Note

Feng-Yang Hsieh

1 SPANet2

Version 2 of SPANet. call it SPANet2.

1.1 Defining event topology

Defining the event topology in .ymal file. The structure of the .yaml file follows this format:

```
INPUTS:
    SEQUENTIAL:
```

1.2 Creating training dataset

.hdf5

1.3 Training options

1.4 Training

Training:

```
python -m spanet.train -of <OPTIONS_FILE> --log_dir <LOG_DIR>  --name <NAME>
```

<OPTIONS_FILE>: JSON file with options. <LOG_DIR>: output directory. <NAME>: subdirectory name

Evaluation:

```
python -m spanet.test <log_directory> -tf <TEST_FILE>
```

<log_directory>: directory containing the checkpoint and options file. <TEST_FILE> will replace the test file in the option file.

Prediction:

```
python predict.py <log_directory> <output name> -tf <TEST_FILE> --gpu ???
```

2 Test SPANet2

2.1 SM SPANet

Generate the correct format $\kappa_\lambda = 1$ training data for SPANet2 training.

- Training sample:
  - Total sample size: 76,131
  - 1h sample size: 14,527
  - 2h sample size: 60,122
  - 5% used on validation
- Testing sample:
  - Total sample size: 8,460
  - 1h sample size: 1,577
  - 2h sample size: 6,744

The training results are presented in Table 1.

Table 1: SPA-NET2 training results on the SM di-Higgs samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.280	0.907	0.907
$= 5$	0.287	0.806	0.847
≥ 6	0.229	0.679	0.753
Total	0.797	0.805	0.841

2.2 $\kappa 5$ SPANet

Generate the $\kappa_\lambda = 5$ training data for SPANet2 training.

- Training sample:
  - Total sample size: 78,388

- 1h sample size: 16,013
- 2h sample size: 59,180
- 5% used on validation

- Testing sample:
  - Total sample size: 8,710
  - 1h sample size: 1,846
  - 2h sample size: 6,486

The training results are presented in Table 2.

Table 2: SPA-NET2 training results on the di-Higgs $\kappa_\lambda = 5$ samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.315	0.689	0.689
$= 5$	0.255	0.617	0.639
≥ 6	0.174	0.499	0.544
Total	0.745	0.620	0.638

2.3 Resonant SPANet

Generate the correct format resonant training data for SPANet2 training.

- Training sample:
  - Total sample size: 51,145
  - 1h sample size: 9,320
  - 2h sample size: 40,991
  - 5% used on validation
- Testing sample:
  - Total sample size: 5,683
  - 1h sample size: 1,011
  - 2h sample size: 4,582

The training results are presented in Table 3.

Table 3: SPA-NET2 training results on the resonant di-Higgs samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.316	0.930	0.930
$= 5$	0.282	0.808	0.839
≥ 6	0.208	0.660	0.727
Total	0.806	0.818	0.846

2.4 Mixing κ_λ SPANet

Generate the correct format mixing κ_λ training data for SPANet2 training.

- Training sample:
  - Total sample size: 51,145
  - 1h sample size: 9,320
  - 2h sample size: 40,991
  - 5% used on validation
- Testing sample:
  - Total sample size: 5,683
  - 1h sample size: 1,011
  - 2h sample size: 4,582

The training results are presented in Table 4.

Table 4: SPA-NET2 training results on the resonant di-Higgs samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.316	0.930	0.930
$= 5$	0.282	0.808	0.839
≥ 6	0.208	0.660	0.727
Total	0.806	0.818	0.846

2.5 Summary

In most cases, the performance of SPANet2 is worse than the old one. Some default options are different between the two versions. But even if the options are identical, the training results cannot be better.

The training results of old and new versions of SPANet have been summarized in Table 5.

Table 5: SPA-NET2 training results on the resonant di-Higgs samples.

	Event efficiency	
	SPANet	SPANet2
SM	0.868	0.805
kappa 5	0.725	0.620
Resonant	0.903	0.818
Mixing κ_λ	0.833	0.830

3 Combine jet assignment and event classification

This section trains the SPANet2 on the jet assignment and event classification task at the same time. This is the new feature of SPANet2.

3.1 $\kappa_\lambda = 5$ sample

For the jet assignment part, use the same sample as in Sec. 2.2.

- Training sample:
  - Total sample size: 168,125
  - Signal sample size: 78,388
  - Background sample size: 89,737
  - 5% used on validation
- Testing sample:
  - Total sample size: 18,681
  - Signal sample size: 8,710

- Background sample size: 9,971

The training results are presented in Table 6.

Table 6: SPA-NET2 training results on the $\kappa_\lambda = 5$ samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.316	0.930	0.930
$= 5$	0.282	0.808	0.839
≥ 6	0.208	0.660	0.727
Total	0.806	0.818	0.846

3.2 Mixing κ_λ

3.2.1 Training samples

For signal, set $\kappa_\lambda = [-5, -3, -1, 1, 2, 3, 5, 7, 9, 12]$ and generate 9,000 samples on each κ_λ point for training. The training samples are required to pass the "Four tag cut", i.e., there are at least four b-tagged jets with $p_{\text{T}} > 40$ GeV and $|\eta| < 2.5$.

Note that the κ_λ value is an input feature. For the background sample, the input κ_λ value is randomly chosen from the above values.

For the jet assignment part,

- Training sample:
  - Total sample size: 90,000
  - 1h sample size: 18,020
  - 2h sample size: 69,267
  - 5% used on validation
- Testing sample:
  - Total sample size: 9,000
  - 1h sample size: 1,802
  - 2h sample size: 6,899

For event classification,

- Training sample:

- Total sample size: 179,737
- Signal sample size: 90,000
- Background sample size: 89,737
- 5% used on validation

- Testing sample:
  - Total sample size: 18,971
  - Signal sample size: 9,000
  - Background sample size: 9,971

3.2.2 Hyperparameters setting

Some options are different between SPANet and SPANet2. List the different options below

- `hidden_dim`: 128 $\to$ 64
- `learning_rate`: 0.0007 $\to$ 0.0015
- `num_attention_heads`: 8 $\to$ 4

The total loss function consists of assignment loss and classification loss. The same weights are assigned to these losses.

- `assignment_loss_scale`: 1.0
- `classification_loss_scale`: 1.0

3.2.3 Training results

Table 7 presents the jet assignment training results.

Table 7: SPANet2 training results on the mixing κ_λ samples.

N_{Jet}	Event Fraction	Event Efficiency	Higgs Efficiency
$= 4$	0.139	0.866	0.866
$= 5$	0.130	0.806	0.834
≥ 6	0.095	0.704	0.766
Total	0.364	0.802	0.829

Table 8 presents the classification training results.

Table 8: The SPANet2 classification training results with mixing κ_λ sample.

Training sample	ACC	AUC
Mixing κ_λ	0.828	0.911

4 κ_λ constraints on SPANet2

4.1 SPANet2 classification

Use the SPANet2 to do the signal background classification task.

When an event is put in SPANet2, SPANet2 will return a signal score p_{signal} which represents the confidence that this event is a signal event. The requirement of $p_{\text{signal}} > p_{\text{th}}$ is imposed for event selection, where $p_{\text{th}} = 0.90$. Where we do not choose the value that can maximize the $S/\sqrt{B}$. Because the value is too close to 1, very few events can pass this selection. Thus we can not do further analysis.

Set the κ_λ limits by the profile likelihood method and CLs method. Table 9 results from κ_λ constraints.

Table 9: The κ_λ constraints of SPANet2.

	Expected Constraint			
	Profile likelihood		CLs	
Selection method	Lower	Upper	Lower	Upper
SPANet2	-3.48	9.18	-3.41	9.09

5 Comparision with previous results

This section summary the results among the "min-ΔR DNN", "κ5 SPANet DNN", "mixing κ SPANet2".

5.1 Pairing performance

Figure 1 shows the pairing efficiency of different methods. Where the mixing κ SPANet2 has the best performance.

5.2 Classification performance

Table 8 presents the classification training results.

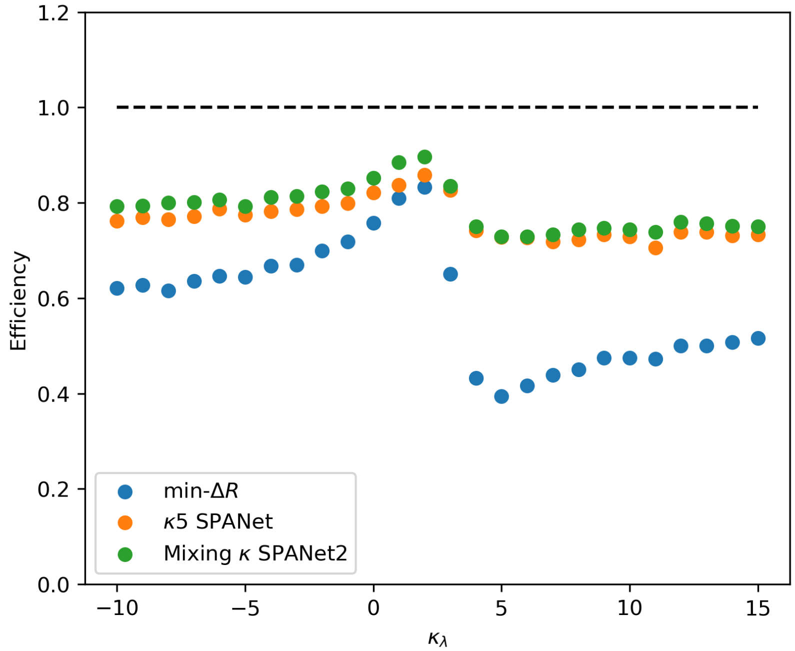


Figure 1: The pairing performance for different κ_λ samples.

Table 10: The classification performance of different selection methods.

Selection method	ACC	AUC
min-ΔR DNN	0.783	0.864
κ5 SPANet DNN	0.792	0.875
mixing κ SPANet2	0.828	0.911

5.3 κ_λ constraints

Table 11 is the κ_λ constraints of the different selection methods.

Table 11: The κ_λ constraints of different selection methods.

		Expected Constraints			
		Profile likelihood		CLs	
Pairing method	Selection method	Lower	Upper	Lower	Upper
min-ΔR	DNN	-3.81	11.16	-3.73	11.15
κ5 SPANet	DNN	-4.08	11.65	-4.02	11.68
Mixing κ SPANet2	SPANet2	-3.48	9.18	-3.41	9.09

5.4 Mass distribution plot

Figure 2 and 3 show the Higgs mass distribution for signal and background events with different pairing methods. The selection does not apply. The mass planes for the signal process all look similar for all pairing methods. For background, the results of min-ΔR are very different from others.

Figure 4 are the m_{HH} distributions after the selection.

6 SPANet2 pairing + DNN selection

This section uses the mixing κ SPANet2 for jet pairing and generates the samples for DNN training. The training samples consisted of different κ_λ value events.

6.1 Training samples

Set $\kappa_\lambda = [-5, -3, -1, 1, 2, 3, 5, 7, 9, 11]$, for each κ_λ point generate samples. For signal, the different κ_λ samples are mixed. For each type, the same number of samples is used. For background, the κ_λ values are randomly chosen from the above list. Training sample sizes are shown in Table 12.

6.2 Training results

The DNN training results are summarized in Table 13.

