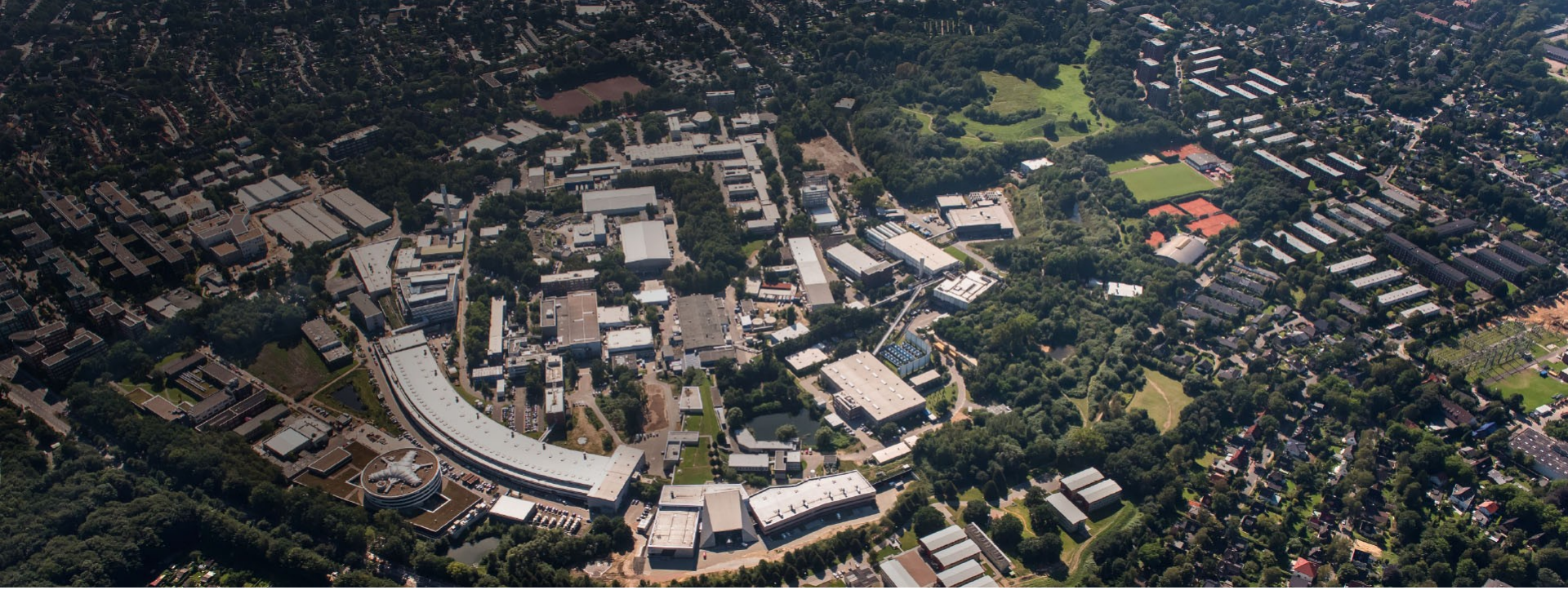


Generative Adversarial Network in the search for SUSY in events with one lepton and multiple jets in proton-proton collisions

Final Report

Summer Student: Farouk Mokhtar

Supervised by: Dirk Kruecker, Ashraf Mohamed

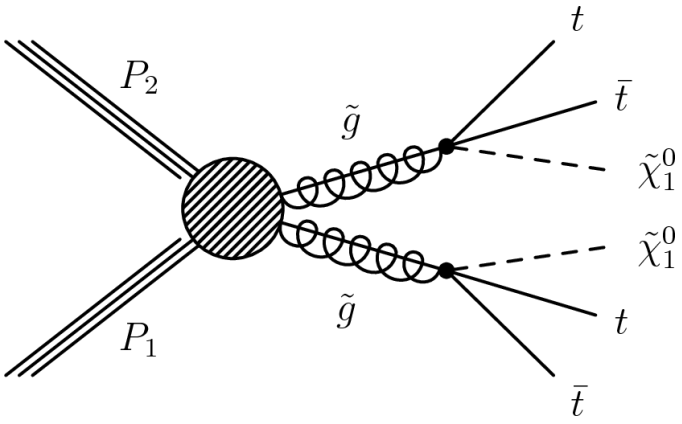
HELMHOLTZ SPITZENFORSCHUNG FÜR GROSSE HERAUSFORDERUNGEN

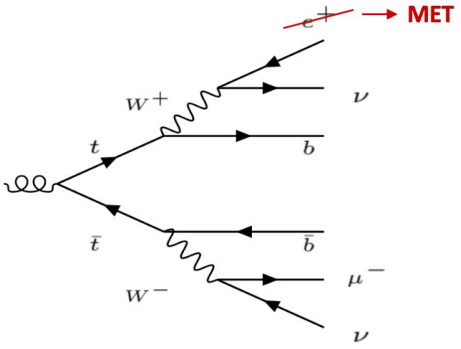
DESY.

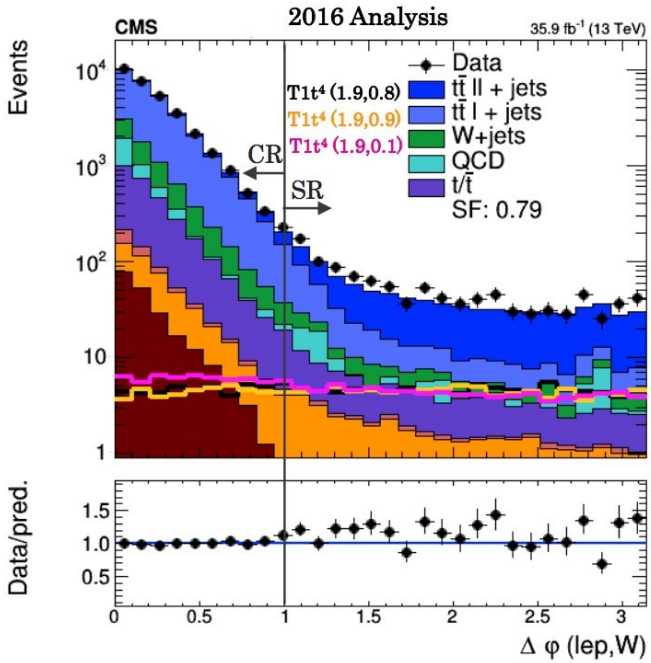
Physics Problem: SUSY 1Lep

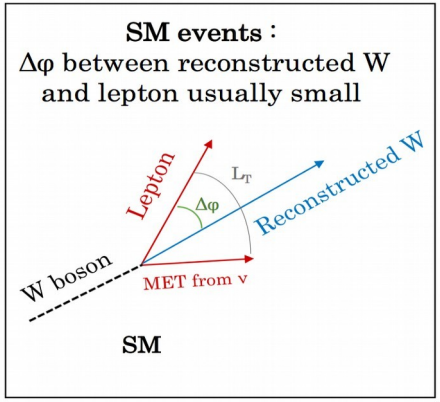
Search Variables
L_T =|Lep p_T| + |MET|
H_T = Sum(Jet p_T)
nJets
Nbjets
Δφ

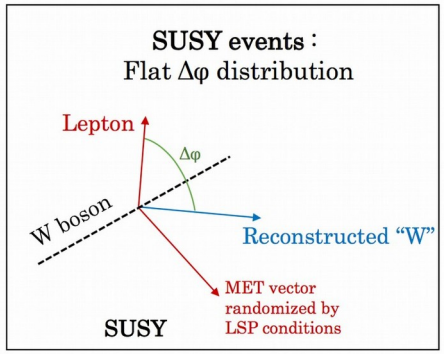
Main Background
DiLep Ttbar
SemiLep Ttbar
Wjets



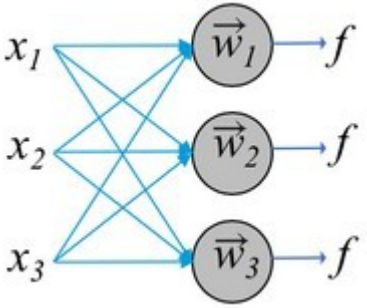






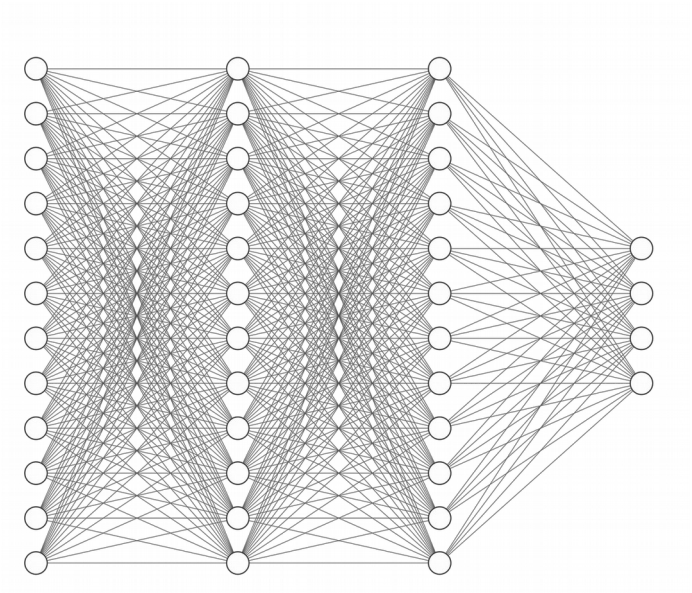


What is a NN?



