise and Breaks in the tracks.
- III In some experiments researchers are unable to identify samples of the positive class of events.

The amount of data is also significant: the experiment generates on the order of 10^5 events per hour running.

Idea

Solve the problem by training deep neural networks.

Challenges with the AT-TPC
I Expensive integration - for each event a fit is computed.
II Assumptions of the integration technique:
(i) Each event is fit against parameters of the event of interest,
(ii) The integration is sensitive to Noise and Breaks in the tracks.
III In some experiments researchers are unable to identify samples of the positive class of events.
The amount of data is also significant: the experiment generates on the order of 10^5 events per hour running.
Idea
Solve the problem by training deep neural networks.

Previous Work

- Work on applying ML to this data started with a supervised learning project by Kuchera et al.[2].
- By fine tuning pre-trained networks the authors achieve very impressive performance.

One of the open questions is then, can we segment the events based on reactions without the ground truth labels?

[2] Michelle. P. Kuchera et al. "Machine learning methods for track classification in the AT-TPC". In: *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 940 (2019), p. 156. DOI: 10.1016/j.nima.2019.05.097. URL: https://linkinghub.elsevier.com/retrieve/pii/S0168900219308046.

Previous Work
- Work on applying ML to this data started with a supervised learning project by Kuchera et al.[2].
- By fine tuning pre-trained networks the authors achieve very impressive performance.
One of the open questions is then, can we segment the events based on reactions without the ground truth labels?
[2] Michelle. P. Kuchera et al. "Machine learning methods for track classification in the AT-TPC". In: Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 940 (2019), p. 156. DOI: 10.1016/j.nima.2019.05.097. URL: https://linkinghub.elsevier.com/retrieve/pii/S0168900219308046.

The authors explored a *supervised classification* problem of identifying reactions when ground-truth labels available.

Why machine learning

I Modern computing resources have created a resurgence in ML fostering strong results in other fields.

II Very strong results on image-like data. The resurgence brought Convolutional Neural Networks to the forefront.

III Agnostic models can be applied to source data, possibly avoiding biases from filtering.

IV Broader perspective on what role ML has in (nuclear) physics?

Why machine learning

I Modern computing resources have created a resurgence in ML fostering strong results in other fields.

II Very strong results on image-like data. The resurgence brought Convolutional Neural Networks to the forefront.

III Agnostic models can be applied to source data, possibly avoiding biases from filtering.

IV Broader perspective on what role ML has in (nuclear) physics?

Machine Learning Principles

The ingredients necessary to solve a Machine Learning problem:

(a) a model,

(b) datasets,

(c) and a cost function

Separate between supervised and unsupervised learning.

Machine Learning Principles

The ingredients necessary to solve a Machine Learning problem:

(a) a model,

(b) datasets,

(c) and a cost function

Separate between supervised and unsupervised learning.

Cost function measures the quality of the model.

Can be MSE, crossent.

Training Neural Networks

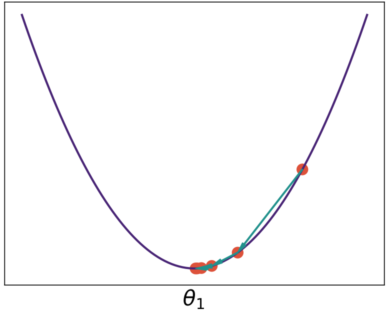


Figure: Gradient descent on a quadratic function

Training neural networks is achieved with stochastic gradient methods. The function to be optimized is the cost: measuring the quality of the current state of the network.

Training Neural Networks

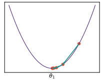


Figure: Gradient descent on a quadratic function

Training neural networks is achieved with stochastic gradient methods. The function to be optimized is the cost: measuring the quality of the current state of the network.

Deep Learning: Neural Networks

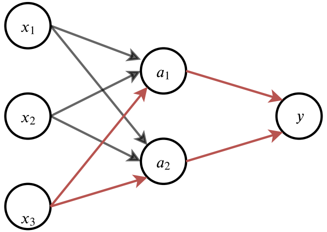
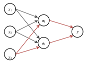


Figure: A neural network with three input nodes, two hidden nodes, and one output node

Output is computed by passing input forwards through the network to the desired output type.