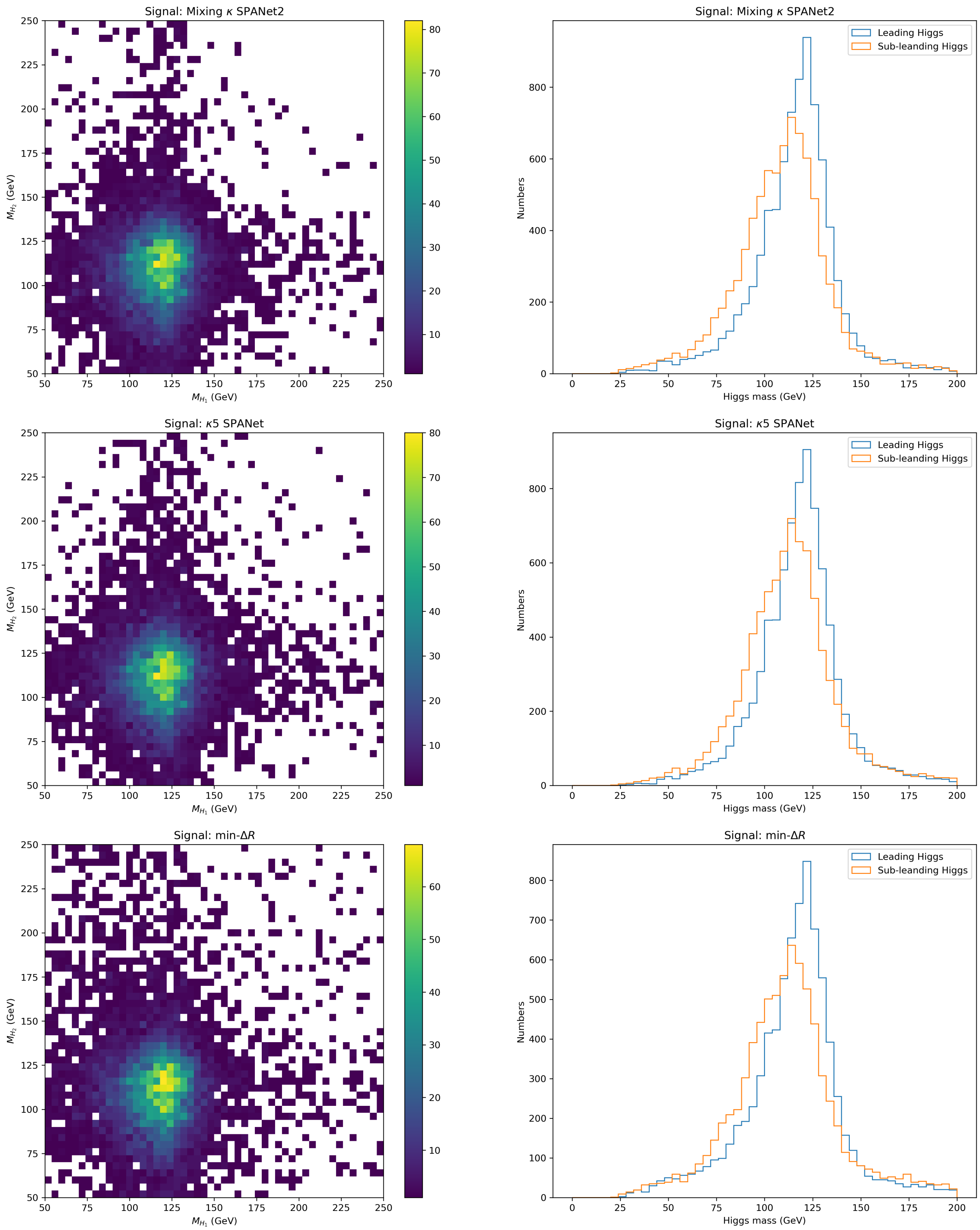


Figure 2: The mass plane and distribution of Higgs candidate for signal events with different pairing methods. The top one is mixing κ SPANet2 pairing, the middle one is $\kappa 5$ SPANet pairing, bottom one is min-ΔR pairing.

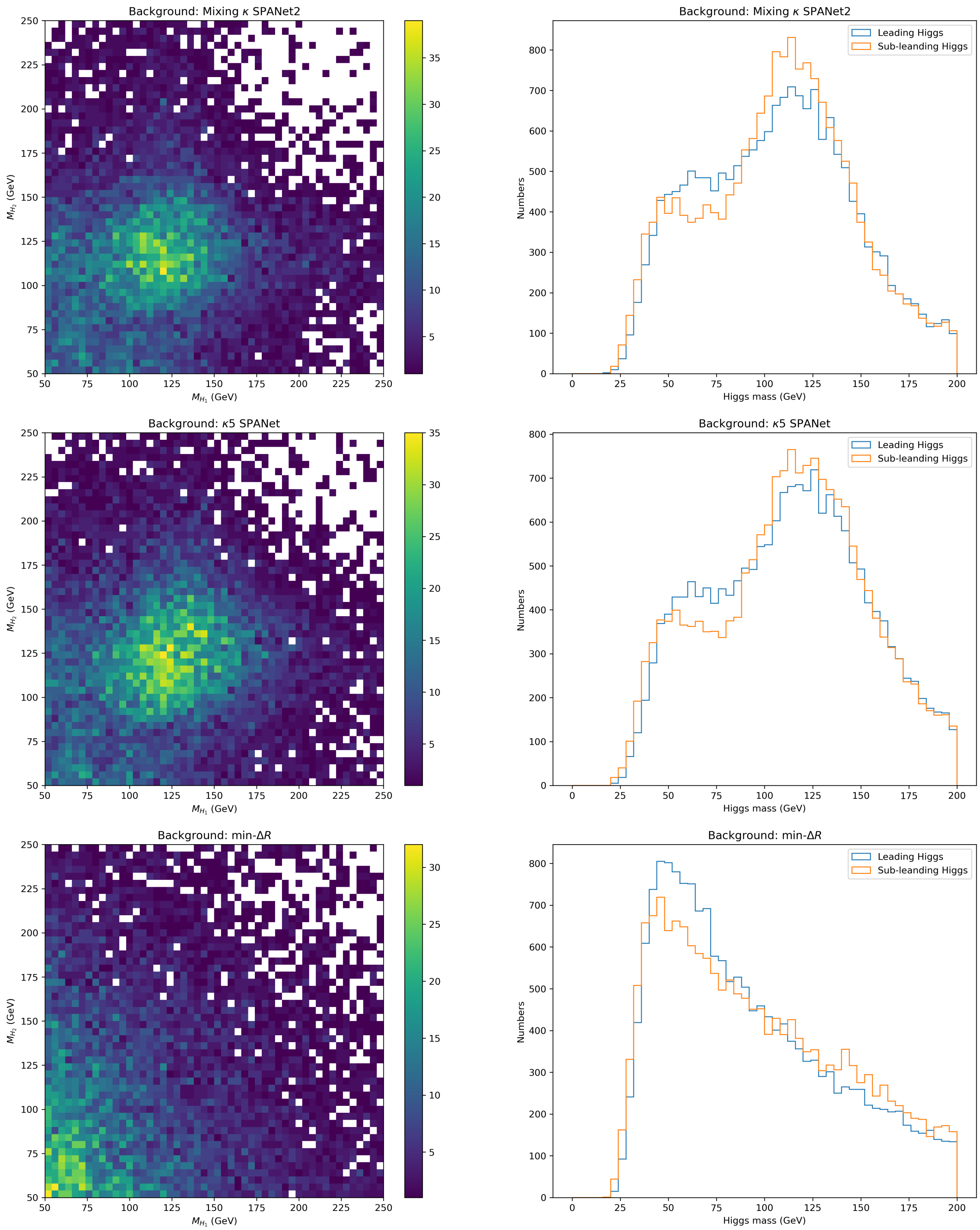
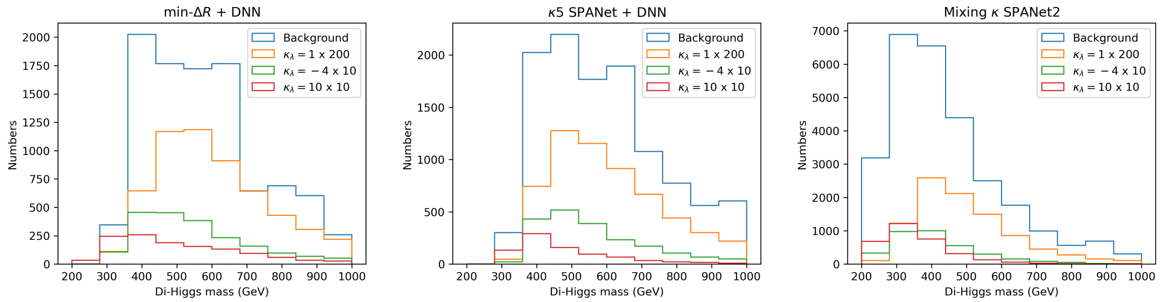


Figure 3: The mass plane and distribution of Higgs candidate for background events with different pairing methods. The top one is mixing κ SPANet2 pairing, the middle one is κ5 SPANet pairing, bottom one is min-ΔR pairing.



(a) min-ΔR DNN (b) κ5 SPANet DNN (c) Mixing κ SPANet2

Figure 4: The m_{HH} distributions after selection. The DNNs are trained with different pairing method samples.

Table 12: The sample size for signal and background, which are the training and testing sample sizes.

Signal	Background
80k + 8k	80k + 8k

Table 13: The DNN training results with different pairing methods. The training samples contain different κ_λ samples. The average and standard deviation of 10 training is presented.

Pairing method	ACC	AUC
min-ΔR	0.800 ± 0.010	0.882 ± 0.010
κ5 SPA-NET	0.794 ± 0.004	0.876 ± 0.004
mixing κ SPANet2	0.800 ± 0.003	0.882 ± 0.004

6.3 κ_λ limits

Set the κ_λ limits by the profile likelihood method and CLs method. The model with the best ACC is used. The p_{th} is chosen such that maximize $S/\sqrt{B}$.

The results of κ_λ constraints are summarized in Table 14.

Table 14: The κ_λ constraints with DNN selection samples.

	Expected Constraint			
	Profile likelihood		CLs	
Pairing method	Lower	Upper	Lower	Upper
min-ΔR	-3.20	10.87	-3.16	10.78
κ5 SPA-NET	-4.44	10.86	-4.35	10.78
mixing κ SPANet2	-3.42	10.85	-3.40	10.77

Here, the bug of the previous testing sample is fixed. In the previous background testing sample, all κ_λ are set to 1 not randomly chosen from the κ_λ value list.

7 Summary

7.1 Classification performance

Table 15 presents the classification training results.

Table 15: The classification performance of different selection methods.

Selection method	ACC	AUC
min-ΔR DNN	0.800 ± 0.010	0.882 ± 0.010
κ5 SPANet DNN	0.794 ± 0.004	0.876 ± 0.004
mixing κ SPANet2 DNN	0.800 ± 0.003	0.882 ± 0.004
mixing κ SPANet2	0.822 ± 0.007	0.906 ± 0.006

7.2 κ_λ constraints

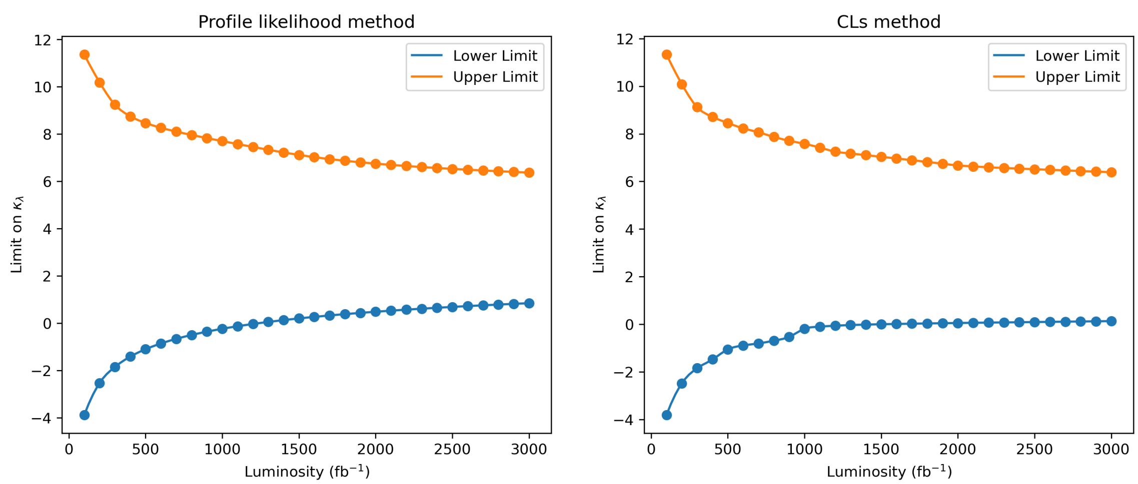
Table 16 is the κ_λ constraints of the different selection methods.

Table 16: The κ_λ constraints of different selection methods.

		Expected Constraints			
		Profile likelihood		CLs	
Pairing method	Selection method	Lower	Upper	Lower	Upper
min-ΔR	DNN	-3.20	10.87	-3.16	10.78
κ5 SPANet	DNN	-4.44	10.86	-4.35	10.78
Mixing κ SPANet2	DNN	-3.42	10.85	-3.40	10.77
Mixing κ SPANet2	SPANet2	-3.18	8.79	-3.13	8.77

8 κ_λ constraints with different luminosities

Use the min-ΔR method to set constraints with different luminosity. The results are shown in Figure 5.



