



THE UNIVERSITY *of* EDINBURGH
School of Physics
and Astronomy

Deep Learning & the Higgs Boson

Dr. Liza Mijović



Deep Learning & the Higgs Boson

Classification with Fully Connected and Adversarial Networks.

- **Lecture1: The Higgs boson and event classification:**

- Event classification with a fully connected neural network (NN) with Keras API.

- **Lecture2: Solving the background sculpting challenge:**

- Event classification with adversarial neural network (ANN).
- Hands-on knowledge of manipulating neural networks in Tensorflow.

- **Lecture3: Putting it all together:**

- Compare ANN classification performance to the fully connected network.



Lecture2: classification with adversarial neural network

- **What were we doing in Lecture1?**

Classification with fully connected neural network.

- **What is the issue with classification from Lecture1?**

The discriminant has an undesired bias.

- **How do we solve this issue?**

Classification with adversarial neural network (ANN).



Any problems during lecture1?

When running:

```
conda create --name DeepLearn23 --file requirements.txt  
conda could not resolve the environment.
```

```
13 lines (12 sloc) | 389 Bytes
```

```
1 # This file may be used to create an environment using:  
2 # $ conda create --name <env> --file <this file>  
3 # platform: linux-64  
4 jupyter=1.0.0=py39h06a4308_7  
5 keras=2.4.3=hd3eb1b0_0  
6 matplotlib=3.5.1=py39h06a4308_1  
7 numpy=1.21.5=py39he7a7128_1  
8 pandas=1.4.2=py39h295c915_0  
9 python=3.9.12=h12debd9_0  
10 scikit-learn=1.0.2=py39h51133e4_1  
11 scipy=1.7.3=py39hc147768_0  
12 tensorflow=2.4.1=mkl_py39h4683426_0
```

Solution: remove package versions. i.e jupyter=1.0.0=py39h06a4308_7 becomes jupyter etc.

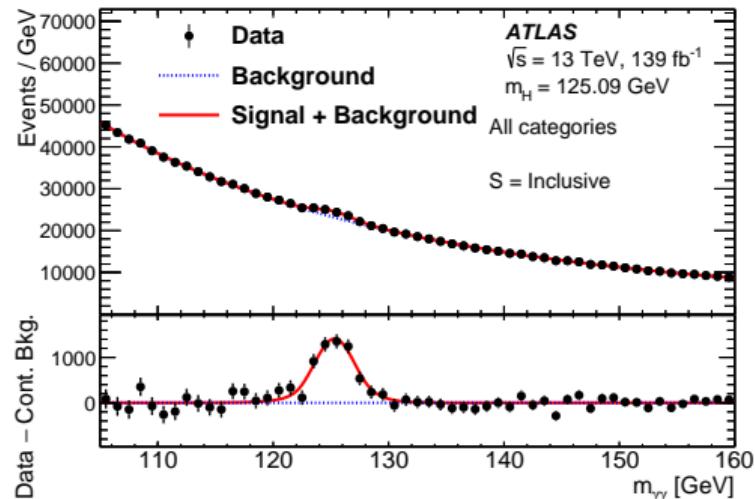
Reminder: Our Challenge

Classification: separate

- **Signal** with Higgs boson.
- **Background** with no Higgs boson.

Approach:

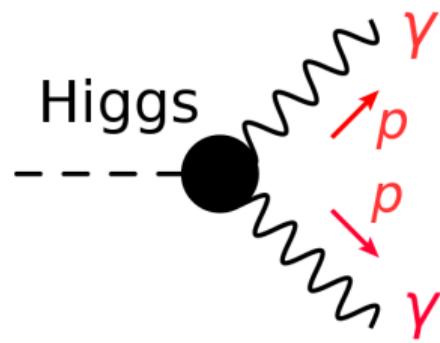
- use synthetic data.
- Introduce no bumps in the $m_{\gamma\gamma}$ distribution; these would hamper the background estimate.
⇒ this lecture.



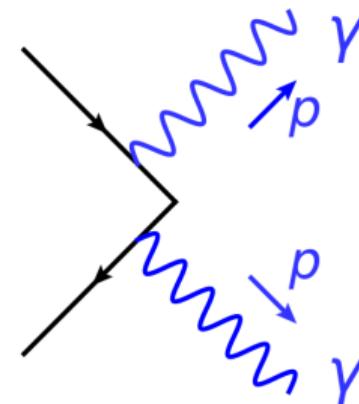
Reminder: What is in the data?

