

---

# ПРИМЕНЕНИЕ МАШИННОГО ОБУЧЕНИЯ К АНАЛИЗУ ПРОЦЕССОВ РОЖДЕНИЯ БОЗОНА ХИГГСА В АССОЦИИ С ТОП-КВАРКОМ

---

Диденко Алиса



Научные руководители:

*Наумов Дмитрий Вадимович*, профессор кафедры ФЭЧ, зам. директора ЛЯП ОИЯИ, доктор ф.-м. наук  
*Елецких Иван Владимирович*, старший научный сотрудник ЛЯП ОИЯИ, кандидат ф.-м. наук



# Цели и Задачи

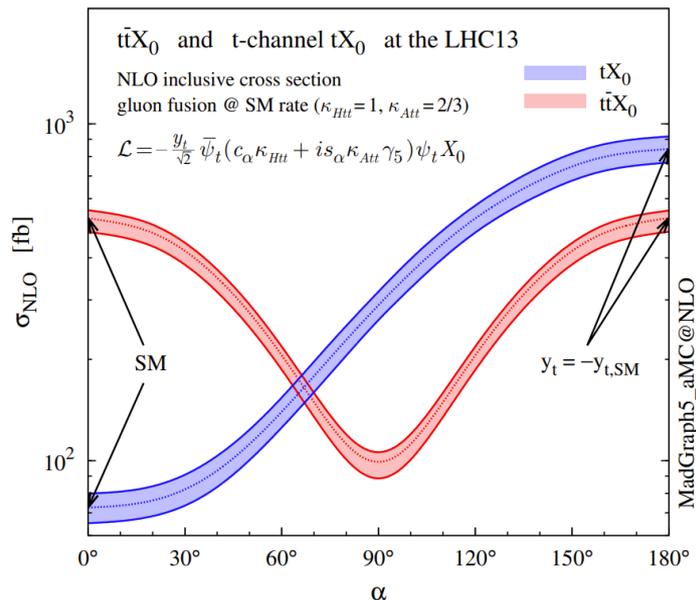
---

**Целью** этой работы является экспериментальное изучение процессов рождения бозона Хиггса в ассоциации с одним топ-кварком ( $pp \rightarrow tH$ ), в частности, изучение возможностей кинематического отделения сигнального процесса от фоновых и фоновых процессов между собой с помощью глубокого машинного обучения.

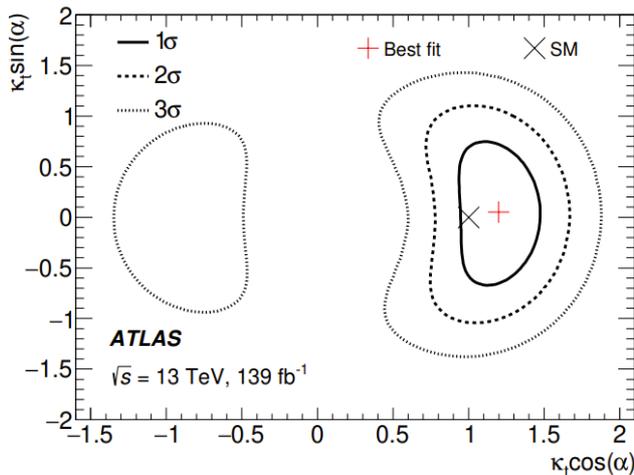
## **Задачи:**

1. Изучить существующие экспериментальные и теоретические результаты, касающиеся процесса рождения бозона Хиггса в ассоциации с одним топ-кварком в протон-протонных столкновениях на LHC;
2. Исследовать кинематику сигнального и фоновых процессов;
3. Изучить существующее программное обеспечение глубокого машинного обучения, применяемого для классификации событий в экспериментах физики высоких энергий;
4. Применить нейронную сеть для классификации событий сигнального и фоновых процессов в эксперименте ATLAS и оптимизировать настройки алгоритма с точки зрения скорости обучения и эффективности классификации;
5. На основе полученных результатов определить сигнальную область, где достигается максимальный вклад сигнала  $tH$ ;
6. Сравнить результаты с другими подходами к классификации данных.