(a) (b)

Figure 5: The κ_λ constraints with different luminosities. Use min-ΔR method for pairing and DNN for selection.

Use mixing κ SPANet2 for constraints setting with different luminosity. The results are presented in Table 17.

9 SPANet classifier

Figure 6 is the model structure of SPANet2. The classifier part takes the outputs of the transformer encoder. The architecture of the classifier part is just the feed-forward structure

Table 17: The κ_λ constraints with different luminosities. Use SPANet2 for event selection.

	Expected Constraints				Equivlent luminosity for min-ΔR			
	Profile likelihood		CLs		Profile likelihood		CLs	
$\mathcal{L}$ (fb^{-1})	Lower	Upper	Lower	Upper	Lower	Upper	Lower	Upper
139	-3.18	8.79	-3.13	8.77	145	388	144	381
300	-1.96	7.96	-1.96	7.89	280	810	275	796

networks.

The SPANet classifier does not take the results from the jet assignment part, because it is worse than if we just take the transformer outputs. The reason is that it can lead to worse performance due to errors in that part.

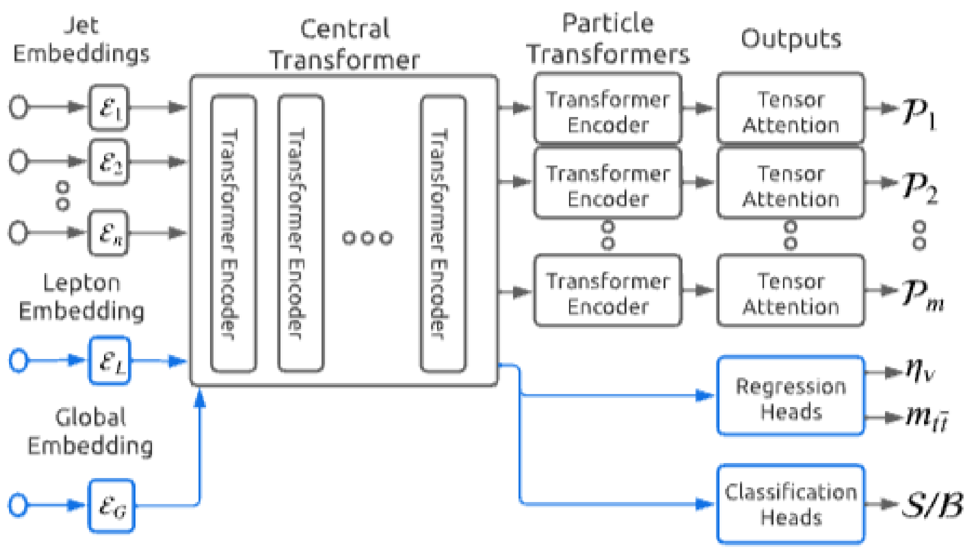


Figure 6: The model structure of SPANet2.

10 SPANet2 classification

This section turns off the jet assignment part in SPANet2 by setting the assignment loss weight to zero.

- `assignment_loss_scale`: 0.0
- `classification_loss_scale`: 1.0

Table 18: The SPANet2 classification training results with mixing κ_λ sample. The jet assignment part is turned off. The average and standard deviation of 10 training are presented.

Training sample	ACC	AUC
Mixing κ_λ	0.809 ± 0.013	0.890 ± 0.014

Using the mixing κ samples for training. The samples are the same as Sec. 3.2.1. Table 18 presents the classification training results.

Set $p_{\text{th}} = 0.93$ and use the profile likelihood method and CLs method for the κ_λ setting. Table 19 is the results of κ_λ constraints. These results are worse than simultaneously training on jet assignment and classification tasks.

Table 19: The κ_λ constraints of SPANet2.

	Expected Constraint			
	Profile likelihood		CLs	
Selection method	Lower	Upper	Lower	Upper
SPANet2	-5.01	10.97	-4.92	10.89

11 SPANet embedding vectors

The SPANet embedding vectors can be saved in `.hdf5` file by this command

```
python -m spanet.predict <log_dir> <output name> -tf <TEST_FILE> \
--gpu --output_vectors
```

<log_dir>: directory containing the checkpoint and options file. <TEST_FILE>: the test file path.

11.1 Principal component analysis

Use the PCA class implemented in scikit-learn to do the principal component analysis (PCA) on the SPANet embedding vectors. The variance ratio of the first ten components is shown in Figure 7.

Calculate the correlation coefficients with principal components and the high-level observables. The high-level observables are the DNN input features that are constructed by the SPANet2 pairing.

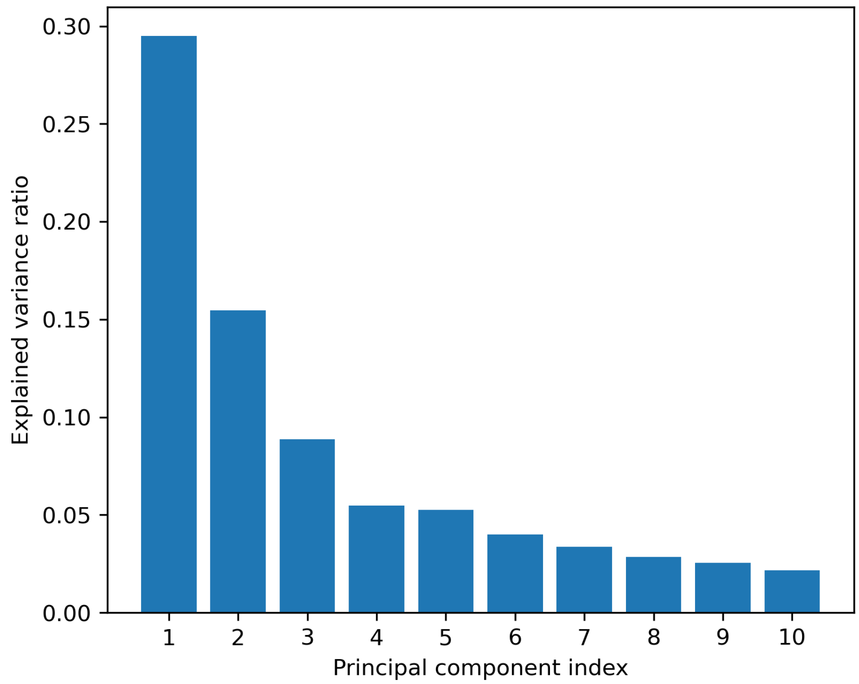


Figure 7: The variance ratio of the first ten principal components.

The results are presented in Figure 8. In Figure 9, the correlation coefficients of signal and background events are calculated separately. The level of correlation of most variables is very low in the background case compared to the signal one's.

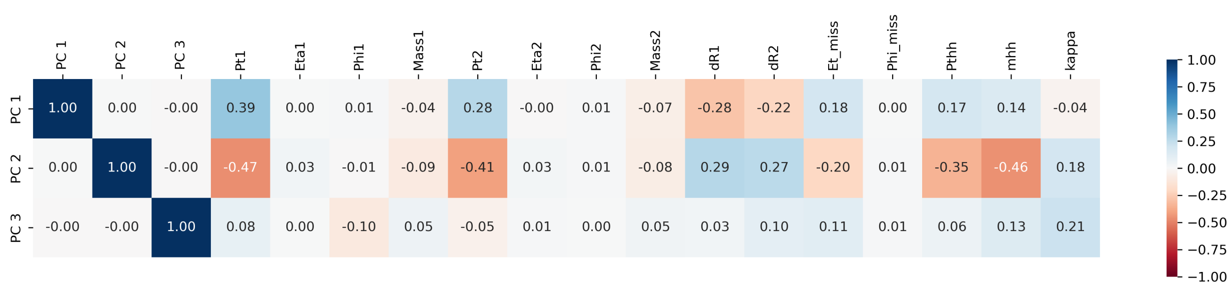


Figure 8: The correlation coefficients of the first three principal components and high-level observables.

12 Hyperparameter tuning

This section uses Optuna to do the hyperparameter optimization.

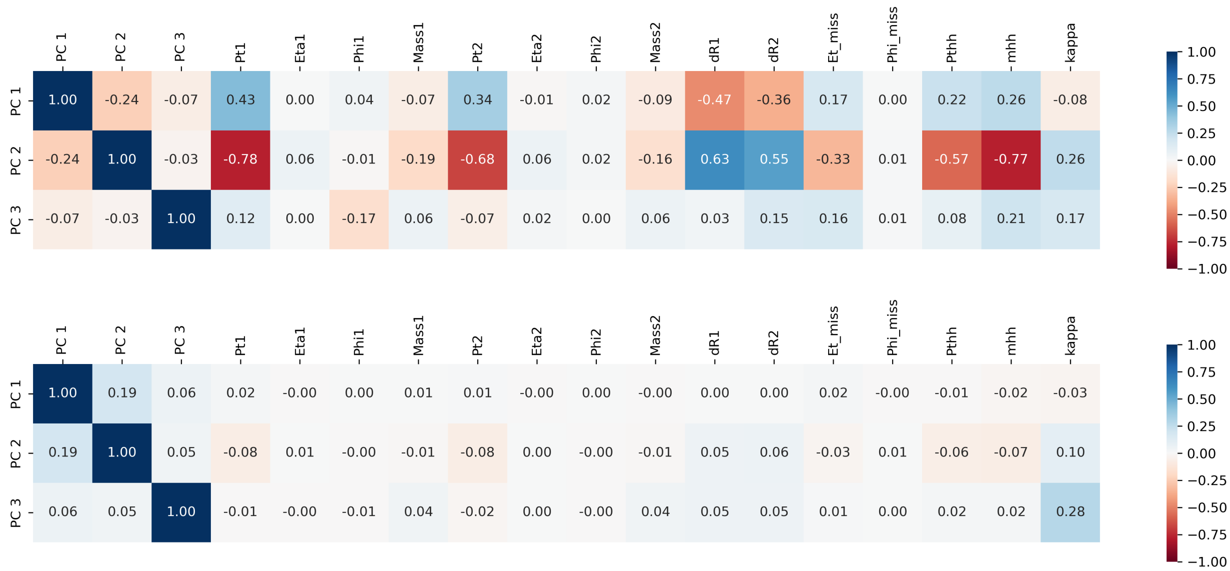


Figure 9: The correlation coefficients of the first three principal components and high-level observables. Where the signal and background samples are calculated separately.

12.1 Optimization range

The optimization range of different hyperparameters is listed in the below

- `learning_rate`: $[10^{-5}, 10^{-2}]$
- `dropout`: $[0, 0.5]$
- `gradient_clip`: $[0, 0.5]$
- `l2_penalty`: $[0, 0.0005]$
- `hidden_dim`: $[16, 32, 64, 128, 256]$
- `num_encoder_layers`: $[2, 8]$
- `num_branch_encoder_layers`: $[2, 8]$
- `num_classification_layers`: $[1, 5]$

Each parameter set was trained for 10 epochs. Test 100 trials.

12.2 Hyperparameter optimization results

The hyperparameter optimization results are listed below

- `learning_rate`: 0.00659
- `dropout`: 0.0059
- `gradient_clip`: 0.425
- `l2_penalty`: 0.000374
- `hidden_dim`: 32
- `num_encoder_layers`: 8
- `num_branch_encoder_layers`: 2
- `num_classification_layers`: 1

Use this parameter set for full training. The samples are the same as Sec. 3.2.1. Table 32 presents the classification training results. This result is a little better than Table 15.

Table 20: The SPANet2 classification training results with mixing κ_λ sample. Use the Optuna hyperparameter optimization results. The average and standard deviation of 10 training are presented.

	ACC	AUC
Best HP	0.828 ± 0.002	0.911 ± 0.001

Set $p_{\text{th}} = 0.95$ and use the profile likelihood method and CLs method for the κ_λ setting. Table 21 is the results of κ_λ constraints. These results are similar to the previous one (Table 16).

Table 21: The κ_λ constraints of SPANet2.

	Expected Constraint			
	Profile likelihood		CLs	
Selection method	Lower	Upper	Lower	Upper
SPANet2	-3.07	8.80	-3.04	8.76

12.3 DNN hyperparameter optimization

The optimization range of different hyperparameters is listed in the below

- `learning_rate`: $[10^{-5}, 10^{-1}]$

- `hidden_dim`: $[16, 32, 64, 128, 256]$
- `n_layers`: $[1, 5]$

Test 100 trials.