Two photons (γ) per event, with momenta p .

Signal, label=1



Background, label=0



Reminder: What is in the data?

- Momenta p are 4-dimensional (Lorentz) vectors.
- They are passed in cylindrical coordinates.

1st photon	2nd photon	mass	S or B?
,pt_y1,eta_y1,phi_y1,e_y1,	pt_y2,eta_y2,phi_y2,e_y2	, myy,	label
0,62.2385,1.2206,2.02509,114.652,	58.7416,0.879753,-1.23967,82.9781,	122.464,	0

1,69.5362,-1.0435,-2.29563,110.958,	48.7077,-2.01968,1.48524,186.759,	125.308,	1
-------------------------------------	-----------------------------------	----------	---

Lecture1: Classification

Fully connected deep neural network.

Inputs:

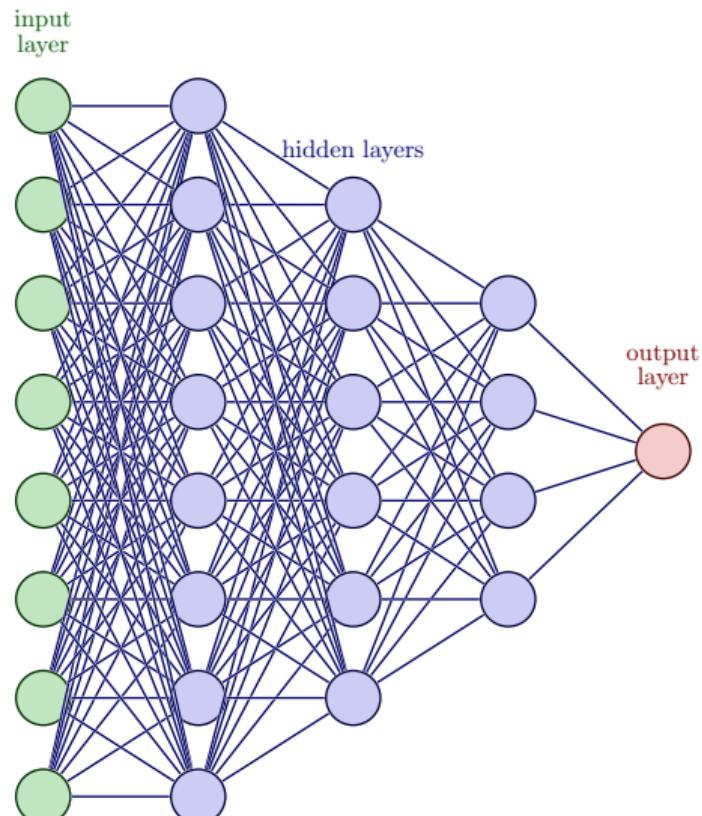
- Features x : photon 4-vectors.
- Labels y : signal (1) or background (0).

Training:

- Combine 8 inputs into 1 classifier.
- Objective: minimise classifier loss L_{clf} .
- Training determines node weights θ_{clf} .