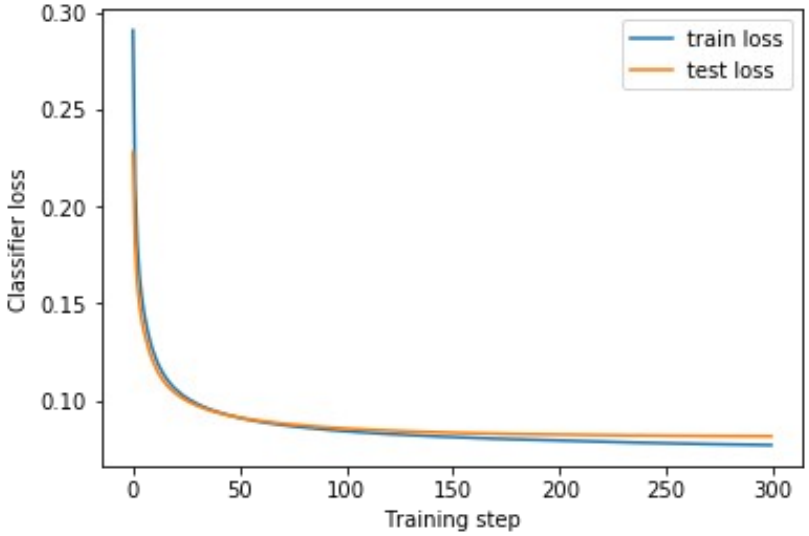
$$f(\sum_{k=1}^{n} w_k x_k + b)$$

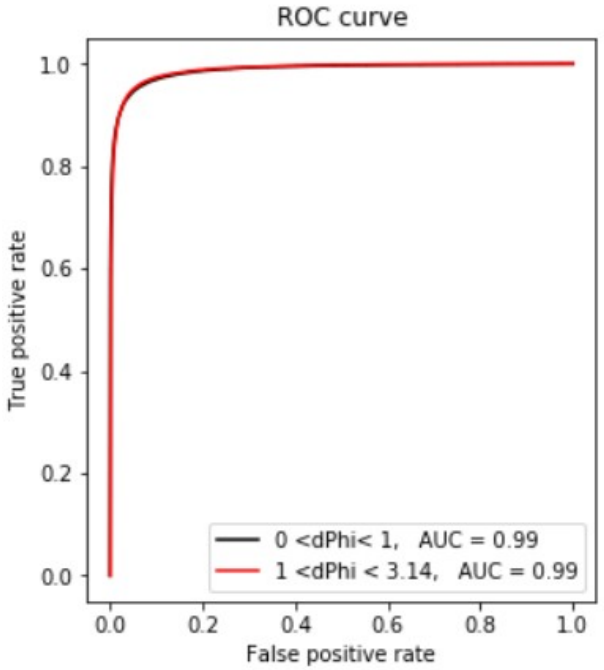
- Composed of layers of neurons
- Takes input and produces output
- **Loss function**: translates the output → scalar (called loss)
- For **sig/bkg classification**, we use **BCE** as Loss function
- **Goal of training** → minimize the loss by adjusting all the weights and biases

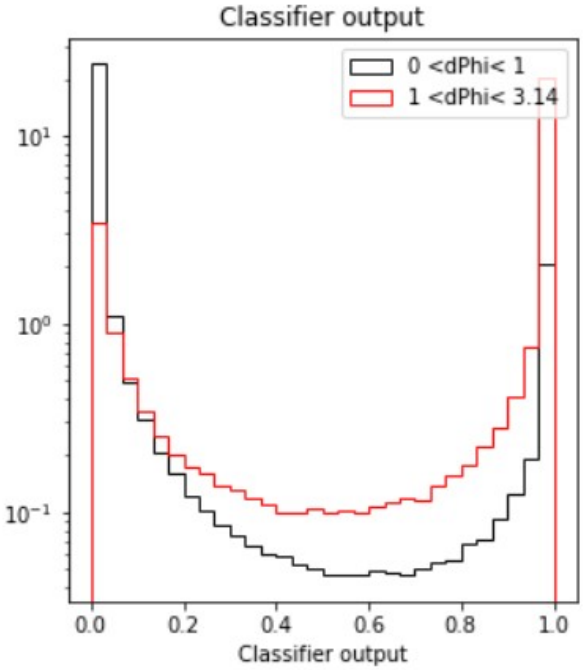


I have a working Classifier

- ✔ Large area under the ROC curve
- ✔ Small Δφ events are classified as more as bkg
- ✔ Peaks near 0 & 1 for *clf* output





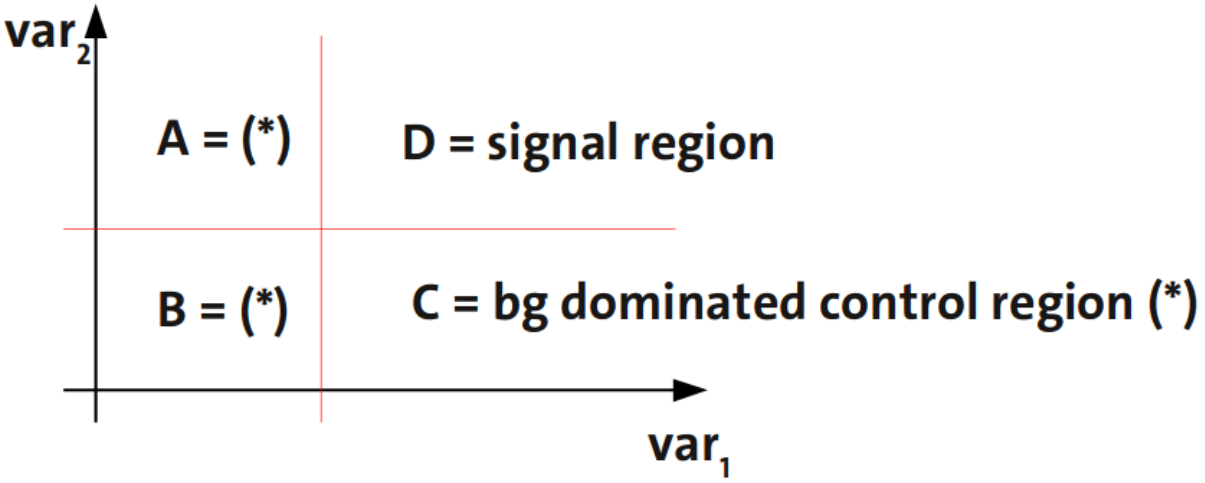


Project Description

- Use **Adversarial Training** to classify sig/bkg events
- **Why**? To decorrelate the relation between the *clf* output and Δφ
- **Why**? To use *Data Driven Background Estimation~ABCD method*

ABCD

If variables var_1 & var_2 are decorrelated → background in signal region is predicted by the ratio: $\mathbf{N_D = (N_C * N_A) / N_B}$



Adversarial Training

What is a GAN?

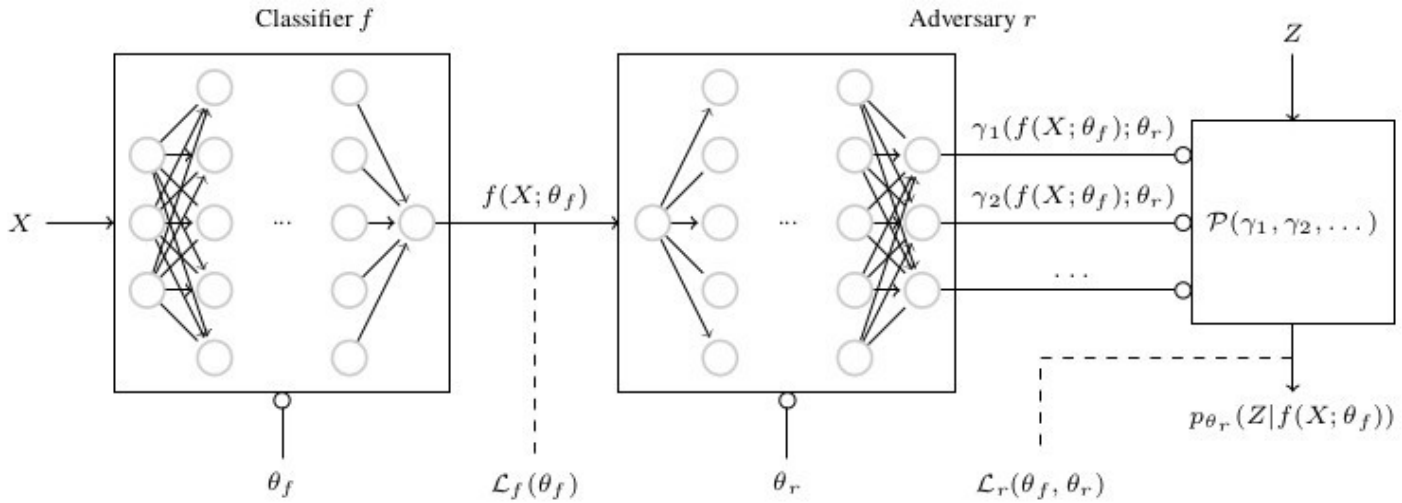
- **Generative Adversarial Network**: 2 networks in contest with each other