12.4 DNN hyperparameter optimization results

For min-ΔR, the hyperparameters optimization results are

- `learning_rate`: 0.00495
- `hidden_dim`: 256
- `n_layers`: 3

For mixing κ SPANet2, the results are

- `learning_rate`: 0.000948
- `hidden_dim`: 256
- `n_layers`: 2

Use these parameter sets for training. The samples are the same as Sec. 6.1. Table 22 presents the DNN classification training results. The results are similar to Table 15.

Table 22: The DNN classification training results with mixing κ_λ sample. Use the Optuna hyperparameter optimization results. The average and standard deviation of 10 training are presented.

	ACC	AUC
min-ΔR	0.799 ± 0.011	0.881 ± 0.012
mixing κ SPANet2	0.803 ± 0.004	0.884 ± 0.004

Set $p_{\text{th}} = 0.95$ and use the profile likelihood method and CLs method for the κ_λ setting. Table 23 is the results of κ_λ constraints. These results are similar to the previous one (Table 16).

Table 23: The κ_λ constraints of DNN with best hyperparameters.

	Expected Constraint			
	Profile likelihood		CLs	
Selection method	Lower	Upper	Lower	Upper
min-ΔR DNN	-3.20	10.31	-3.16	10.19
mixing κ SPANet2 DNN	-3.29	11.14	-3.14	11.04

13 Summary

13.1 Pairing performance

Figure 10 shows the pairing efficiency of different methods. Table 24 is the pairing efficiency of some κ_λ points. Where the mixing κ SPANet2 has the best performance.

Table 24: The classification performance of different selection methods.

	κ_λ		
Pairing method	-5	1	5
min-ΔR	0.644	0.809	0.395
mixing κ SPANet2	0.793	0.885	0.729

13.2 Classification performance

Table 25 presents the classification training results. Where the hyperparameter optimization is finished.

Table 25: The classification performance of different selection methods.

Selection method	ACC	AUC
min-ΔR DNN	0.799 ± 0.011	0.881 ± 0.012
mixing κ SPANet2 DNN	0.803 ± 0.004	0.884 ± 0.004
mixing κ SPANet2	0.828 ± 0.002	0.911 ± 0.001

13.3 κ_λ constraints

Table 26 is the κ_λ constraints of the different selection methods. Where the hyperparameter optimization is finished.

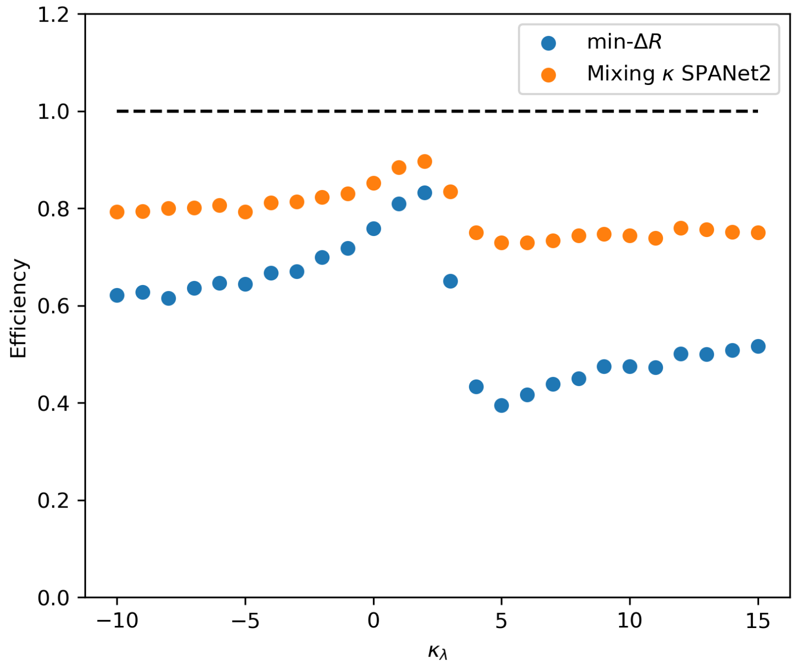


Figure 10: The pairing performance for different κ_λ samples.

Table 26: The κ_λ constraints of different selection methods.

		Expected Constraints			
		Profile likelihood		CLs	
Pairing method	Selection method	Lower	Upper	Lower	Upper
min-ΔR	DNN	-3.20	10.31	-3.16	10.19
Mixing κ SPANet2	DNN	-3.29	11.14	-3.14	11.04
Mixing κ SPANet2	SPANet2	-3.07	8.80	-3.04	8.76

14 Compared 13 TeV and 14 TeV samples

This section plots the p_{T}, η, and invariant mass distribution with different energy.

14.1 Signal plots

Generate the di-Higgs samples with $\sqrt{s}$ = 13 TeV and 14 TeV, then plot the total invariant mass of di-Higgs m_{hh} and the p_{T} of Higgs. The results are presented in Figure 11. Different energy distributions are similar and the ratio is close to 1.

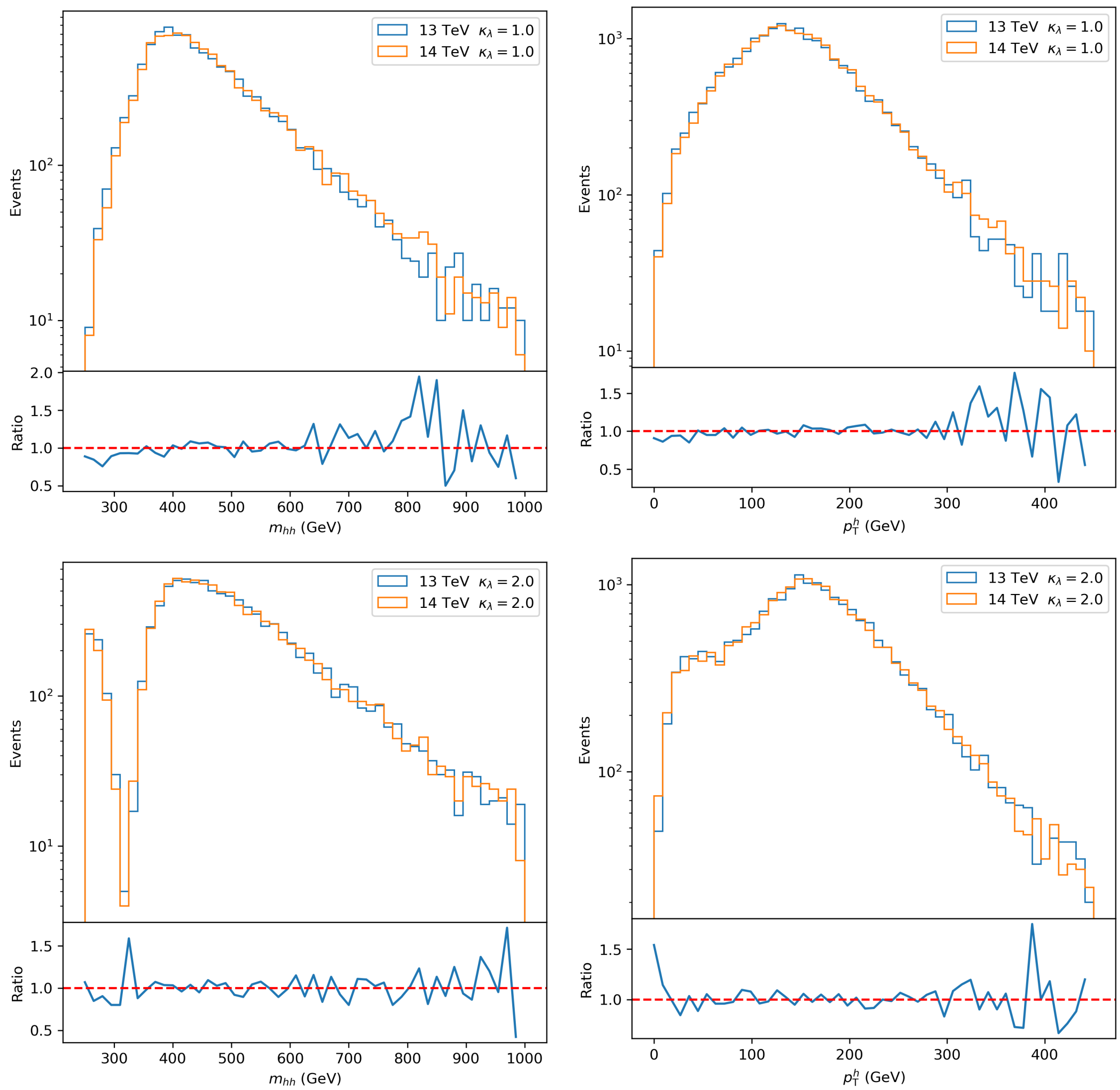


Figure 11: The total invariant mass m_{hh} distribution of di-Higgs system and the p_{T}^{h} distribution of Higgs. The ratio is obtained by dividing the 14 TeV samples by the 13 TeV samples. The distribution of different energy looks similar.

14.2 Background plots

Generate the pp4b samples with $\sqrt{s} = 13$ TeV and 14 TeV, then plot the total invariant mass of 4b quarks and the p_T, η of b quarks. The results are presented in Figure 12 and Figure 13. Different energy distribution is similar and the ratio is close to 1.

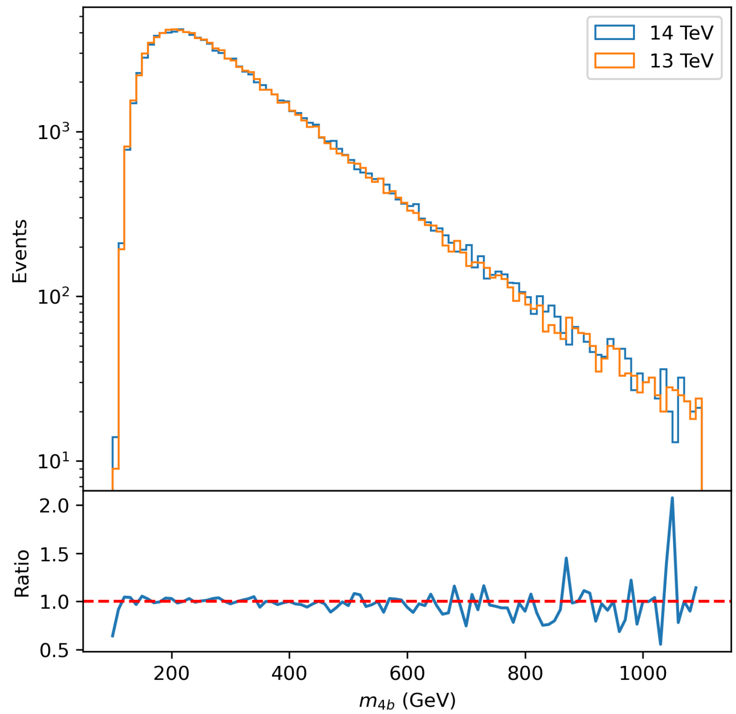


Figure 12: The total invariant mass m_{4b} distribution. The ratio is obtained by dividing the 14 TeV samples by the 13 TeV samples.

15 κ_λ constraints with different cross section and luminosity

Because the kinematics for 13 TeV and 14 TeV are similar, we can just scale the cross section to 14 TeV ones. Calculate the κ_λ constraints with 14 TeV cross section and luminosity $\mathcal{L} = 300$, 3000 fb^{-1}. Table 27 is the result.

The $\Delta\kappa_\lambda$ for different luminosity are shown in Figure 14. Where the $\Delta\kappa_\lambda$ is the upper limit minus the lower limit.

Use mixing κ SPANet2 classifier to set κ_λ constraints, then evaluate the equivalent luminosity for min-ΔR + DNN classifier. The results are shown in Table 28.