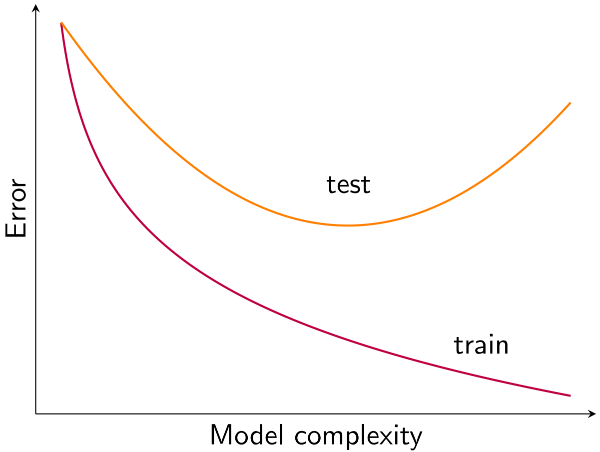


Deep Learning: Neural Networks

Figure: A neural network with three input nodes, two hidden nodes, and one output node

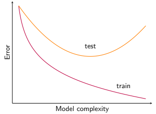
Output is computed by passing input forwards through the network to the desired output type.

Hazards in Training



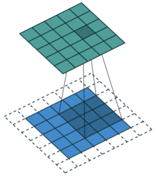
The quality of a model is measured by how well it does on data not seen during training.

Hazards in Training

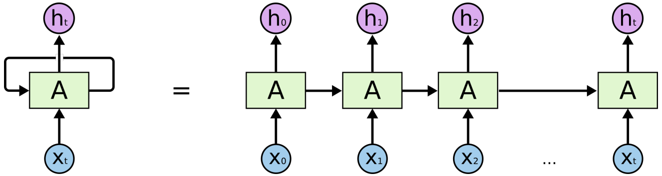


The quality of a model is measured by how well it does on data not seen during training.

Deep Learning: Network Types



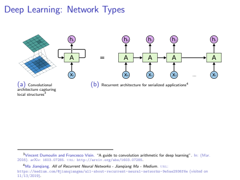
(a) Convolutional architecture capturing local structures[3]



(b) Recurrent architecture for serialized applications[4]

[3] Vincent Dumoulin and Francesco Visin. "A guide to convolution arithmetic for deep learning". In: (Mar. 2016). arXiv: 1603.07285. URL: http://arxiv.org/abs/1603.07285.

[4] Ma Jianqiang. *All of Recurrent Neural Networks - Jianqiang Ma - Medium*. URL: https://medium.com/@jianqiangma/all-about-recurrent-neural-networks-9e5ae2936f6e (visited on 11/13/2019).



Autoencoders

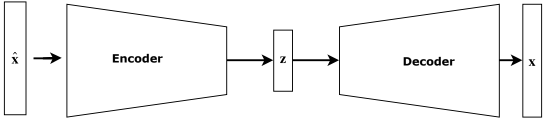


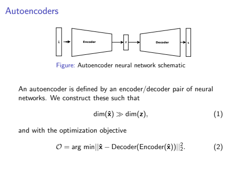
Figure: Autoencoder neural network schematic

An autoencoder is defined by an encoder/decoder pair of neural networks. We construct these such that

$$\dim(\hat{\mathbf{x}}) \gg \dim(\mathbf{z}), \tag{1}$$

and with the optimization objective

$$\mathcal{O} = \arg\min ||\hat{\mathbf{x}} - \text{Decoder}(\text{Encoder}(\hat{\mathbf{x}}))||_2^2. \tag{2}$$



- Recall that we want to separate classes of reaction products
- Additionally - we assume that we have access to very little or no ground truth labelled data

Idea

Learn the distribution over the events through two nonlinear maps which compress and inflate a representation of the events.

Central Hypothesis

If we can create a compressed representation of an event from which we can reconstruct the event - the compression should be informative[5] of the type of event that occurred.