  (1) Classifier *clf* and (2) Adversary

- Training is done simultaneously

- **Goal of Adversary** → confuse the *clf* and provide feedback until the *clf* output is independent of the nuisance parameter *Z*

✗ This decorrelation comes at the expense of classification efficiency



My Network's Architecture

$$loss = clf_{loss} - \lambda * adv_{loss}$$

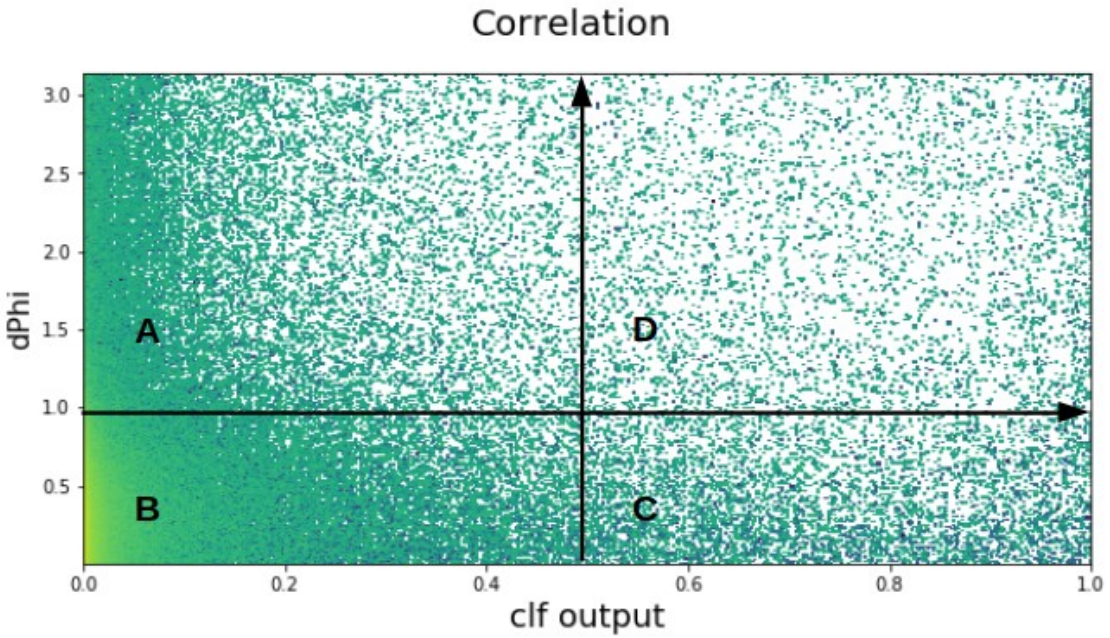
Network	Classifier	Adversary
# of inputs	17	1
# of outputs	1	1
Hidden Layers (2)	125 neurons each	50 neurons each
Activation Function	reLu	
Output Activation Function	sigmoid	blank~normal
Loss	BCE	MSE
Learning Rate	0.001	0.005
Optimizer	Adam	

$$loss = clf_{loss} - \lambda * adv_{loss}$$

AIM:

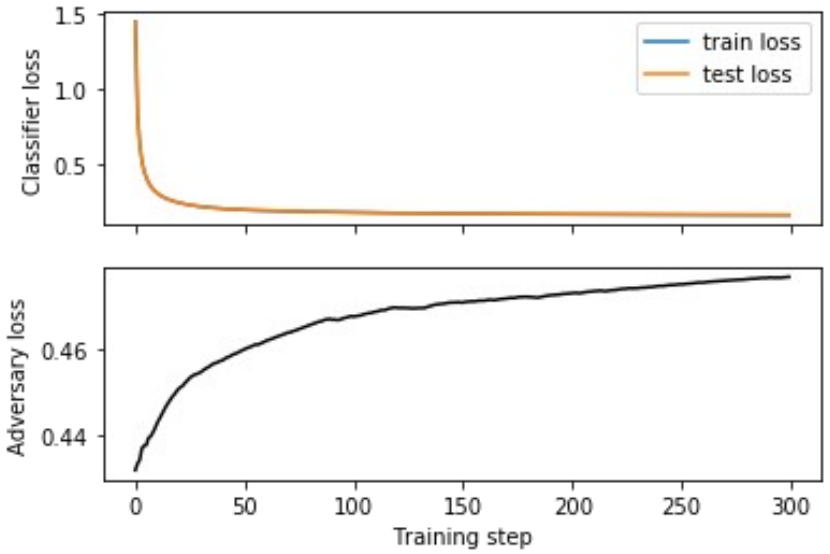
Find the optimal value of λ which makes $N_A/N_B=N_D/N_C$ → **Ratio = 1**

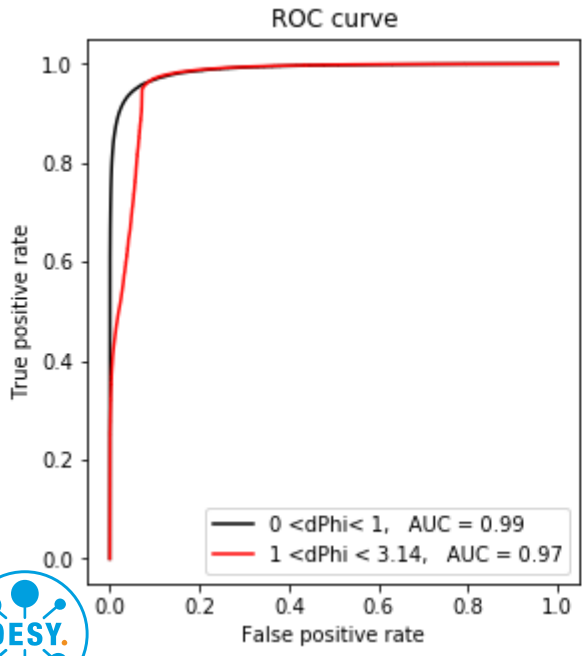
- We **only use bkg** events to get the ratio
- We weigh in each event by the parameter **Finalweight**, which includes: cross-section weight and all other corrections

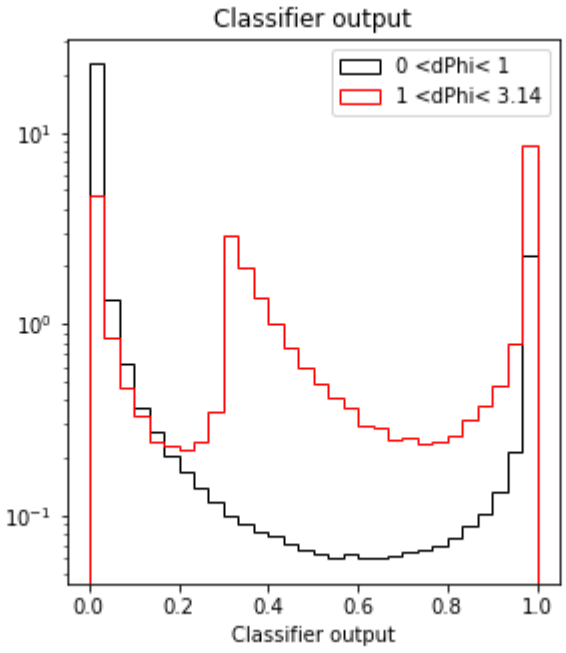


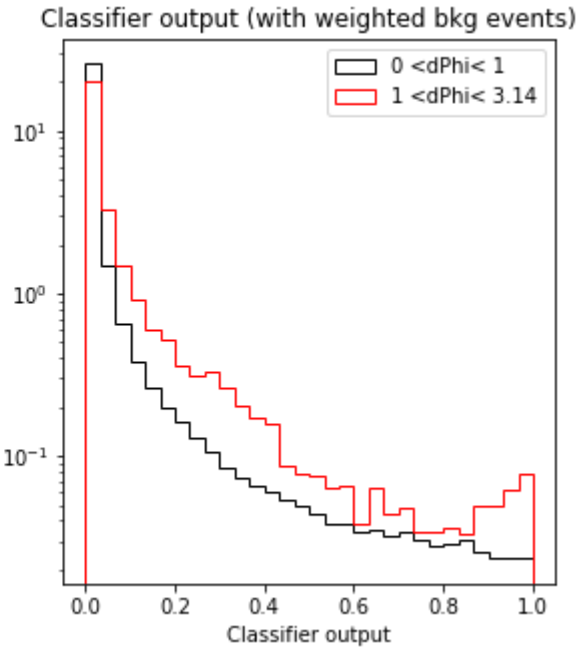
λ=0.85

- ✔ Still: Large area under the ROC curve
- ✔ Still: Small Δφ events are classified more as bkg
- ✗ No longer good classification for high Δφ
- ✔ Clf output is getting **more independent** of Δφ