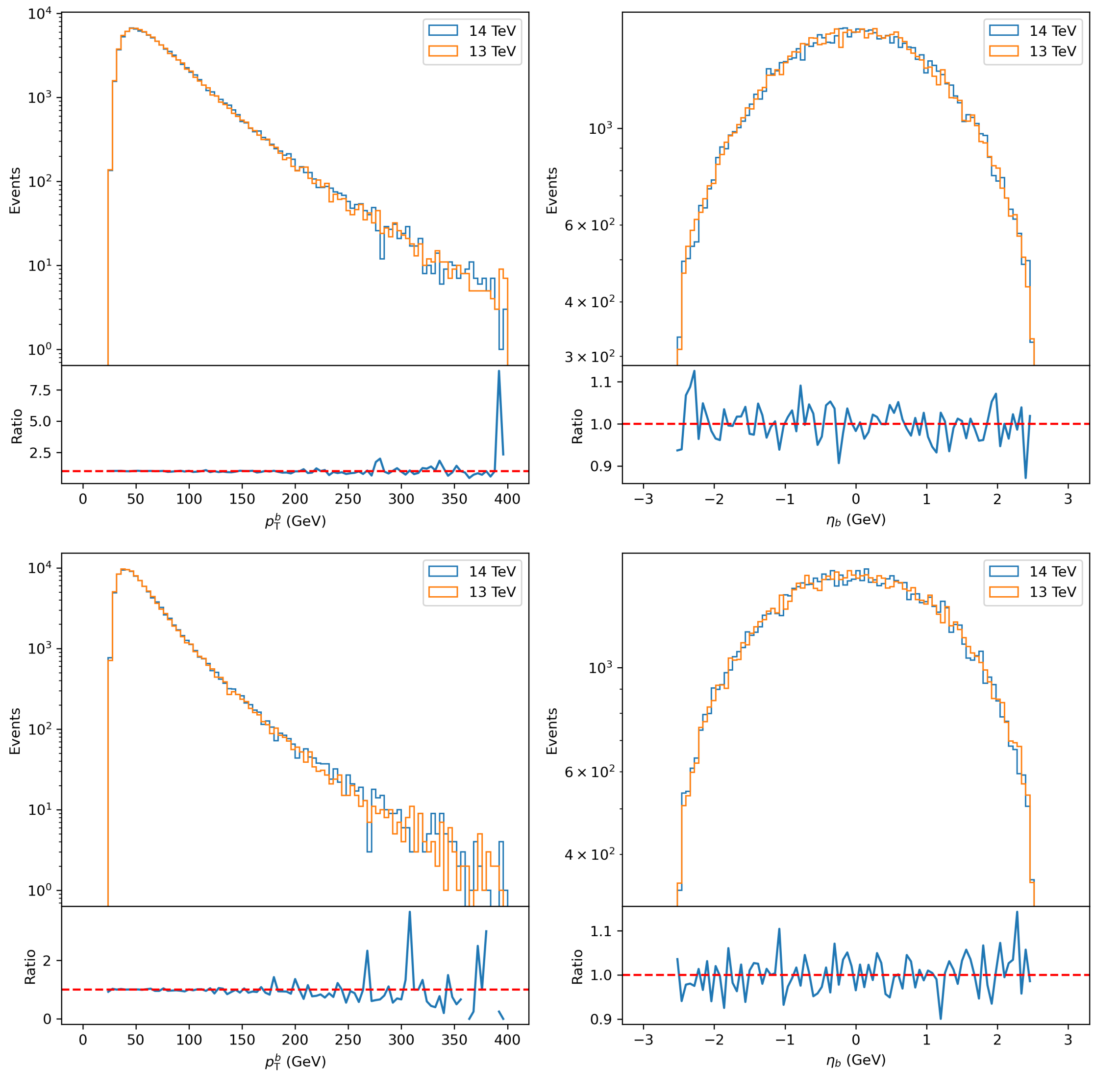
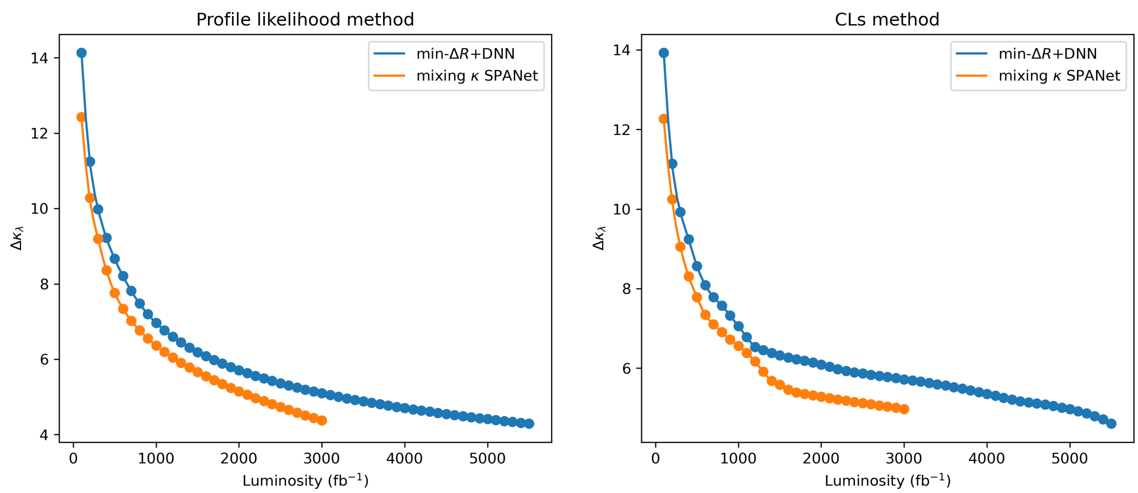


Figure 13: The p_{T} and η distribution of b quarks, the first row is for leading b quarks and the second row is for sub-leading b quarks. The ratio is the 14 TeV samples divided by the 13 TeV samples. The distribution of different energy looks similar.

Table 27: The κ_λ constraints of different selection methods. Where the 14 TeV cross sections are used.

		Expected Constraints					
		Profile likelihood			CLs		
S/B classifier	$\mathcal{L}$ (fb^{-1})	Lower	Upper	$\Delta\kappa_\lambda$	Lower	Upper	$\Delta\kappa_\lambda$
min-ΔR + DNN	300	-1.47	8.51	9.98	-1.44	8.49	9.92
min-ΔR + DNN	3000	1.01	6.11	5.10	0.36	6.08	5.72
Mixing κ SPANet2 + DNN	300	-1.75	9.03	10.78	-1.72	8.94	10.67
Mixing κ SPANet2 + DNN	3000	0.87	6.23	5.36	0.33	6.20	5.87
Mixing κ SPANet2	300	-1.59	7.60	9.19	-1.57	7.49	9.06
Mixing κ SPANet2	3000	0.91	5.29	4.38	0.33	5.31	4.98



(a) (b)

Figure 14: The $\Delta\kappa_\lambda$ for different luminosities.

Table 28: Use mixing κ SPANet2 classifier to set κ_λ constraints. The equivalent luminosity for min-ΔR + DNN are presented. Where the $\Delta\kappa_\lambda$ are used to evaluate the equivalent luminosity.

	Expected Constraints							
	Profile likelihood			CLs			Profile likelihood	CLs
$\mathcal{L}$ (fb^{-1})	Lower	Upper	$\Delta\kappa_\lambda$	Lower	Upper	$\Delta\kappa_\lambda$	Equivalent $\mathcal{L}$ for min-ΔR + DNN	
300	-1.59	7.60	9.19	-1.57	7.49	9.06	405	425
3000	0.91	5.29	4.38	0.33	5.31	4.98	5131	4992

16 Resonant samples with various mass

This section trains a SPANet by mixing m_H samples, and then using it on the analysis. The various m_H value samples are generated, and then mix it for training.

16.1 Training results

Set m_H from 300 GeV to 1200 GeV with step size 100 GeV, for each mass point generate $100,000$ samples. The training samples are required to pass the "Four tag cut", i.e., there are at least four b-tagged jets with $p_{\mathrm{T}} > 40$ GeV and $|\eta| < 2.5$. The b-tagging efficiency is the same as the MV2c10 b-tagger at 70% WP. Then mix these samples for training.

- Training sample:
  - Total sample size: 1,000,000
  - 1h sample size: 139,704
  - 2h sample size: 848,708
  - 5% used on validation
- Testing sample:
  - Total sample size: 100,000
  - 1h sample size: 13,974
  - 2h sample size: 84,845

The training results are presented in Table 29.

Table 29: SPANet training results on the mixing m_H samples.

N_{Jet}	Event Efficiency	Higgs Efficiency
$= 4$	0.979	0.979
$= 5$	0.930	0.952
≥ 6	0.859	0.912
Total	0.928	0.951

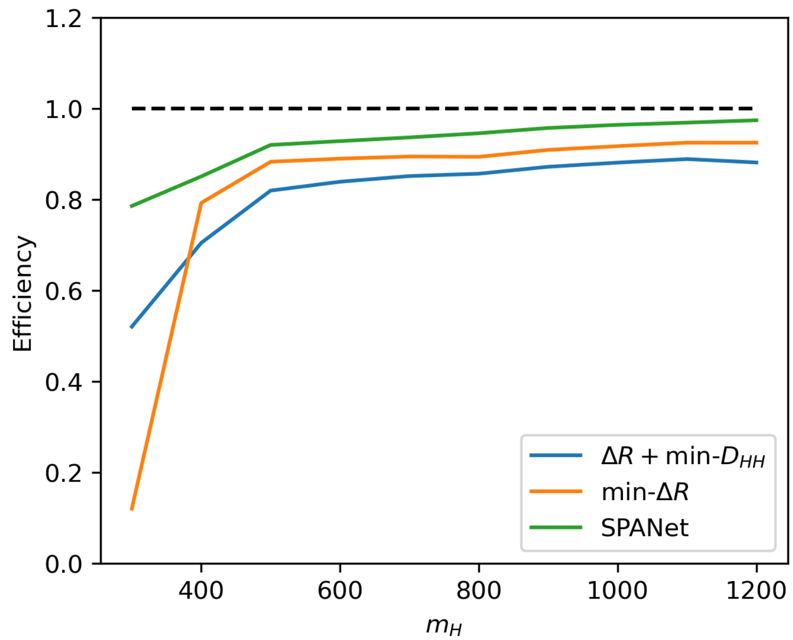


Figure 15: The pairing performance for different m_H samples.

16.2 Pairing performance

Figure 15 shows the pairing efficiency of different methods.

Figure 16 shows the pairing efficiency of the min-ΔR method. The 330 GeV and 360 GeV samples are added to make sure that pairing efficiency quickly decreases as the mass decreases in range from 300 GeV to 400 GeV.

Figure 17 shows the pairing efficiency for different numbers of jets.

16.3 Sensitivity

Figure 18 shows the sensitivity of different methods.

Figure 19 shows the Higgs mass distribution for SPANet pairing. For signal, the results are similar to before, but the shape is sculpted for the background.

16.4 Event classification

To reduce the sculpting effect, SPANet is trained on jet pairing and event classification tasks simultaneously.

The signal sample is the same as Section 16.1. The background sample is required to pass the Four-tag cut.

For the jet assignment part,

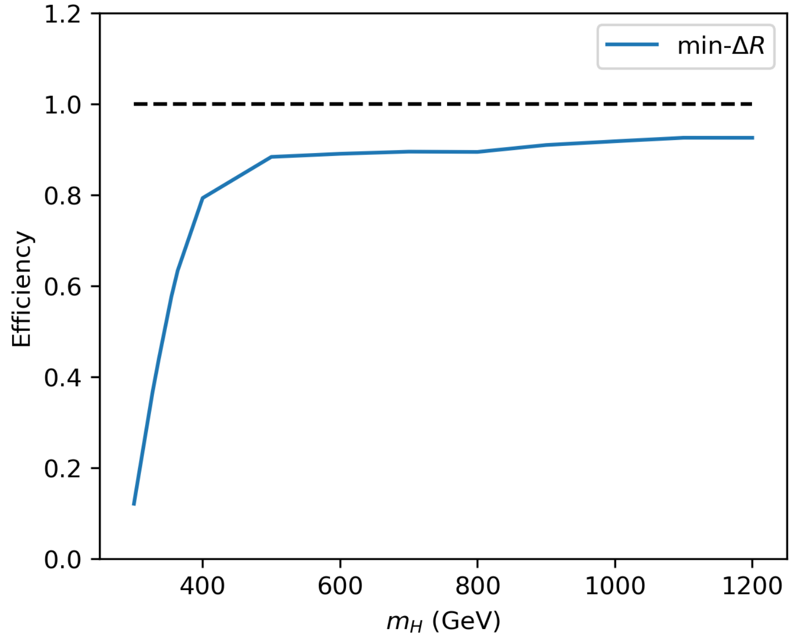


Figure 16: The pairing performance for different m_H samples.