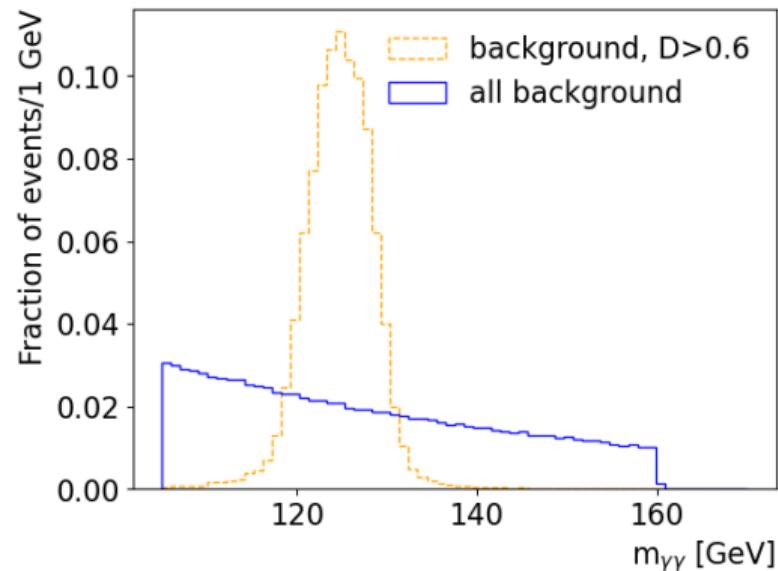
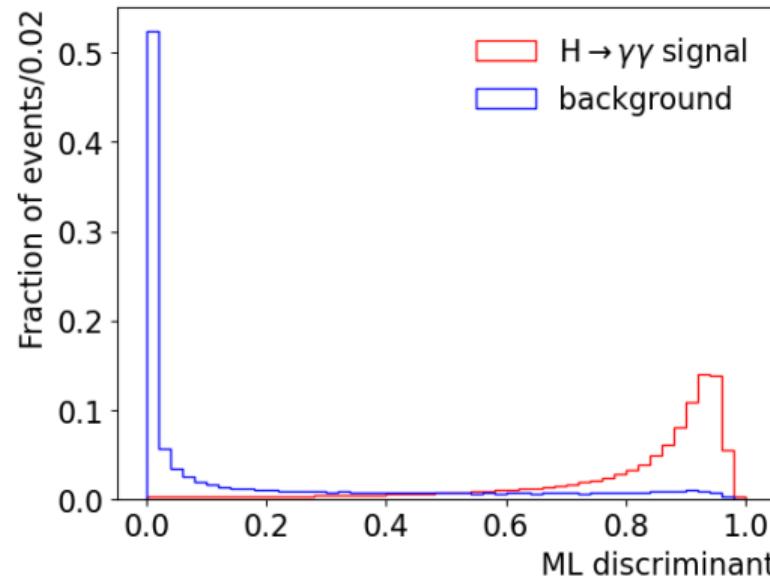
Output:

- discriminant: $z = p_{clf}(y|x, \theta_{clf})$.



Lecture1: Classification Results

We can separate **Signal** from **Background**.

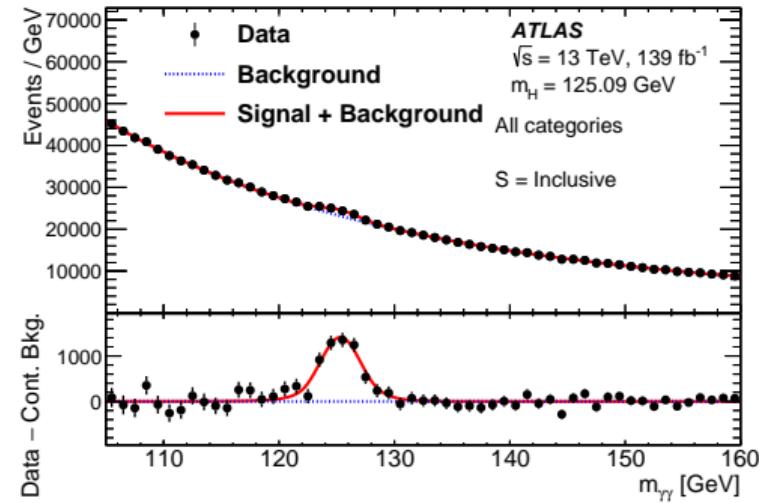
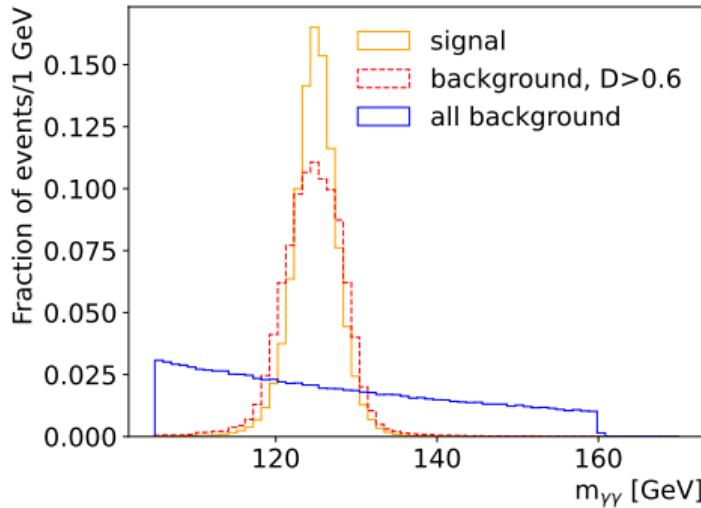


However: **Background** grows a bump at **high discriminant values**.

Lecture1: Classification Issue

We can separate **Signal** from **Background**.

However: **Background** grows a bump at **high discriminant values**.