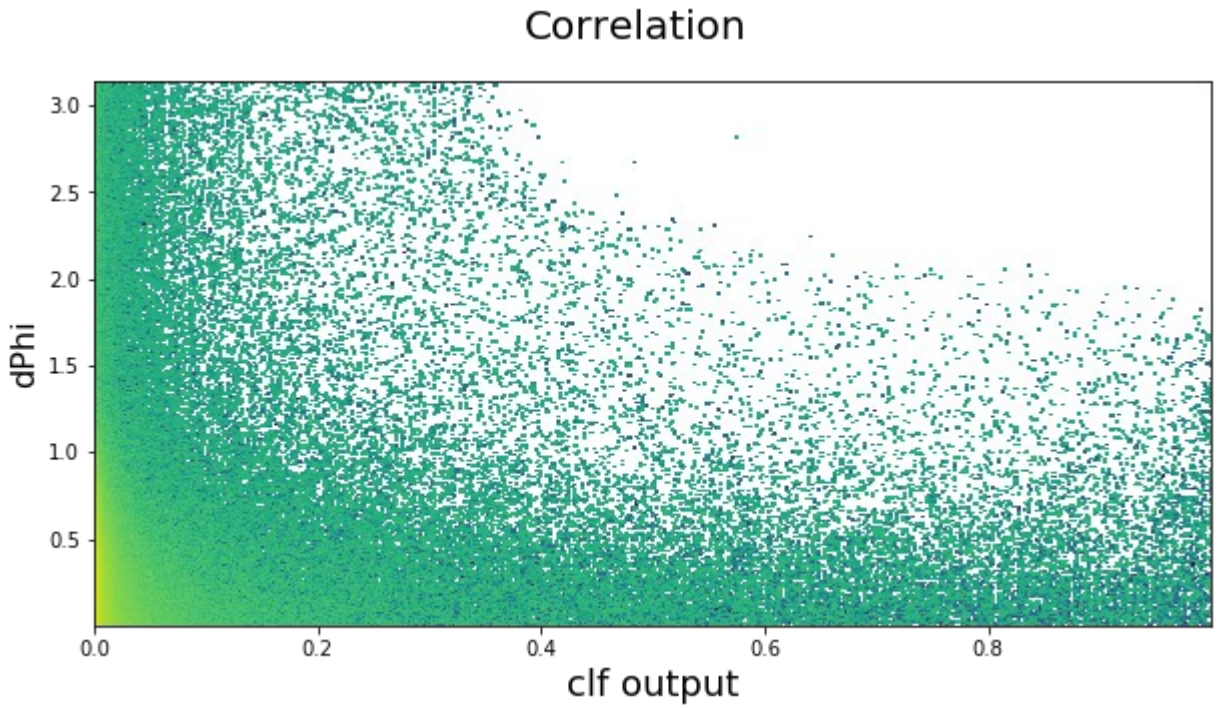




λ=0.85

```
Classifier Output:
A = [0.13270593]
B = [4.87374984]
C = [0.08086097]
D = [0.00173608]
Ratio = [1.26822195]
```

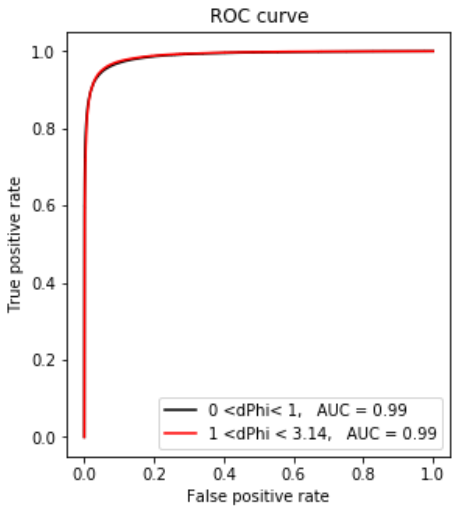
Ratio ≈ 1

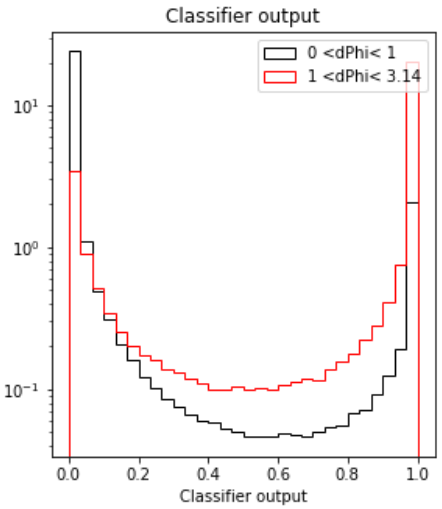


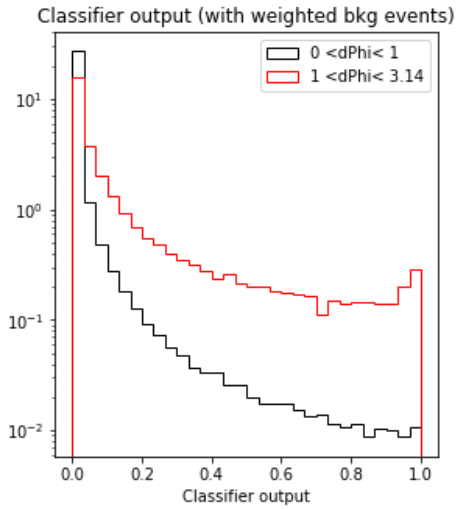
- ✔ **Ratio is close to 1**

For Comparison

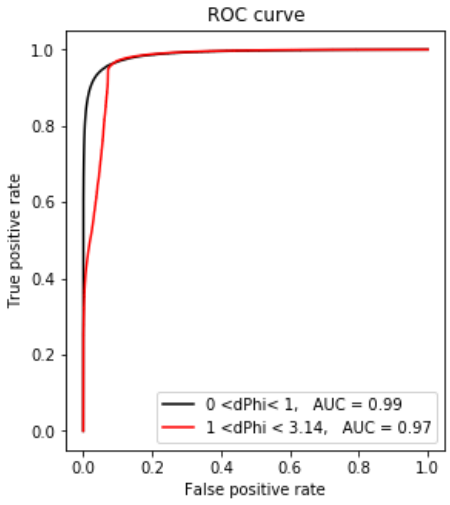
Normal Classifier

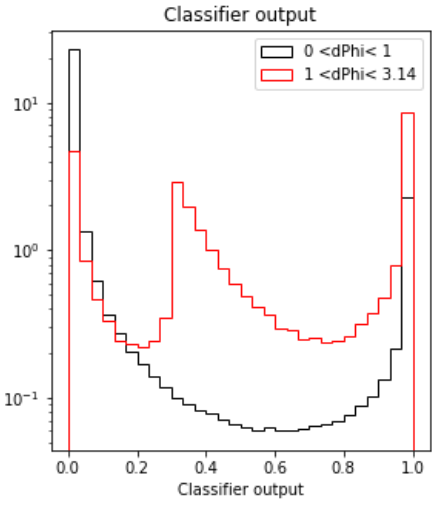


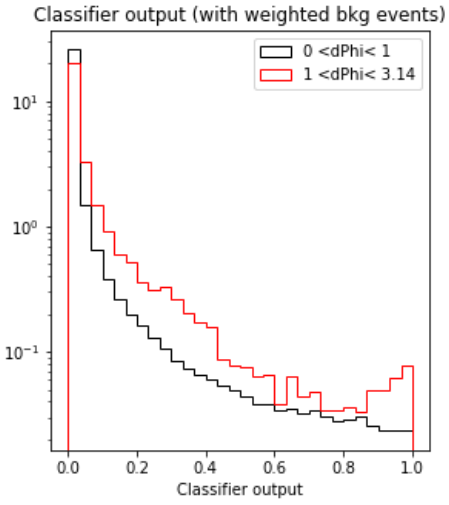




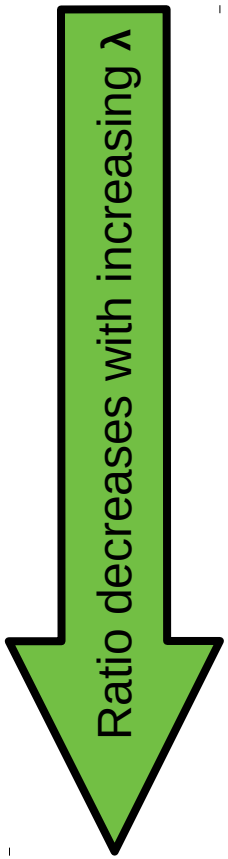
Adversarial







Results for different values of λ



λ	Ratio	AUC	Time	# of epochs
0 (normal classifier)	0.06624977	0.99	9 min	300
0.5	2.60919688	0.99	32 min	300
0.75	1.4703602	0.98	32 min	300
0.85	1.26822195	0.98	32 min	300
1	0.62123818	0.97	32 min	300
10	0.06472567	0.99	32 min	300

Summary

Achieved so far

- Found optimal values of λ which provide the decorrelation at the expense of small efficiency loss