- Training sample:
  - Total sample size: 500,000
  - 1h sample size: 69,736
  - 2h sample size: 424,493
  - 5% used on validation
- Testing sample:
  - Total sample size: 50,000
  - 1h sample size: 6,851
  - 2h sample size: 42,556

For event classification,

- Training sample:
  - Total sample size: 1,000,000
  - Signal sample size: 500,000
  - Background sample size: 500,000

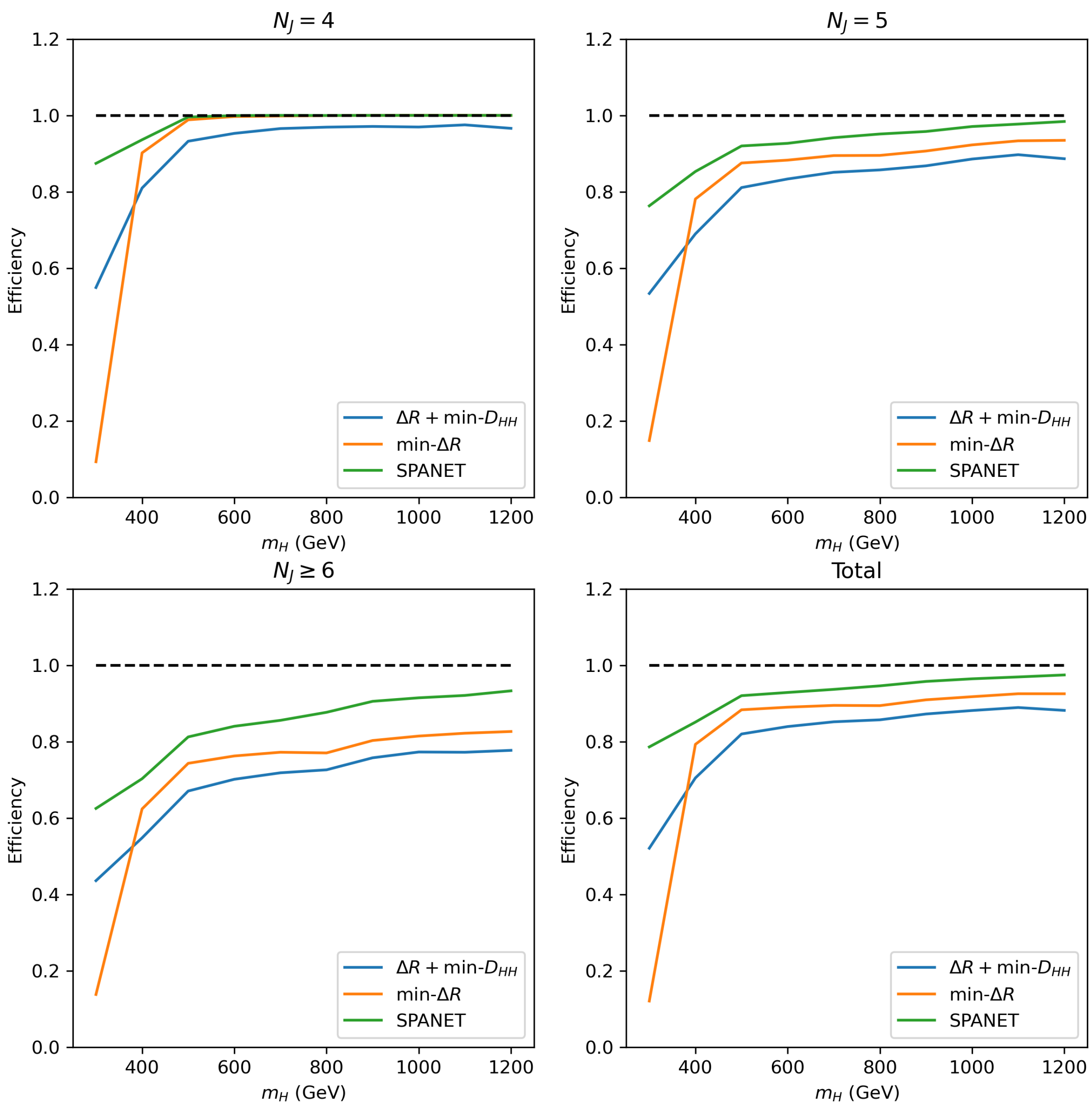


Figure 17: The pairing performance for different m_H samples.

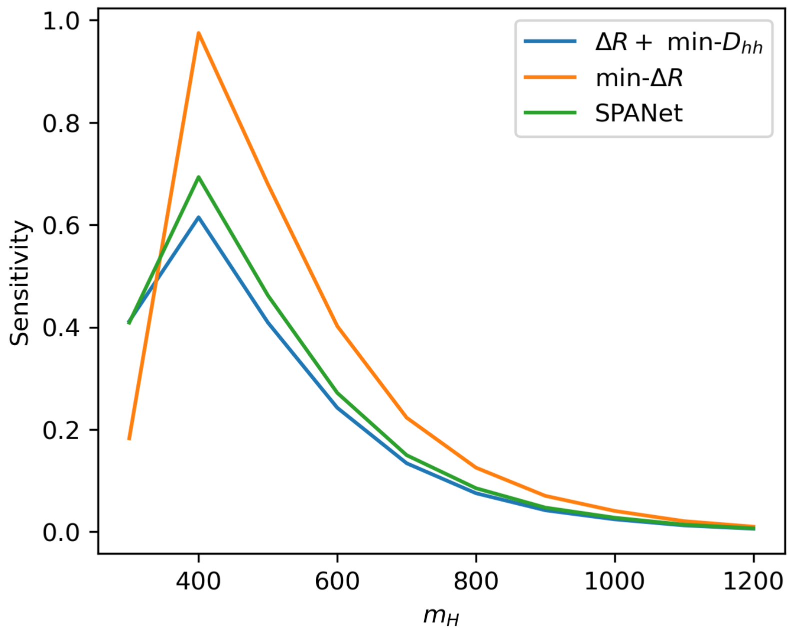


Figure 18: The sensitivity for different m_H samples.

  - 5% used on validation

- Testing sample:
  - Total sample size: 100,000
  - Signal sample size: 50,000
  - Background sample size: 50,000

The training results are presented in Table 30. The training result is close to the previous one (Table 29).

Table 30: SPANet training results on the mixing m_H samples. The SPANet is trained on jet pairing and event classification tasks at the same time.

N_{Jet}	Event Efficiency	Higgs Efficiency
$= 4$	0.978	0.978
$= 5$	0.922	0.945
≥ 6	0.847	0.904
Total	0.922	0.946

Table 31 presents the classification training results.

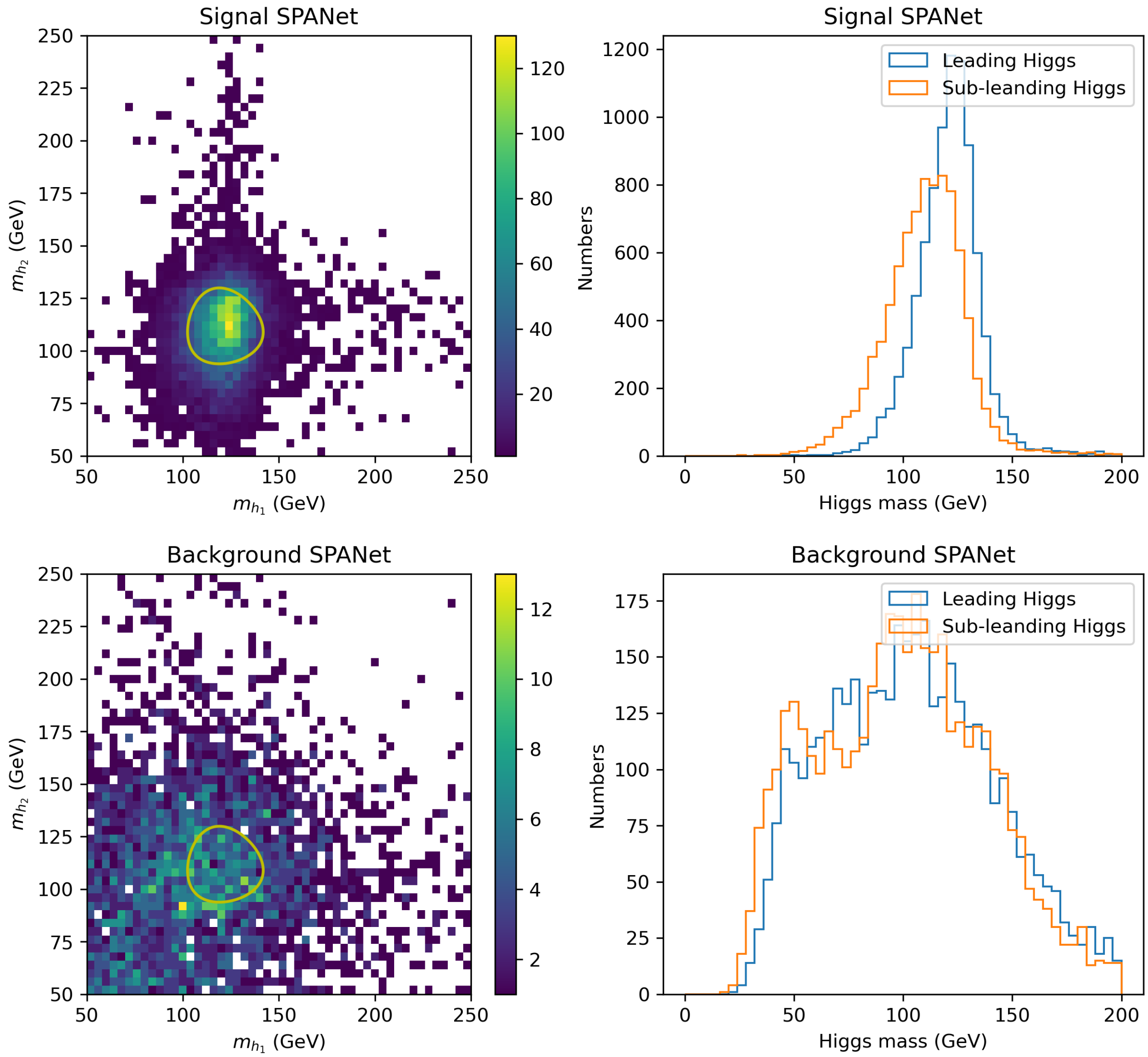


Figure 19: The mass plane and distribution of Higgs candidates for the SPANet pairing. SPANet is trained on mixing mass samples. The above figure is for the $m_H = 1000$ GeV signal sample and the below one is for the background sample.

Table 31: The SPANet classification training results with mixing m_H sample.

	ACC	AUC
SPANet	0.897	0.963

Figure 20 shows the pairing efficiency of different methods. The pairing performance of SPANet is similar to the previous one. In this section, the SPANet is trained on jet pairing and event classification tasks at the same time, but in Section 16.1 the SPANet is only trained on the jet pairing task.

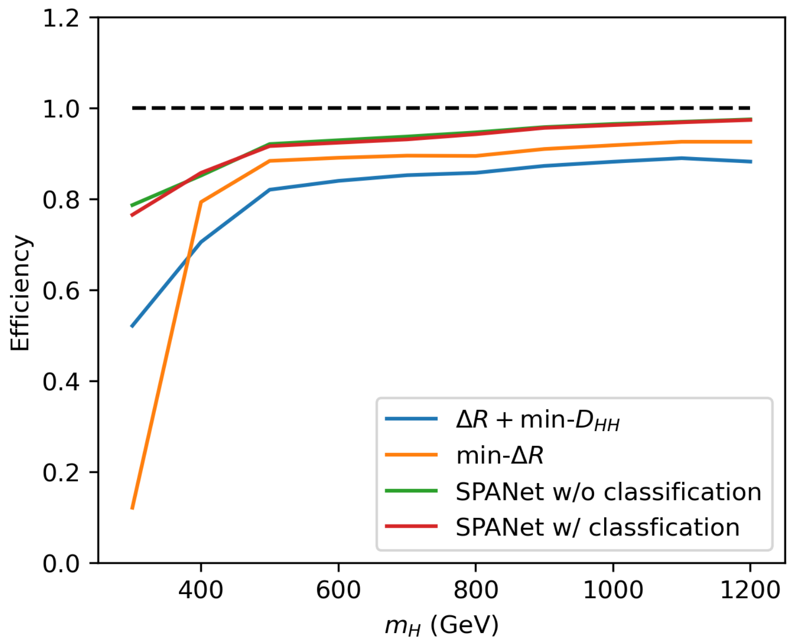


Figure 20: The pairing performance for different m_H samples. "SPANet w/o classification" is the SPANet only trained on jet pairing task. "SPANet w/classification" is the SPANet trained on jet pairing and event classification tasks at the same time.