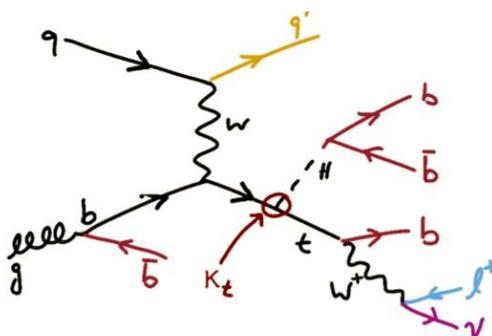
# Процесс рождения бозона Хиггса и топ-кварка. Константа взаимодействия



Сечение рождения  $tH$  и  $ttH$  в р-столкновениях при энергии 13 ТэВ в зависимости от угла CP-смешивания  $\alpha$ . Где значение  $\alpha = 0$  соответствует Стандартной модели.

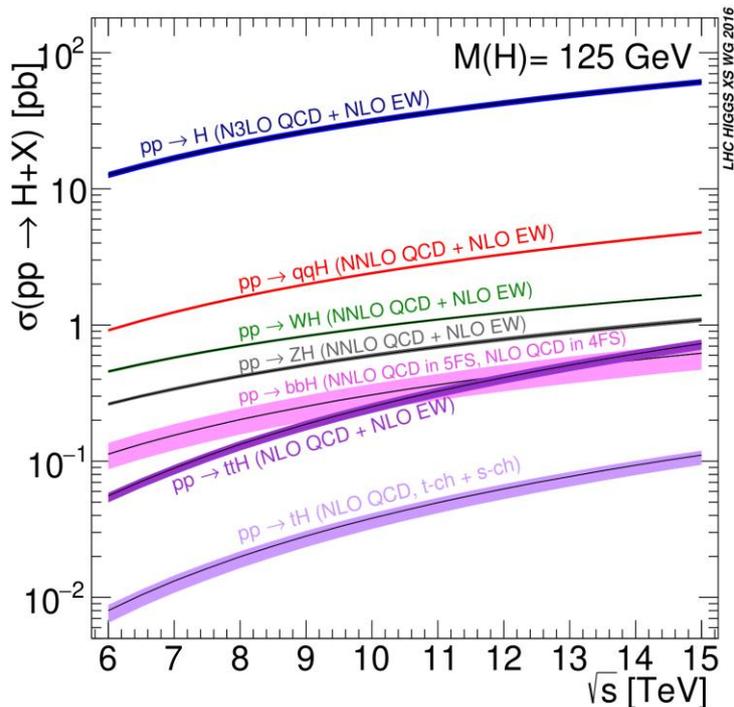


$\alpha = 180$  исключена на уровне 2.5  $\sigma$ .  
 $\alpha$  более 43 градусов исключена с уровнем достоверности 95%



Сигнатура  $tH$  с распадом  $H \rightarrow b\bar{b}$ :

( $\geq 3$  b-струи) + (1 легкая струя) + (1 лептон) + (недостающий поперечный импульс)