Next step

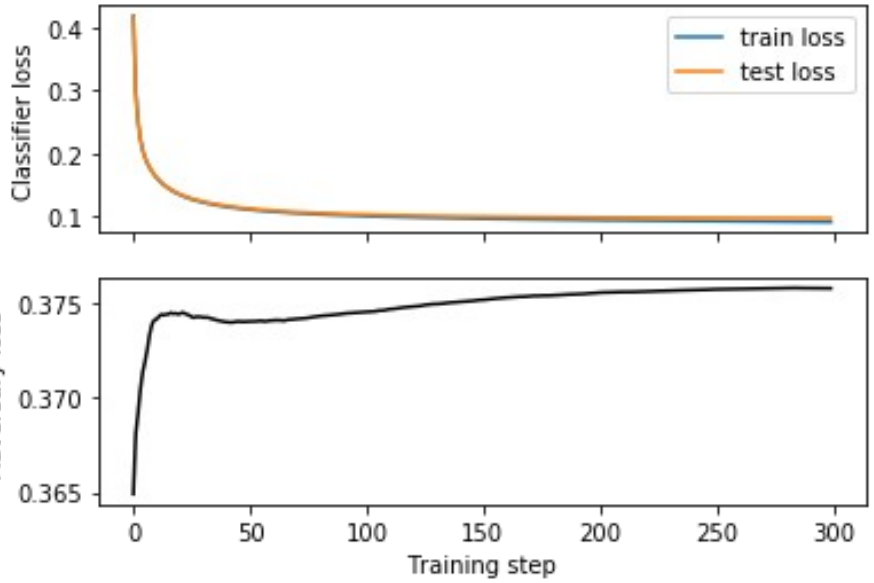
- Use the network on real data to predict bkg in signal region

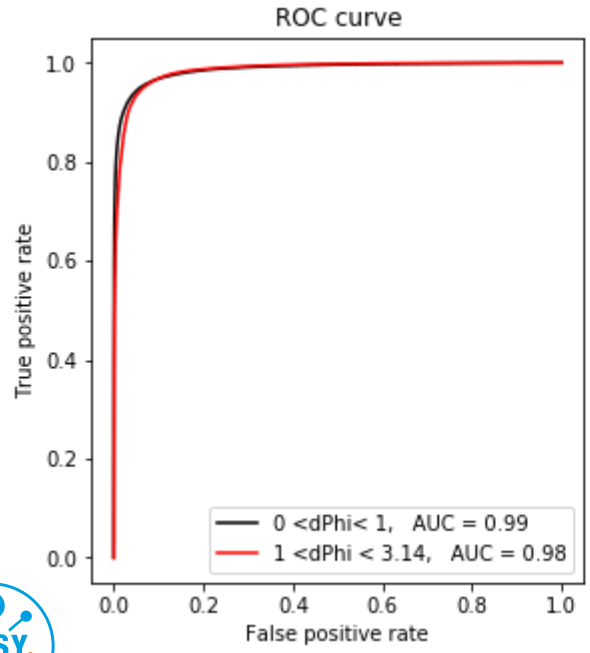
Thank you
Danke

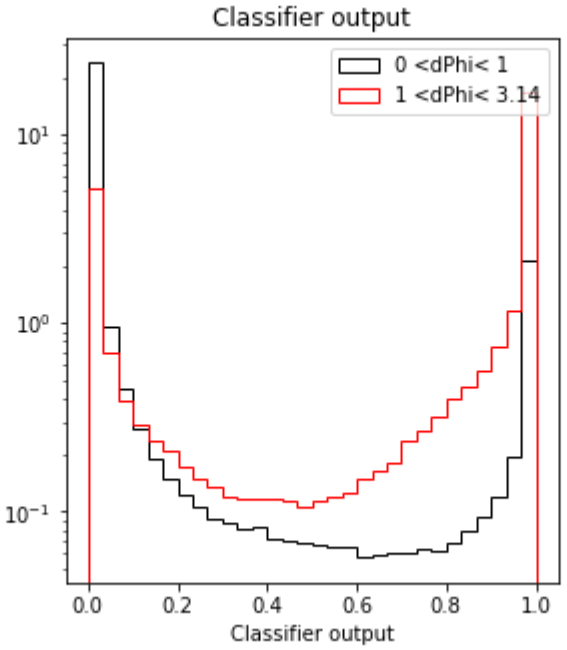
Backup

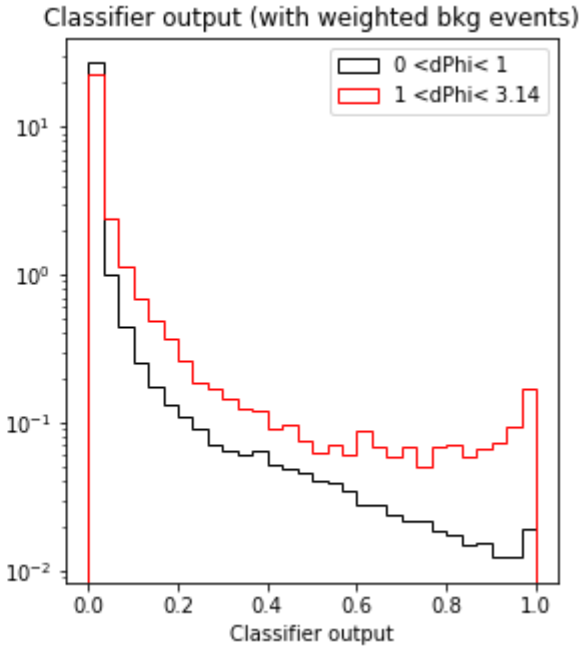
λ=0.5

- ✔ Large area under the ROC curve
- ✔ Small Δφ events are classified more as bkg
- ✔ Peaks near 0 & 1 for clf output
- • Clf output is getting **more independent** of Δφ





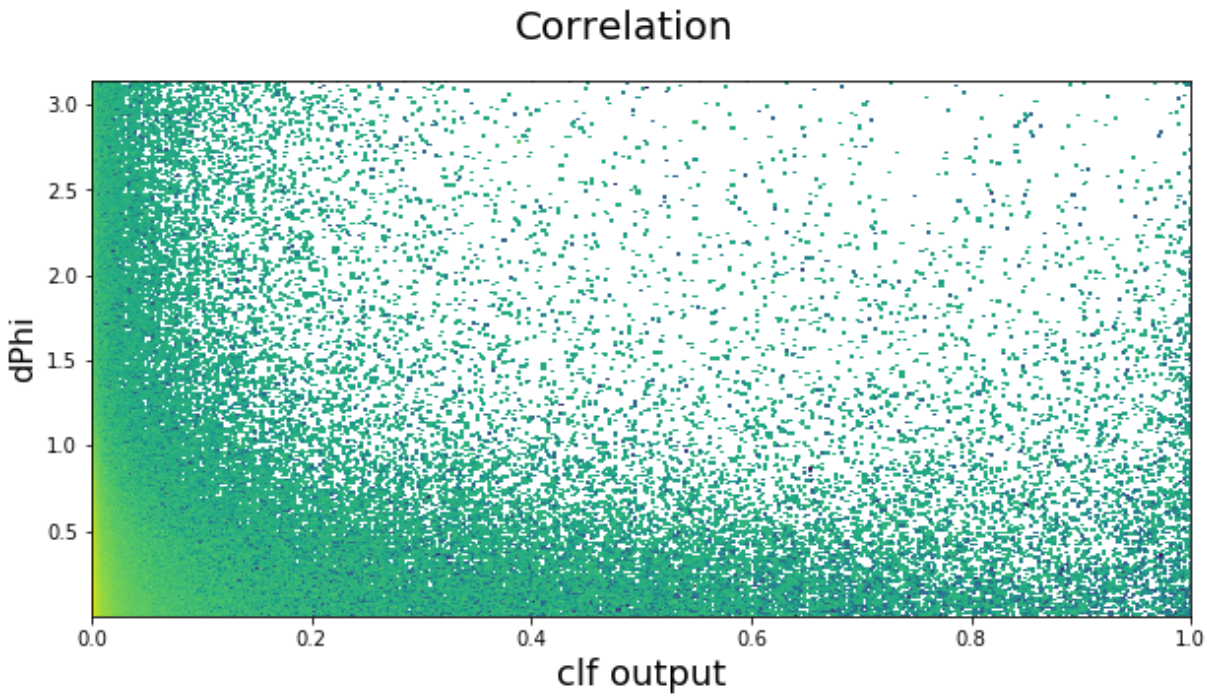




λ=0.5

```
Classifier Output:
A = [4.89377176]
D = [0.13021808]
B = [0.06083906]
C = [0.00422393]
Ratio = [2.60919688]
```