Figure 21 shows the sensitivity of different methods. The sensitivity of "SPANet with classification" is similar to the previous one.

Figure 22 shows the Higgs mass distribution for SPANet pairing. The shape is still sculpted for the background sample.

16.5 SPANet classifier

In this section, SPANet is used for event selection. For each event, SPANet will give a signal score p_{signal}, representing the confidence that this event is a signal event. The requirement of $p_{\text{signal}} > p_{\text{th}}$ is imposed for event selection, where $p_{\text{th}} = 0.90$.

Figure 23 shows the sensitivity of different methods. The sensitivity of "SPANet classifier" has the best performance.

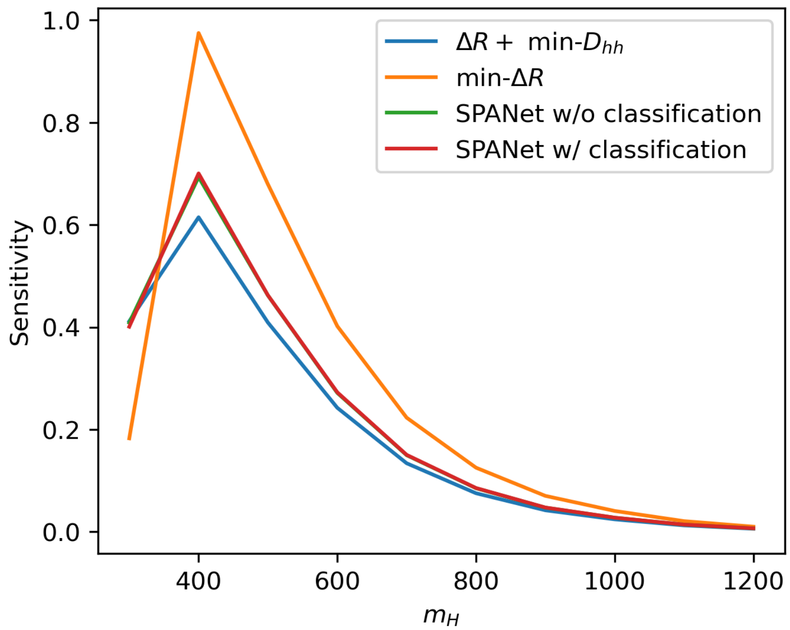


Figure 21: The sensitivity for different m_H samples.

16.6 Hyperparameter optimization

The optimization range of different hyperparameters is listed in the below

- `learning_rate`: $[10^{-5}, 10^{-2}]$
- `dropout`: $[0, 0.5]$
- `gradient_clip`: $[0, 0.5]$
- `l2_penalty`: $[0, 0.0005]$
- `hidden_dim`: $[16, 32, 64, 128, 256]$
- `num_encoder_layers`: $[2, 8]$
- `num_branch_encoder_layers`: $[2, 8]$
- `num_classification_layers`: $[1, 5]$

Each parameter set was trained for 10 epochs. Test 100 trials.

The hyperparameter optimization results are listed below

- `learning_rate`: 0.00049
- `dropout`: 0.061

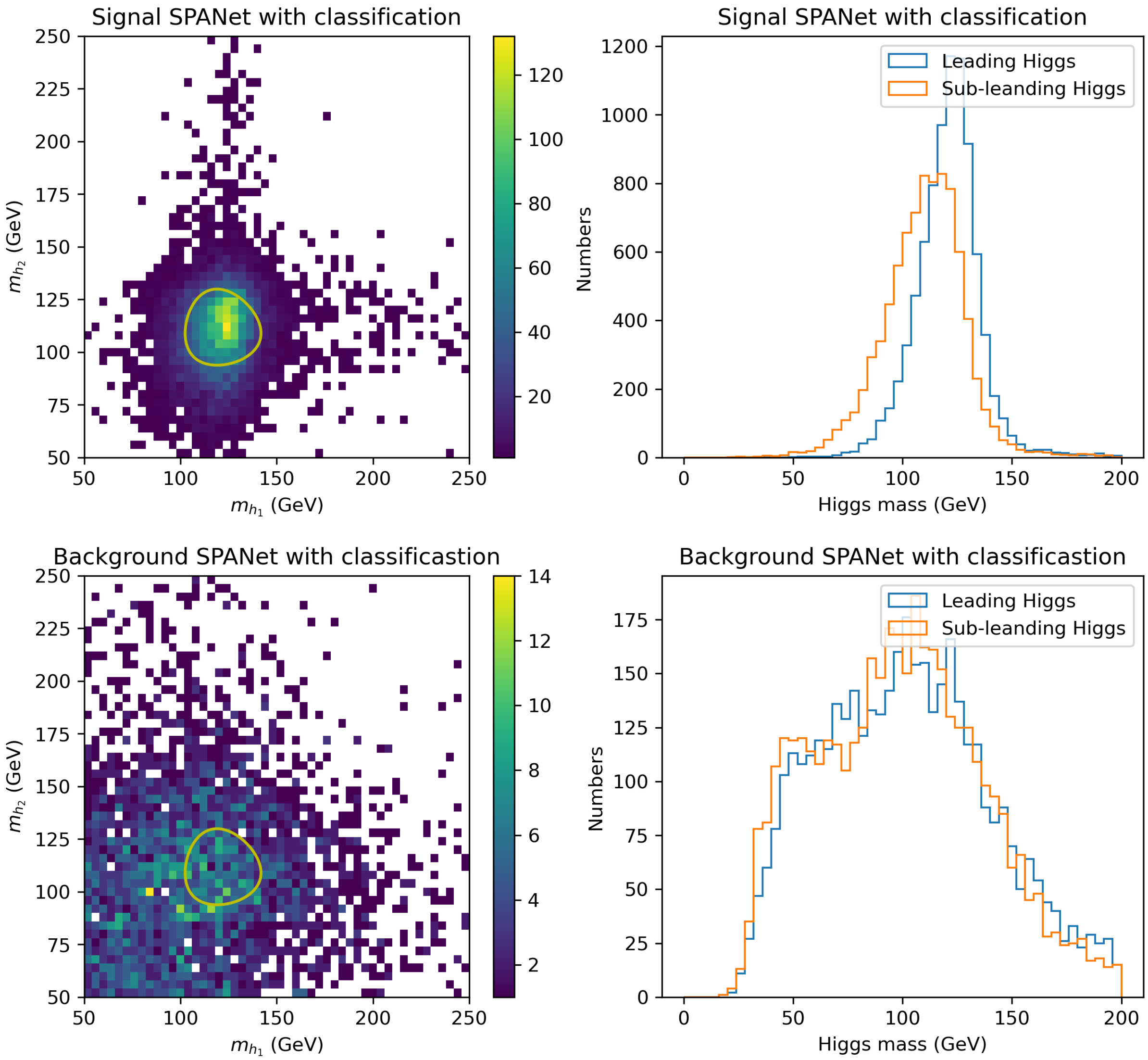


Figure 22: The mass plane and distribution of Higgs candidates for the SPANet pairing. SPANet is trained on mixing mass samples. The above figure is for the $m_H = 1000$ GeV signal sample and the below one is for the background sample.

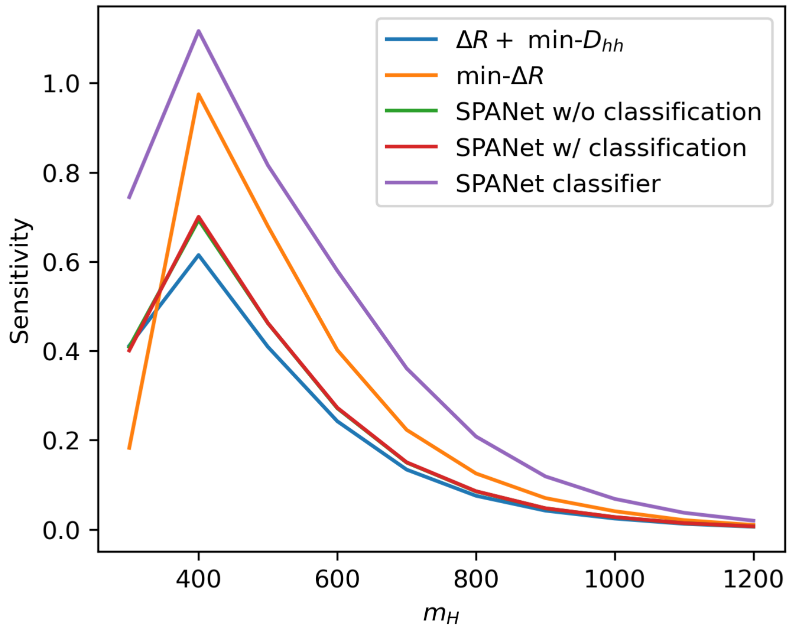


Figure 23: The sensitivity for different m_H samples. "SPANet classifier" uses SPANet signal score for event selection.

- `gradient_clip`: 0.445
- `l2_penalty`: 0.000382
- `hidden_dim`: 256
- `num_encoder_layers`: 8
- `num_branch_encoder_layers`: 4
- `num_classification_layers`: 1

Use this parameter set for full training. Table 32 presents the classification training results. This result is similar to Table 31.

Table 32: The SPANet classification training results with mixing m_H sample. Use the Optuna hyperparameter optimization results.

	ACC	AUC
Best HP	0.893	0.958

16.7 Optimize threshold

This section scans the threshold p_{th} from 0 to 1 with step size 0.01. Choose the p_{th} which can maximize the sensitivity $S/\sqrt{B}$. The results are summarized in Table 33. For $m_H = 300$ GeV, the p_{th} is much lower than other mass points.

Table 33: The threshold p_{th} optimization results. The number of signal events S and background events B are computed at luminocity $\mathcal{L} = 139\ \text{fb}^{-1}$. The p_{th} is optimized at each mass point.

m_H (GeV)	p_{th}	S	B	$S/\sqrt{B}$
300	0.57	257	87112	0.872
400	0.88	183	25426	1.148
500	0.89	127	23678	0.828
600	0.97	56	8301	0.616
700	0.99	26	2621	0.504
800	0.99	19	2621	0.376
900	0.99	13	2621	0.246
1000	0.99	8	2621	0.154
1100	0.99	4	2621	0.086
1200	0.99	2	2621	0.046

Figure 24 shows the signal score distributions. When m_H is set to 300 GeV, the signal score distribution appears to be more evenly spread compared to other cases. As the resonant mass increases, a larger number of events tend to achieve higher signal scores, resulting in higher threshold values.

Figure 25 shows the sensitivity of different methods. We add the sensitivity of the "SPANet classifier" with optimize p_{th}.

16.8 Cross section upper limit

This section utilizes the m_{hh} distribution to set the upper limit of cross section. The CL_s method is used. The signal strength is chosen as the parameter of interest. The POI is excluded at the 95% confidence level when the CL_s is less than 0.05. Where the package `pyhf` is used to calculate the upper limit.