Applied algorithms

Algorithms used from packages:

I Logistic Regression

II K-means clustering

Available architectures in code

I Ordinary Convolutional Autoencoder

II Variational Autoencoder (β, MMD)

III Mixture of Autoencoders

IV Deep Convolutional Embedded Clustering

Applied algorithms
Algorithms used from packages:
I Logistic Regression
II K-means clustering
Available architectures in code
I Ordinary Convolutional Autoencoder
II Variational Autoencoder (β, MMD)
III Mixture of Autoencoders
IV Deep Convolutional Embedded Clustering

Experiment

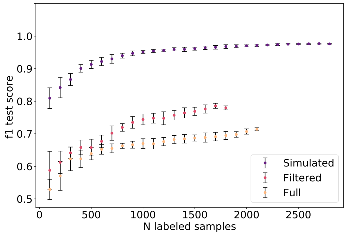
- We chose to represent the data as 2D projections.
- An autoencoder network is then fit.
- After traing, we extract the compressed (or *latent*) expressions.
- With varying amounts of labelled ground-truth data we fit a very simple classifier to the latent expressions.

We wish to construct latent spaces that are as close to trivially separable, and so use a logistic regression classifier.

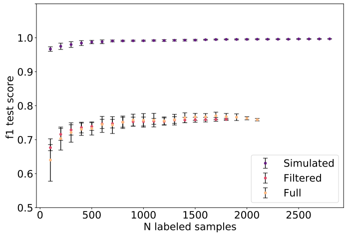
Experiment
- We chose to represent the data as 2D projections.
- An autoencoder network is then fit.
- After traing, we extract the compressed (or *latent*) expressions.
- With varying amounts of labelled ground-truth data we fit a very simple classifier to the latent expressions.
We wish to construct latent spaces that are as close to trivially separable, and so use a logistic regression classifier.

- Neglecting the z-axis.
- autoencoder is fit events in an end to end manner.
- of the data
- With varying amounts of labelled ground-truth data we fit a very simple classifier to the latent expressions

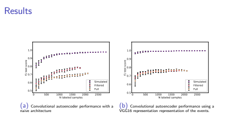
Results



(a) Convolutional autoencoder performance with a naive architecture



(b) Convolutional autoencoder performance using a VGG16 representation representation of the events.



Most important region is the left-most side of the plot, with few samples

Clustering

- We demonstrated that we can construct high quality spaces that are linearly separable
- The next step then is a *clustering* task. Can we separate classes without knowing the ground truth labels?

We explored two autoencoder-based clustering algorithms, in addition to ordinary K-means clustering.

Clustering
- We demonstrated that we can construct high quality spaces that are linearly separable
- The next step then is a *clustering* task. Can we separate classes without knowing the ground truth labels?
We explored two autoencoder-based clustering algorithms, in addition to ordinary K-means clustering.

Clustering, while similar to classification have some principal differences.

K-means clustering

- Ordinary K-means on the two-dimensional representations would obviously not be productive because of the dimensionality of the problem.
- Instead we try to cluster the representation of the events as seen by an image classifying algorithm: VGG16

With $8e3$ output nodes the VGG16 representation is still very much high dimenisonal, but importantly it is sparse.

K-means clustering
- Ordinary K-means on the two-dimensional representations would obviously not be productive because of the dimensionality of the problem.
- Instead we try to cluster the representation of the events as seen by an image classifying algorithm: VGG16

With 8e3 output nodes the VGG16 representation is still very much high dimenisonal, but importantly it is sparse.