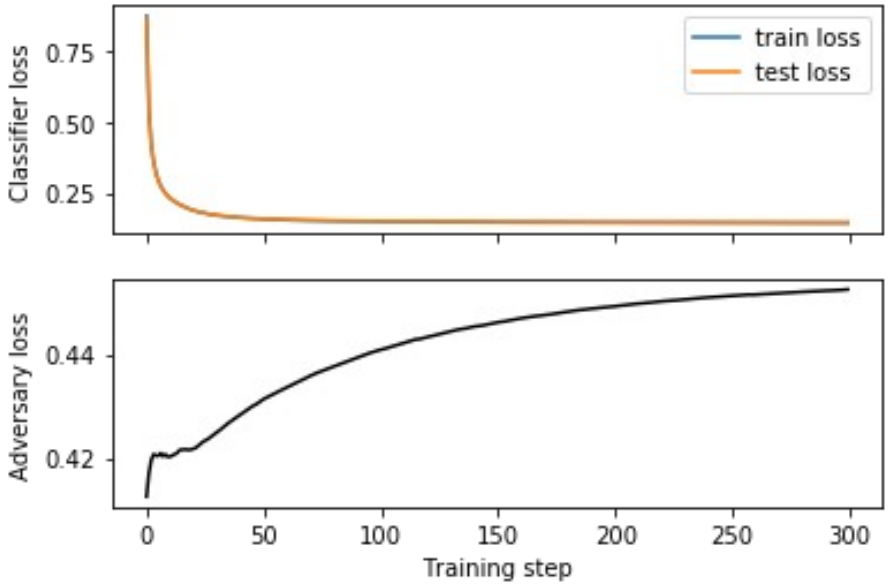
Ratio ≠ 1

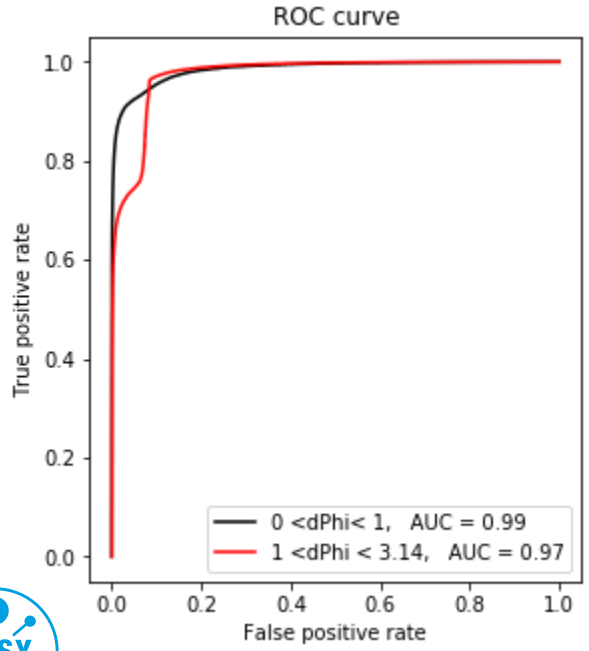


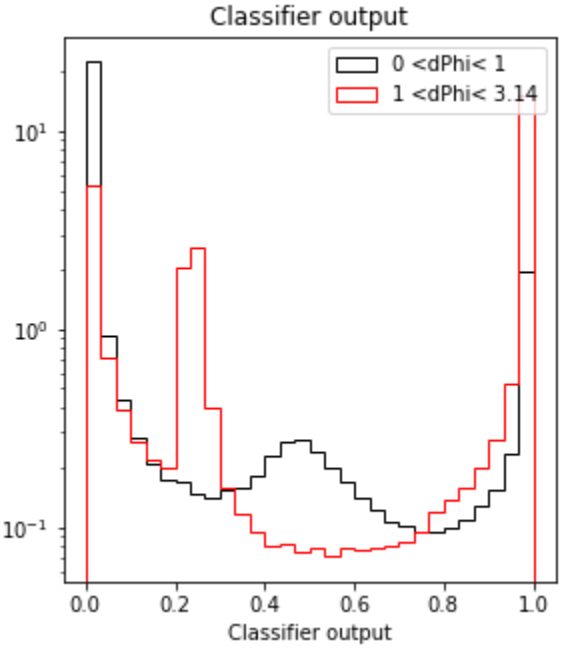
➔ **Try a different value of λ**

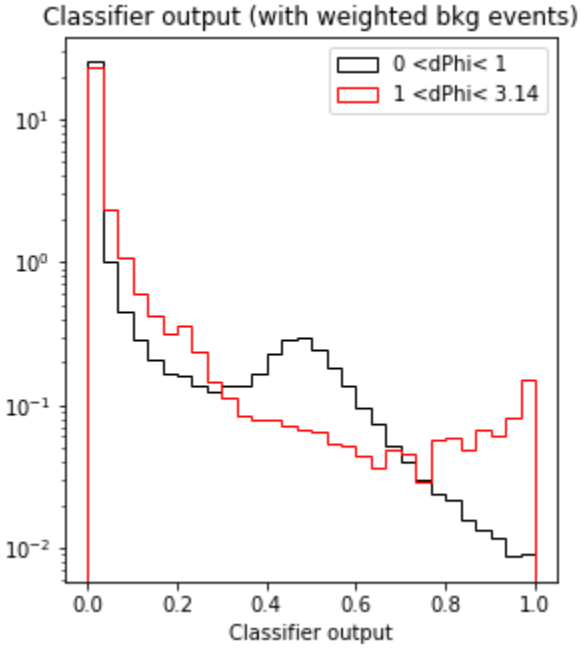
λ=0.75

- ✔ Large area under the ROC curve
- ✔ Small Δφ events are classified more as bkg
- ✗ No longer ideal classification
- • Clf output is getting **more independent** of Δφ





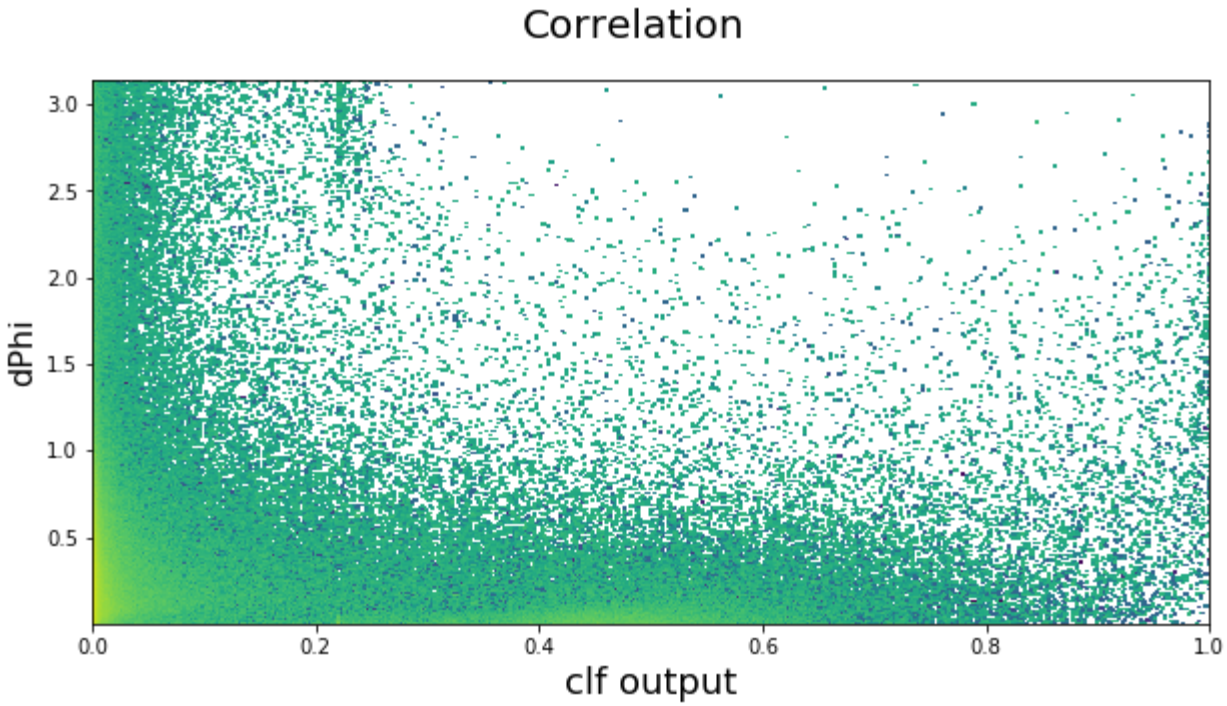




λ=0.75

```
Classifier Output:
A = [0.13147355]
B = [4.79541105]
C = [0.15919977]
D = [0.00296846]
Ratio = [1.4703602]
```