Parameter setting:

- Number of bins: 11

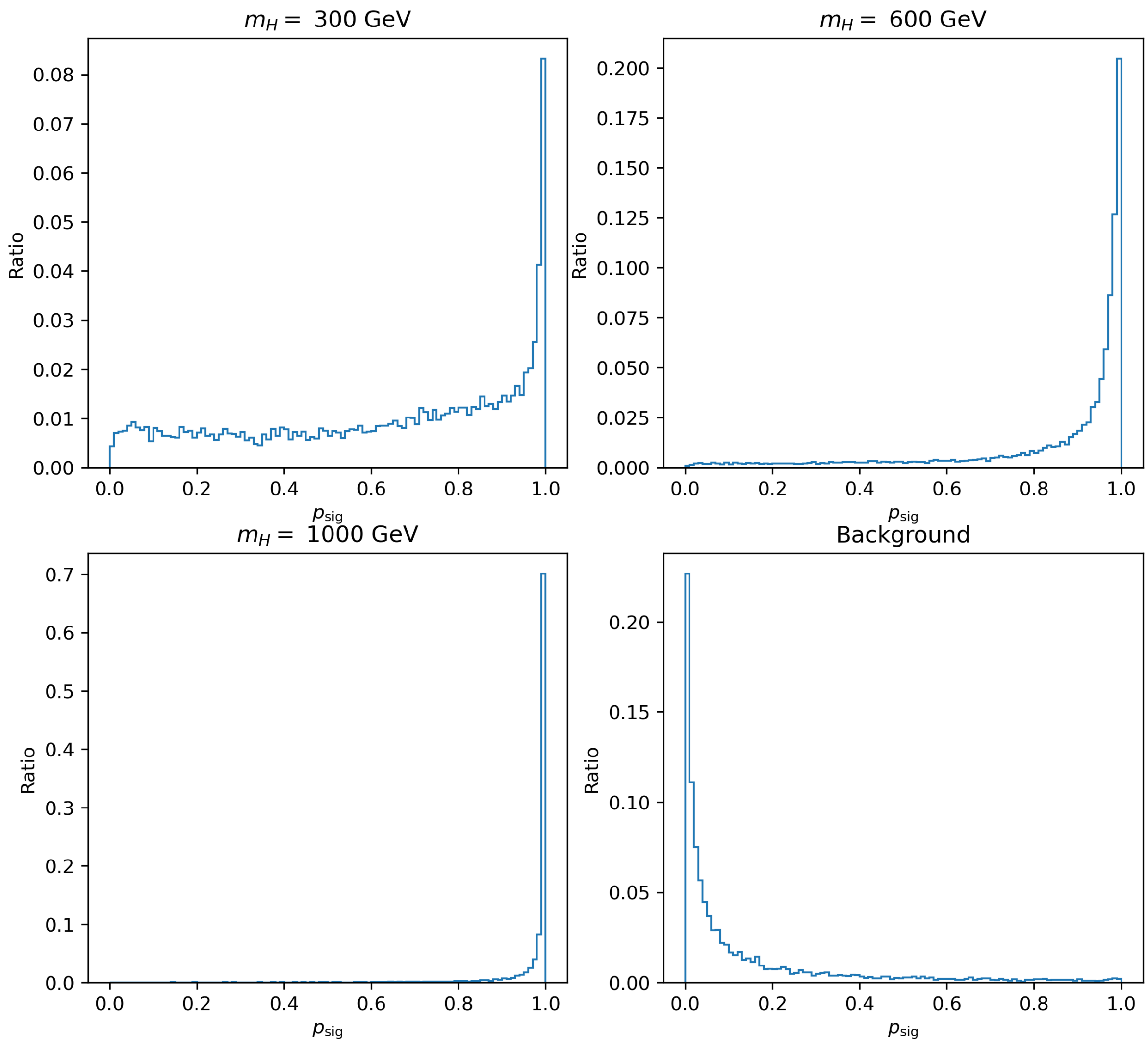


Figure 24: The signal score p_{sig} distribution for signal and background samples.

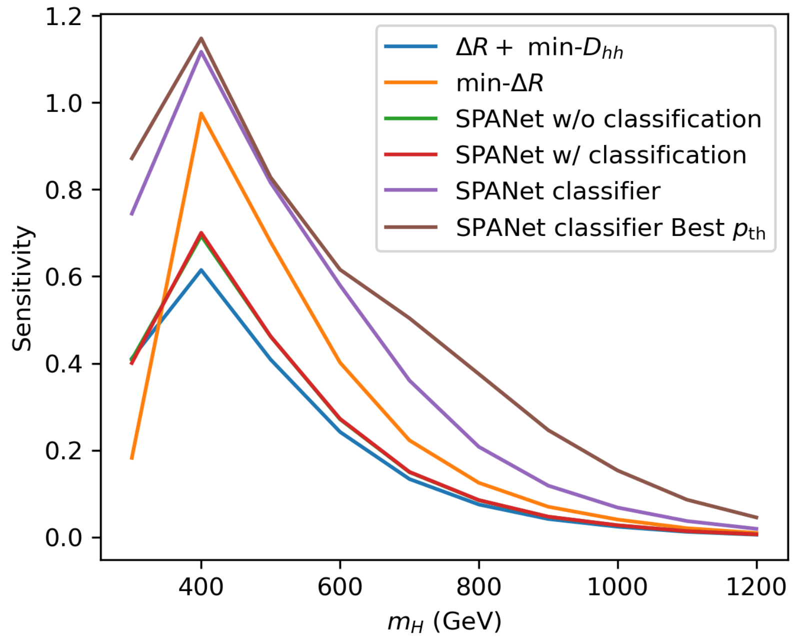


Figure 25: The sensitivity for different m_H samples. "SPANet classifier Best p_{th}" uses SPANet signal score for event selection with the optimized thresholds p_{th}.

- Range: $[200, 1300]$
- Luminosity: $\mathcal{L} = 139\ \mathrm{fb}^{-1}$

Figure 26 is the upper limits scans for m_H with different selection method. In low mass range, SPANet selection provides the most stringent constraints, while in the high mass range, all methods give similar results.

The invariant mass m_{hh} distribution is shown in Figure 27.

16.9 SPANet Fine-tuning

To implement "SPANet Fine-tuning", first we need to train a SPANet on pairing and classification tasks at the same time. Use this SPANet we can obtain the embedding vector for each event. Then utilize this embedding as the the inputs to train a signal background classifier.

The SPANet trained in Section 16.4 is used to generate the embedding vectors. For the signal background classifier, dense neural network (DNN) is used. The DNN has two internal layers each with 256 nodes.

Table 34 presents the classification training results. The results are similar to SPANet pretraining (Table 31).

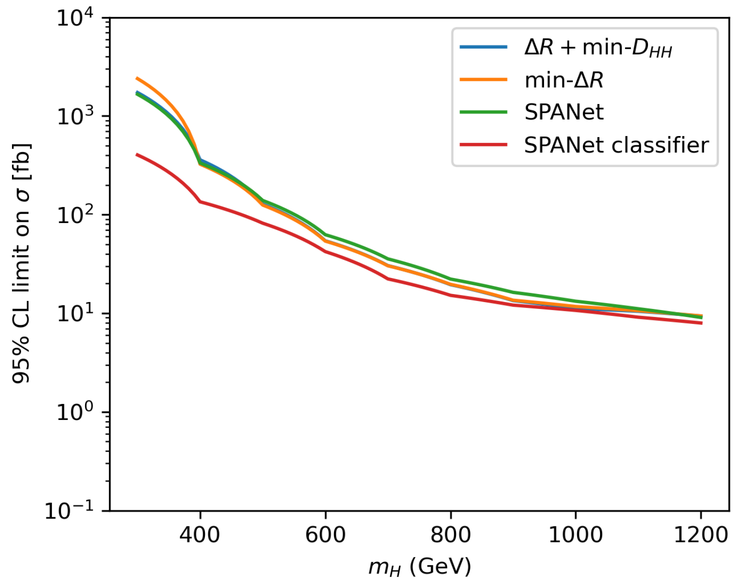


Figure 26: The upper limit of the cross-section with different m_H. The "SPANet classifier" utilize the SPANet's outputs for event selection, while other methods utilize cut-based selection with various pairing methods.

Table 34: The SPANet fine-tuning classification training results with mixing m_H sample.

	ACC	AUC
SPANet fine-tuning	0.897	0.963

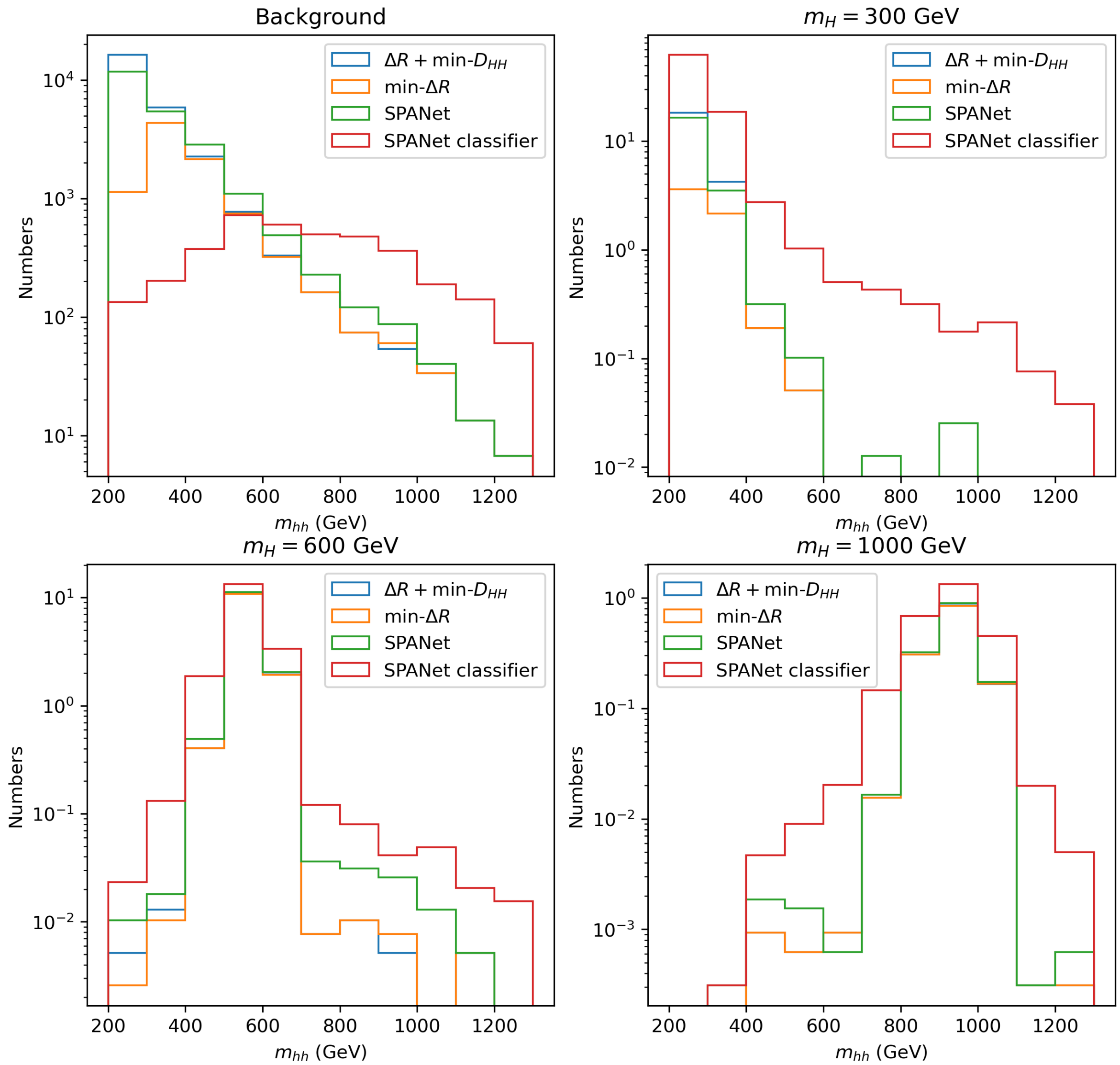


Figure 27: The invariant mass m_{hh} distribution with different selection methods.

Use the same selection method as in Section 16.5 with the optimized thresholds. Then, utilize the m_{hh} distribution to set the upper limit of cross section. The results is shown in Figure 28. The upper limits of "SPANet fine-tuning" are similar the results of "SPANet classifier".

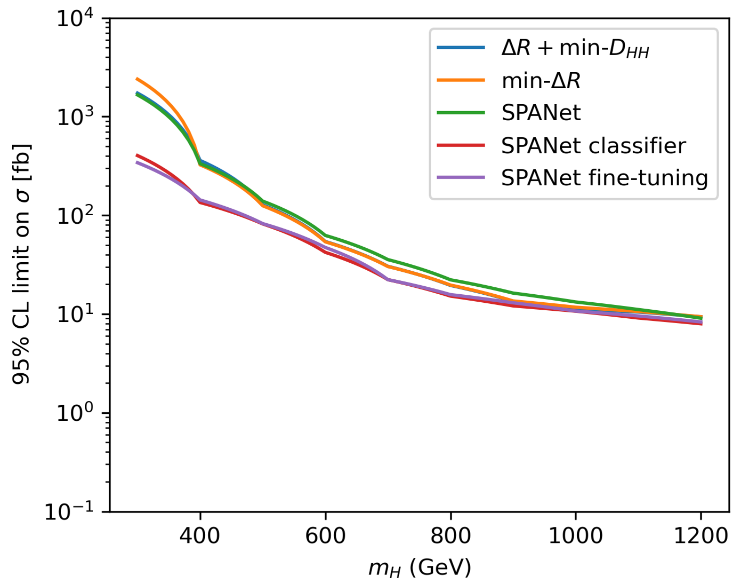


Figure 28: The upper limit of the cross-section with different m_H. "SPANet classifier" and "SPANet fine-tuning" utilize the SPANet's outputs for event selection, while other methods utilize cut-based selection with various pairing methods.