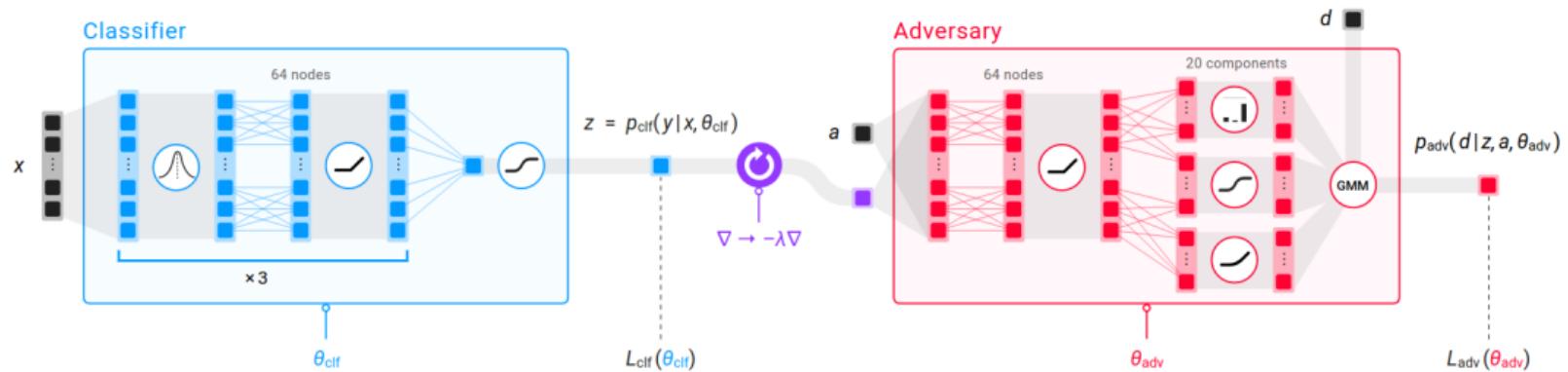
This would prevent us from reliably estimating the background.

Can we design classification which does not sculpt $m_{\gamma\gamma}$?

Adversarial Neural Network

Can we design classification which does not sculpt $m_{\gamma\gamma}$?

- Pit **classifier** network against **adversary**.
- **Classifier** tries to guess event label y (0 or 1) from inputs x .
- **Adversary** tries to guess $m_{\gamma\gamma}$ from classifier output.
- If possible, the **classifier** is penalised.

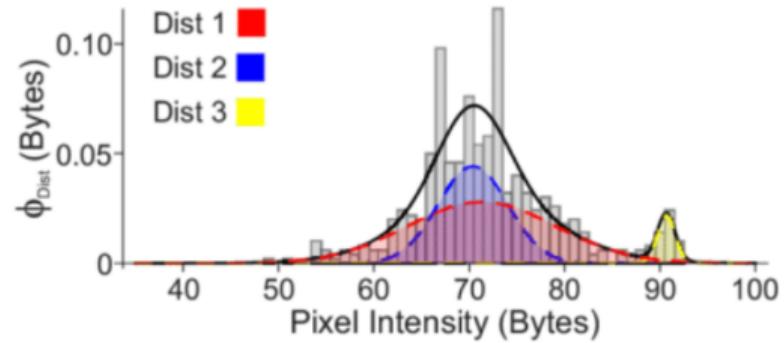
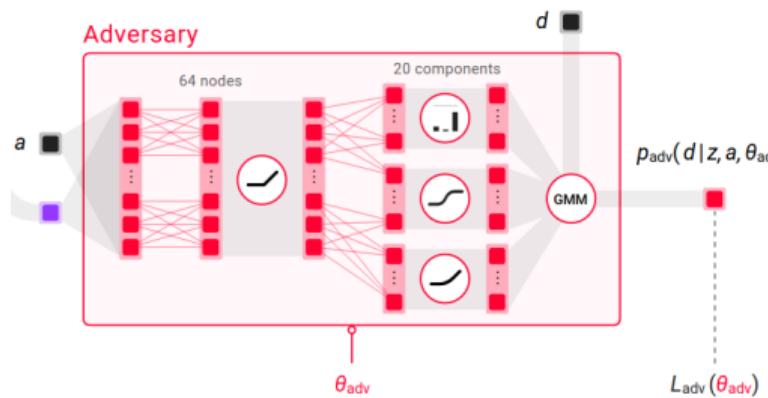


Gaussian Mixture Model

- Adversary tries to guess $m_{\gamma\gamma}$ from classifier output.
- If possible, the classifier is penalised.