K-means results

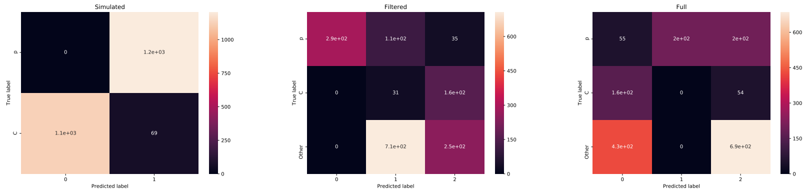
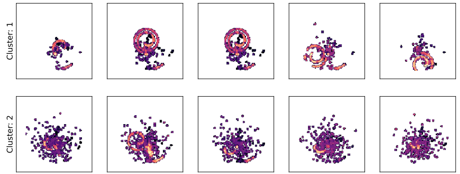
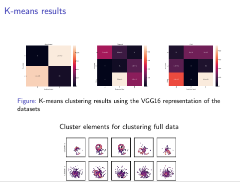


Figure: K-means clustering results using the VGG16 representation of the datasets

Cluster elements for clustering full data



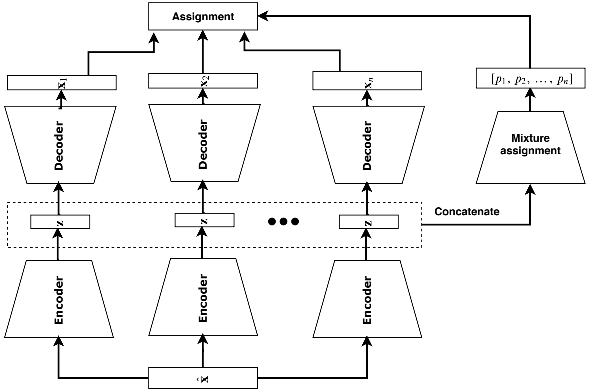


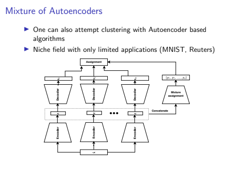
Clusters are well defined if they have one cell along the vertical axis.

And perfect if they are empty in a cross

Mixture of Autoencoders

- One can also attempt clustering with Autoencoder based algorithms
- Niche field with only limited applications (MNIST, Reuters)





Deep Clustering Results

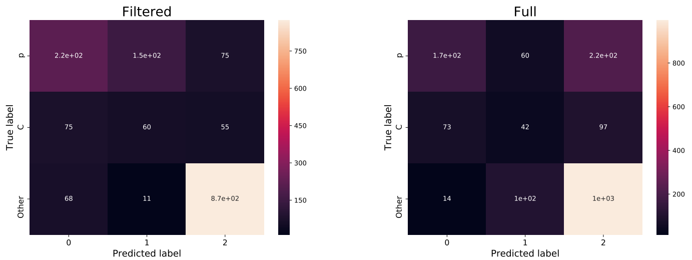
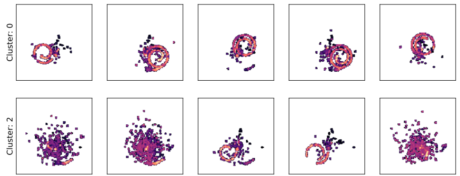
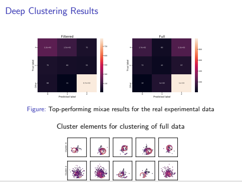


Figure: Top-performing mixae results for the real experimental data

Cluster elements for clustering of full data





Plagued by instability even with the same hyperparameters Remember to link the plots together

Summary and Outlook

- We have shown that Autoencoder models are suitable for both semi-supervised and unsupervised applications.
- Autoencoder clustering has a fundamental challenge with issues of stability and convergence. We still need labelled samples to verify.
- Including physical parameters increases the quality of the latent space.

Neural network models clearly have a place in this analyisis going forward: but connecting the anlysis to the physical properties of the system remains a challenge to solve.

Summary and Outlook

- We have shown that Autoencoder models are suitable for both semi-supervised and unsupervised applications.
- Autoencoder clustering has a fundamental challenge with issues of stability and convergence. We still need labelled samples to verify.
- Including physical parameters increases the quality of the latent space.

Neural network models clearly have a place in this analyisis going forward: but connecting the anlysis to the physical properties of the system remains a challenge to solve.

AT-TPC pad plane

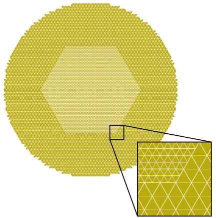


Figure: Detector pad plane of the AT-TPC[6]

Each triangle represents spatial discrete regions of the detector surface. The pad-plane consists of some 10^4 sensor pads on a circle with $r = 29cm$

[6] Bradt et al., "Commissioning of the Active-Target Time Projection Chamber".