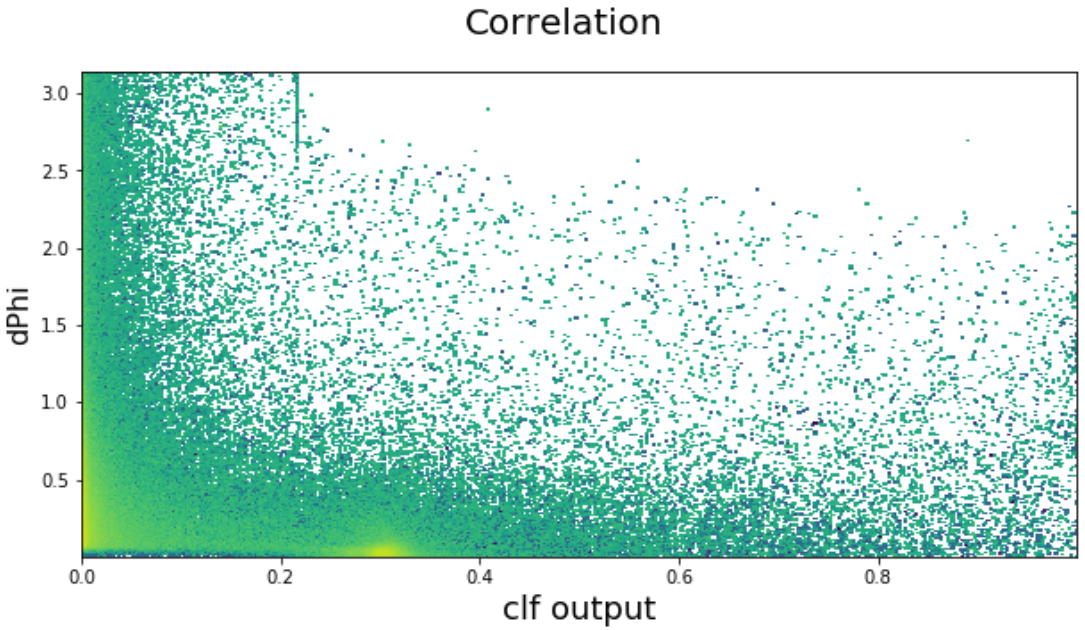
Ratio → 1

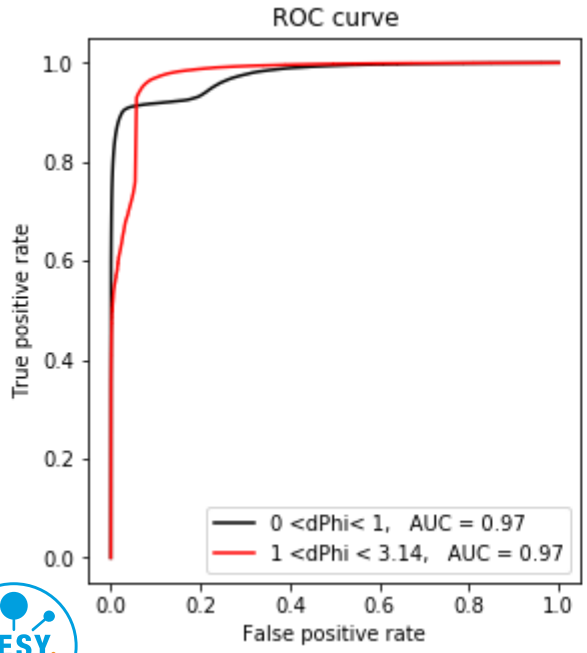


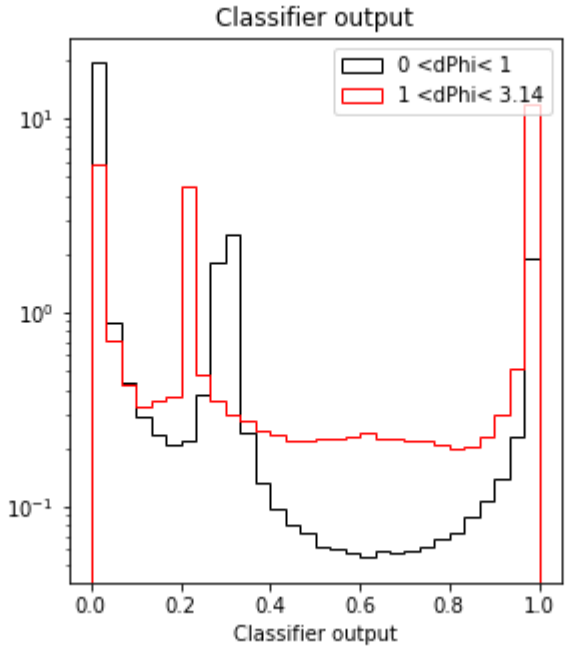
- ✔ **Ratio is getting closer to 1**
- ➔ **Try a different value of λ**

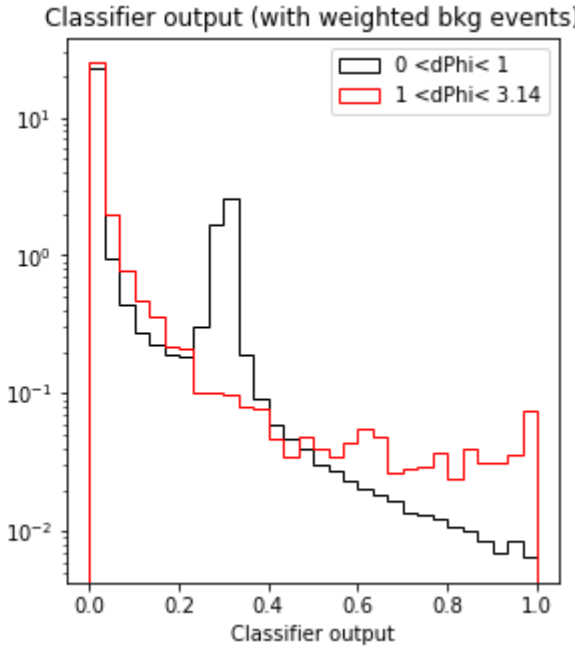
λ=1

```
Classifier Output:
A = [0.13270666]
B = [4.91468552]
C = [0.0399253]
D = [0.00173535]
Ratio = [0.62123818]
```