Q: How do we know if the adversary is guessing the $m_{\gamma\gamma}$?

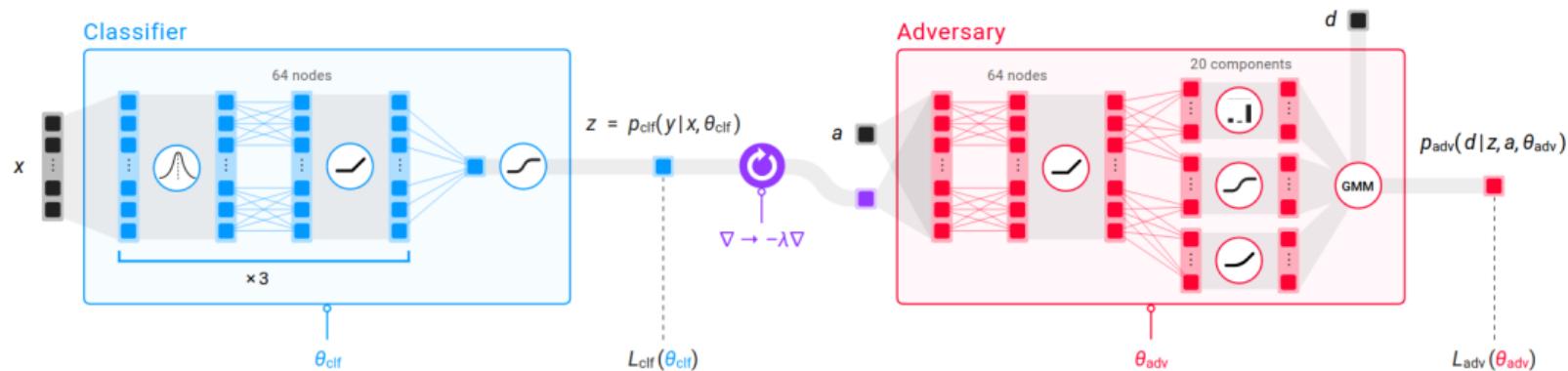
A: Gaussian Mixture Model (GMM).



Mass decorrelation: gradient reversal

Can we design classification which does not sculpt $m_{\gamma\gamma}$?

- **Adversary:** parametrises $d = m_{\gamma\gamma}$ conditional on classifier output; $p_{adv}(m_{\gamma\gamma}|z)$.
- Trained with: **adversary loss** $L_{adv}(\theta_{adv})$.
- Gradient minimising L_{adv} is back-propagated to **classifier**: gradient reversal layer.

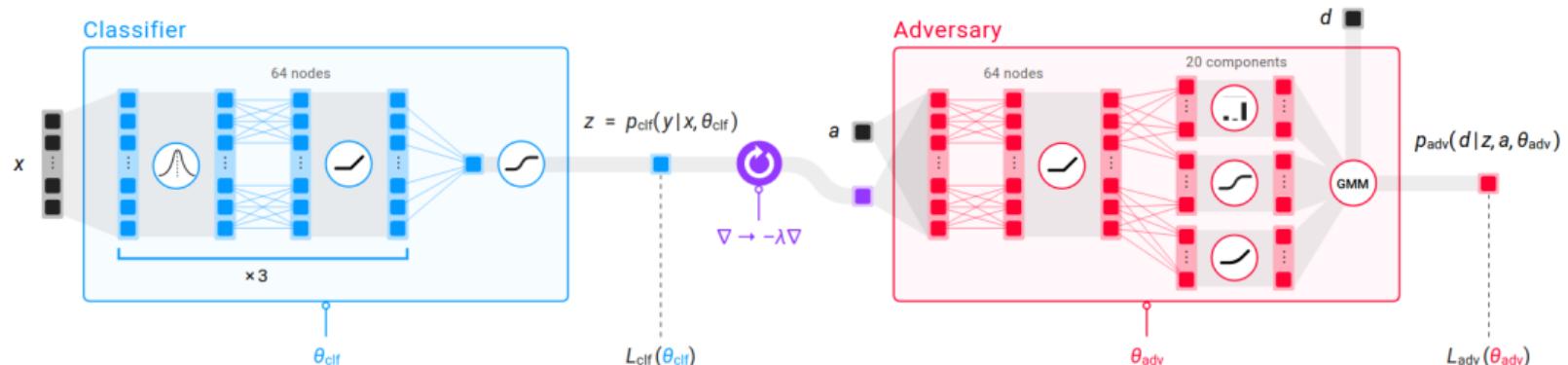


Mass decorrelation: gradient reversal

- Both networks trained simultaneously with a loss:

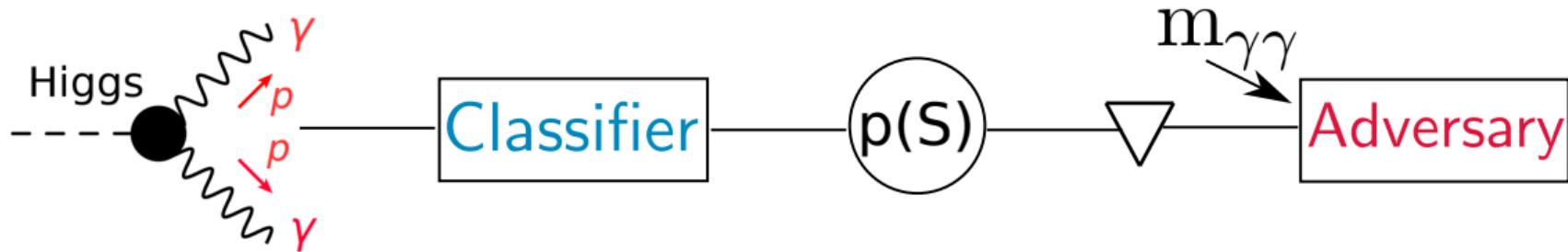
$$L = L_{clf}(\theta_{clf}) - \lambda L_{adv}(\theta_{clf}, \theta_{adv})$$

- Classifier:** tries to guess event label (y = signal or background).
- Adversary:** tries to guess $d = m_{\gamma\gamma}$.
- Trade-off controlled by parameter λ .

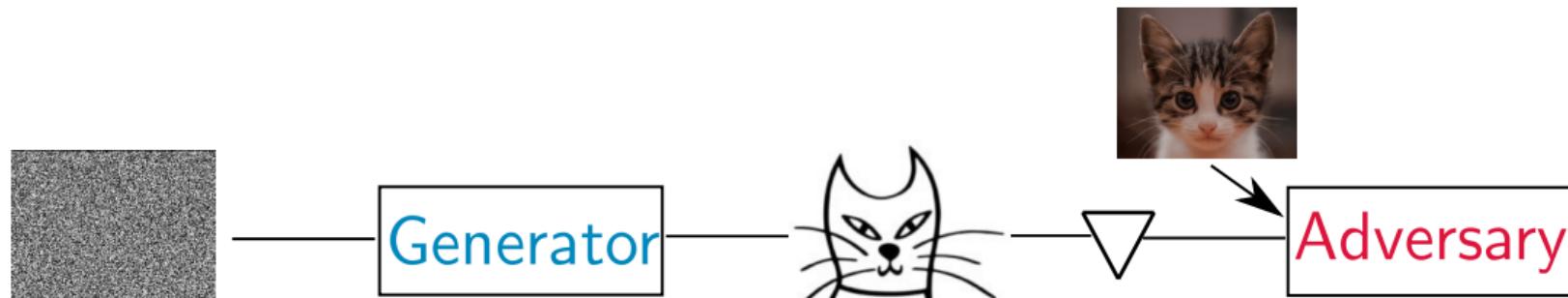


Aside: Generative adversarial network

Our Lecture: Adversarial Neural Network (ANN). Higgs classification.



Computer vision: Generative Adversarial Network (GAN). Synthetic image generation.



Hands-on work

Classification with fully connected network (NN) & adversarial network (ANN):

Using data_200k.csv:

- Run [git repository](#): code/final/ann_classification.ipynb
- Look through the notebook; do you understand how the ANN works?
- Share your understanding:
 - ANN concepts: <https://cern.ch/lect2survey1>.
 - Advanced ANN concepts: <https://cern.ch/lect2survey2>.

Optional: after lecture:

- run ann_classification.ipynb over data_2M.csv (~ 3h running time on laptop).

Reminder:

- Download the data: <https://cern.ch/dl23data>
- Set up the environment: <https://cern.ch/dl23code>

Any feedback? <https://cern.ch/lect2feed>

Extra



Correlation

$m_{\gamma\gamma}$ is not used as input feature to fully connected network (NN) classifier.

Q: Why does the background get sculpted?

A: Some of the input features are correlated to $m_{\gamma\gamma}$.