AT-TPC pad plane

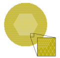



Figure: Detector pad plane of the AT-TPC[6]
Each triangle represents spatial discrete regions of the detector surface. The pad-plane consists of some 10^4 sensor pads on a circle with $r = 29cm$
[6] Bradt et al., "Commissioning of the Active-Target Time Projection Chamber".

f1 score

$$\text{recall} = \frac{TP}{TP + FP}, \quad \text{precision} = \frac{TP}{TP + FN}. \tag{3}$$

The $f1$ score is defined as the harmonic mean of precision and recall for each class. Formally we define it as

$$f1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \tag{4}$$

f1 score

$$\text{recall} = \frac{TP}{TP + FP}, \quad \text{precision} = \frac{TP}{TP + FN}. \tag{3}$$

The $f1$ score is defined as the harmonic mean of precision and recall for each class. Formally we define it as

$$f1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \tag{4}$$

Bias-variance decomposition

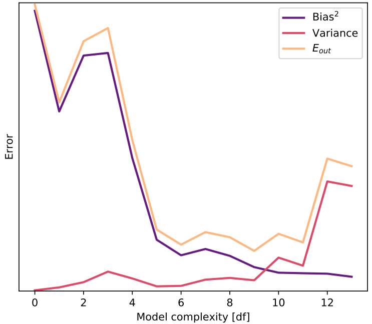


Figure: Bias - variance decomposition of the MSE objective with iid. noise. Generalization error is denoted E_{out}, and is the sum of the Bias and Variance terms.

The quality of a model is measured by how well it does on data not seen during training, measured by E_{out}.

Bias-variance decomposition

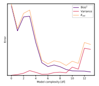


Figure: Bias - variance decomposition of the MSE objective with iid. noise. Generalization error is denoted E_{out}, and is the sum of the Bias and Variance terms.

The quality of a model is measured by how well it does on data not seen during training, measured by E_{out}.

Implementing details

Models implemented in Python with TensorFlow (TF), a ML framework developed by Google. TensorFlow provided a base on which to build complex NN models. Code written for this analyis includes

I Abstraction for autoencoder networks,

II Implementation of variations on the Variational Autoencoder,

III Custom class implementations based on the high-level Keras API for deep clustering.

Implementing details

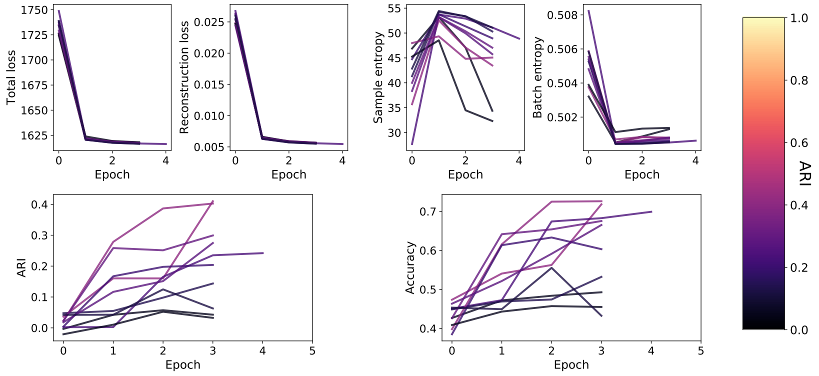
Models implemented in Python with TensorFlow (TF), a ML framework developed by Google. TensorFlow provided a base on which to build complex NN models. Code written for this analyis includes

I Abstraction for autoencoder networks,

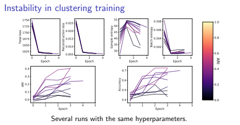
II Implementation of variations on the Variational Autoencoder,

III Custom class implementations based on the high-level Keras API for deep clustering.

Instability in clustering training



Several runs with the same hyperparameters.



Perhaps the most curious result, not only the instability but the high Sample Entropy.