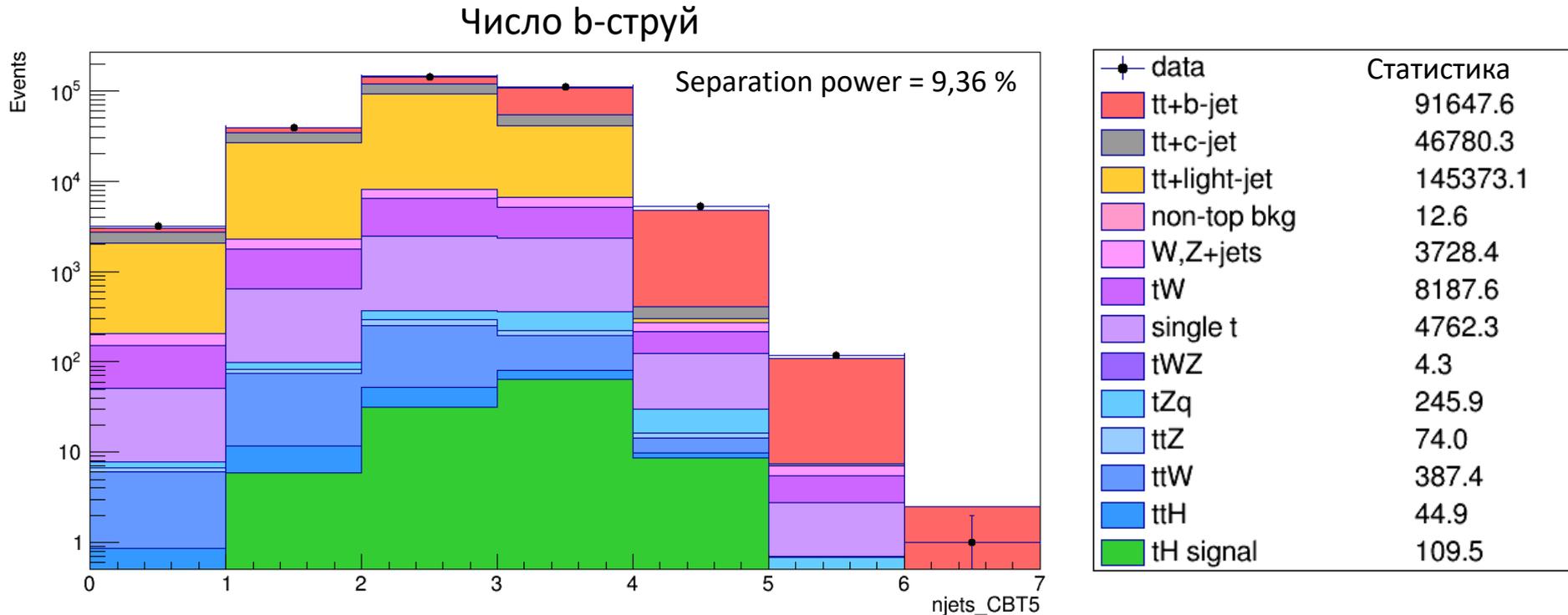
Зависимость сечения рождения  $H_{SM}$  от энергии р-столкновений.

# Кинематика сигнального и фоновых процессов

Сигнальный и фоновые процессы различаются по большому числу кинематических переменных, но для каждой из них различия невелики.

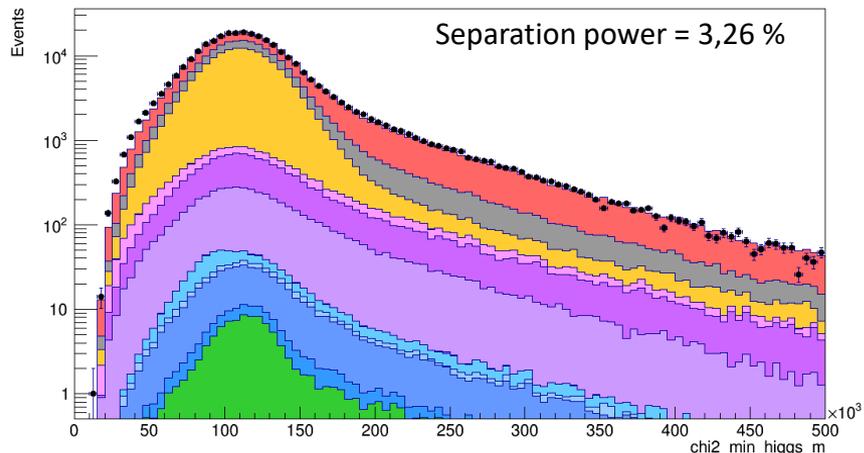
Задача разделения может быть решена с помощью машинного обучения.

Для событий, прошедших преселекцию, проверяется качество моделирования распределений основных кинематических переменных.