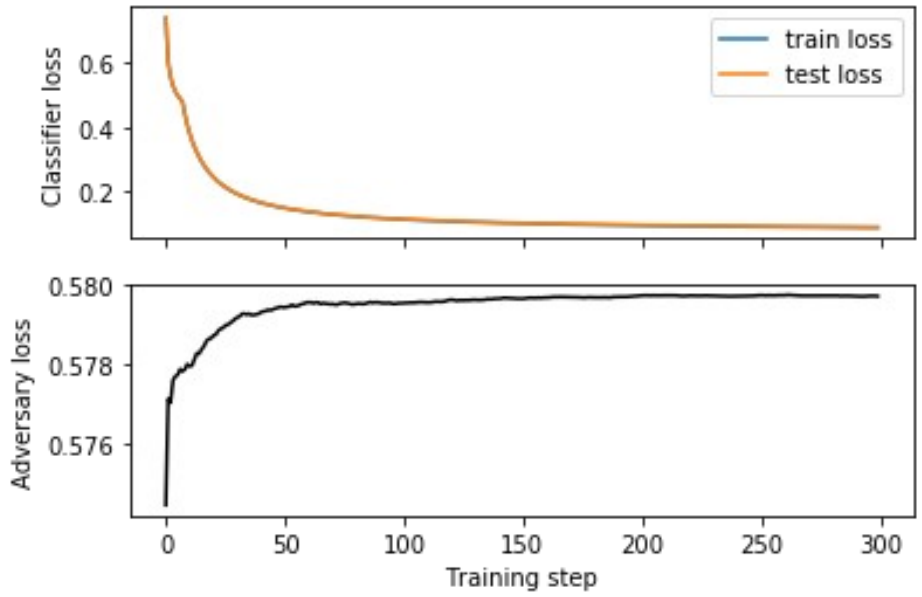


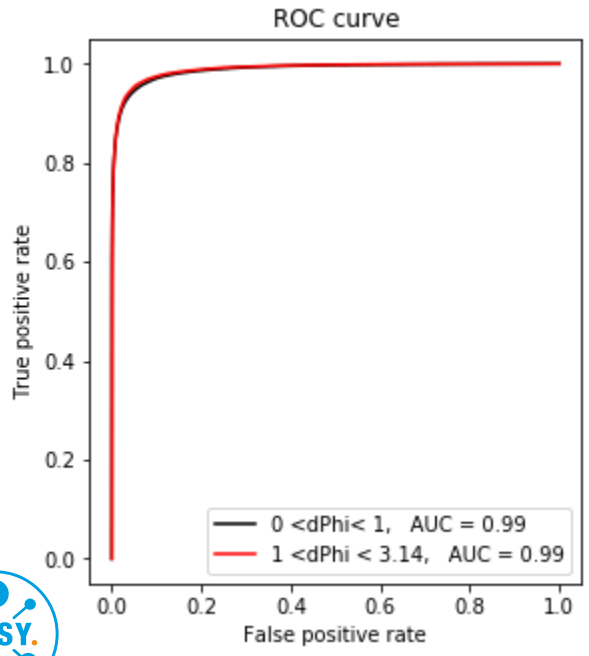


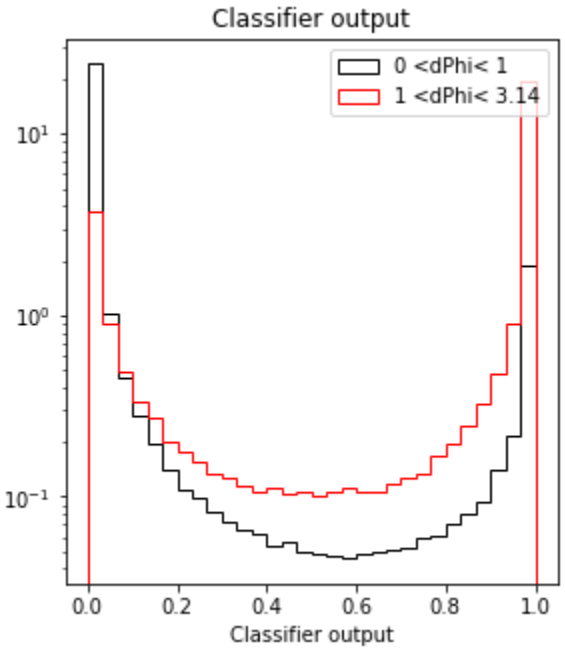


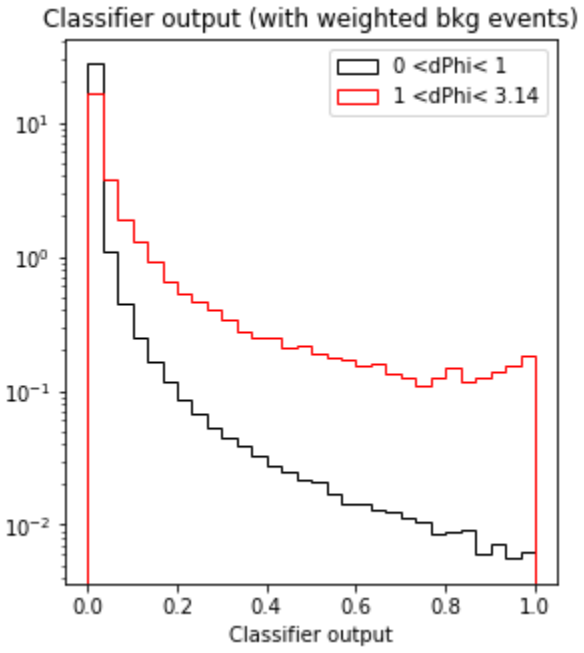
λ=10

- ✔ Large area under the ROC curve
- ✔ Small Δφ events are classified more as bkg
- ✔ Peaks near 0 & 1 for clf output
- ✗ Clf output is dependent on Δφ
- ➔ **Just like the Normal Classifier result**





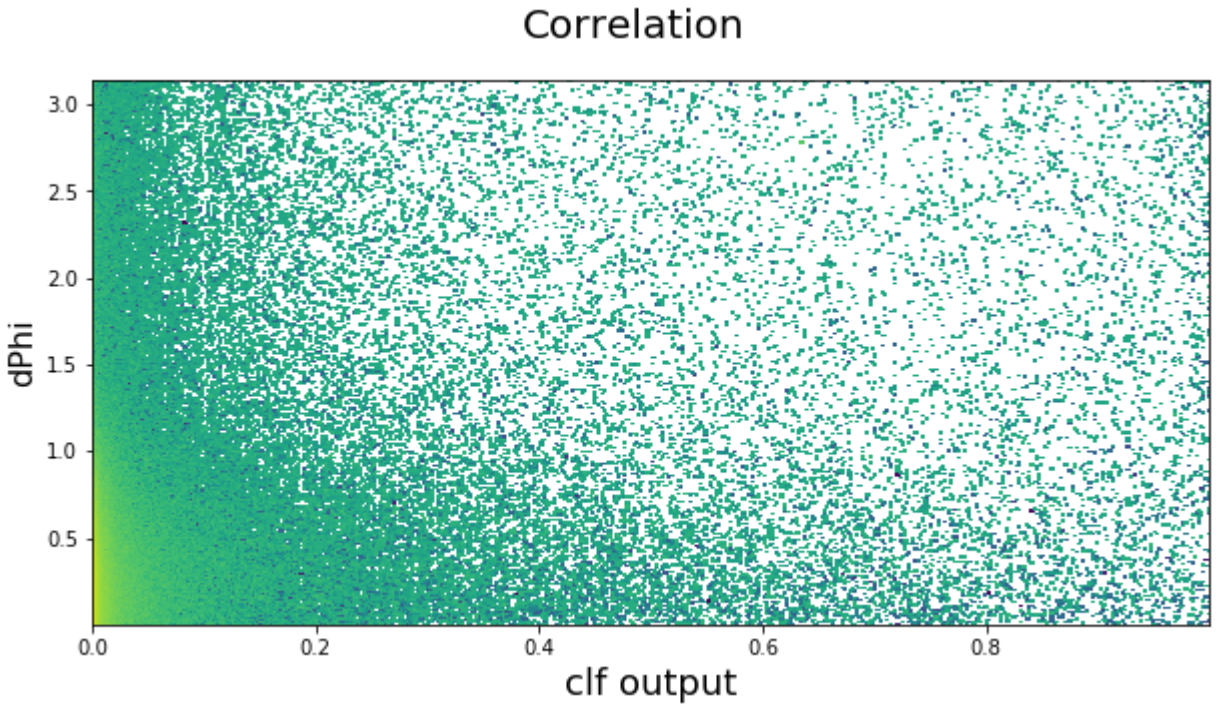




λ=10

```
Classifier Output:
A = [0.12198268]
B = [4.92207056]
C = [0.03254026]
D = [0.01245933]
Ratio = [0.06472567]
```

Ratio ≠ 1



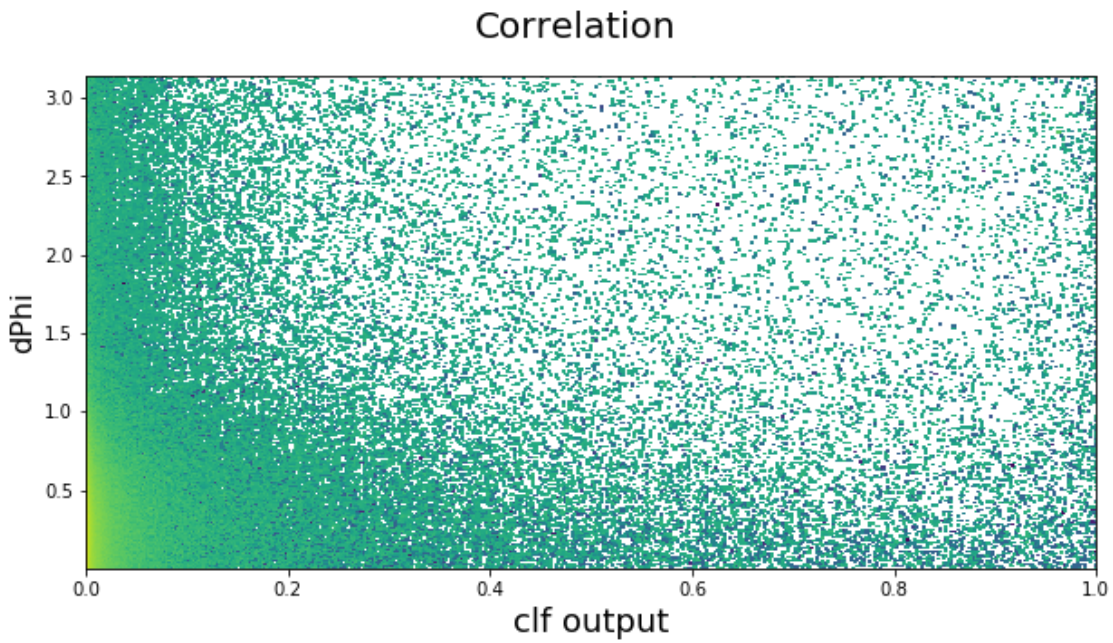
✗ **Ratio is further from 1**

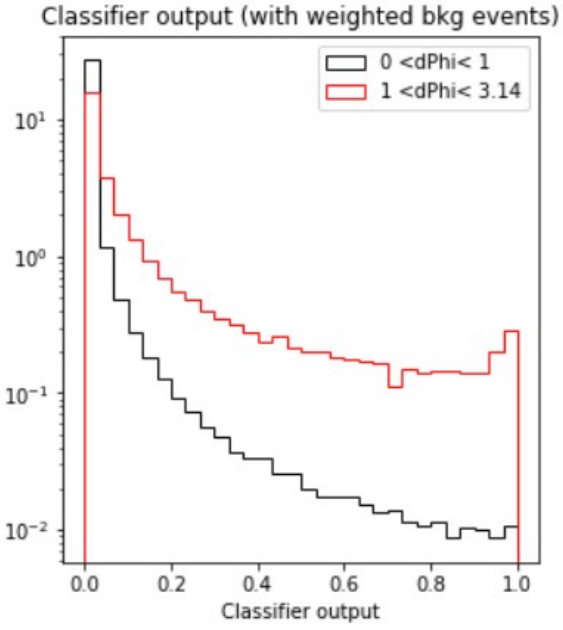
➔ **The adversarial result is just the same as the normal classifier**

Back to Normal Classifier

```
Classifier Output:
A = [0.12016173]
B = [4.91590657]
C = [0.03870425]
D = [0.01428028]
Ratio = [0.06624977]
```

Ratio ≠ 1





➔ *clf* output is dependent on Δφ

➔ **Try Adversarial Training for different values of λ**