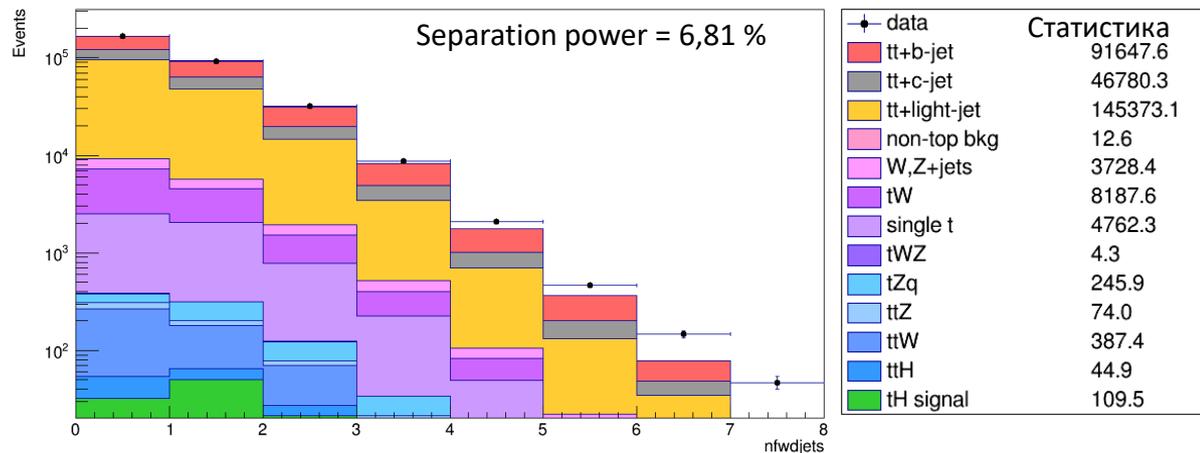


# Кинематика сигнального и фонового процессов

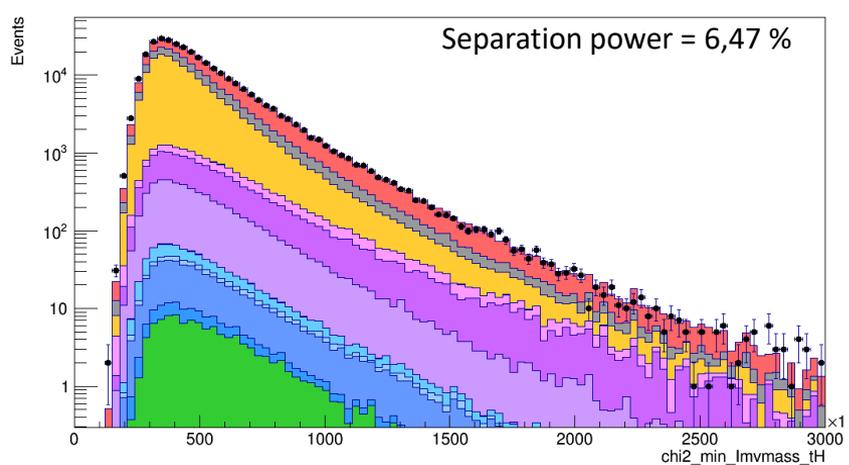
## Реконструированная масса бозона Хиггса



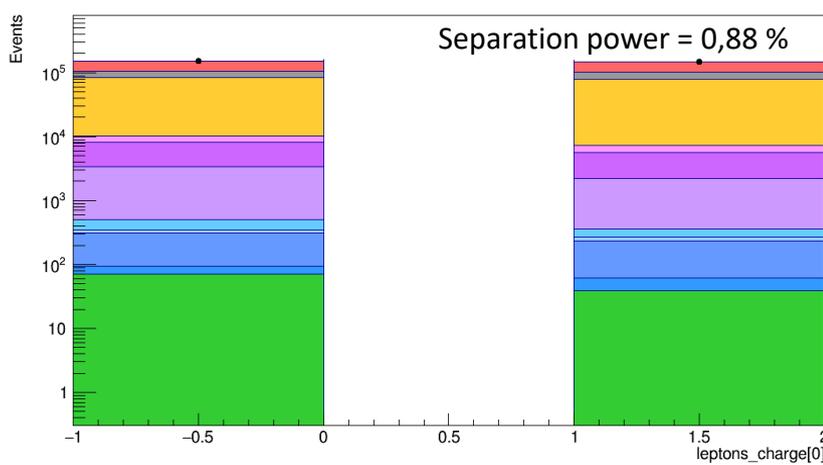
## Число струй в передней области детектора



## Инвариантная масса tH



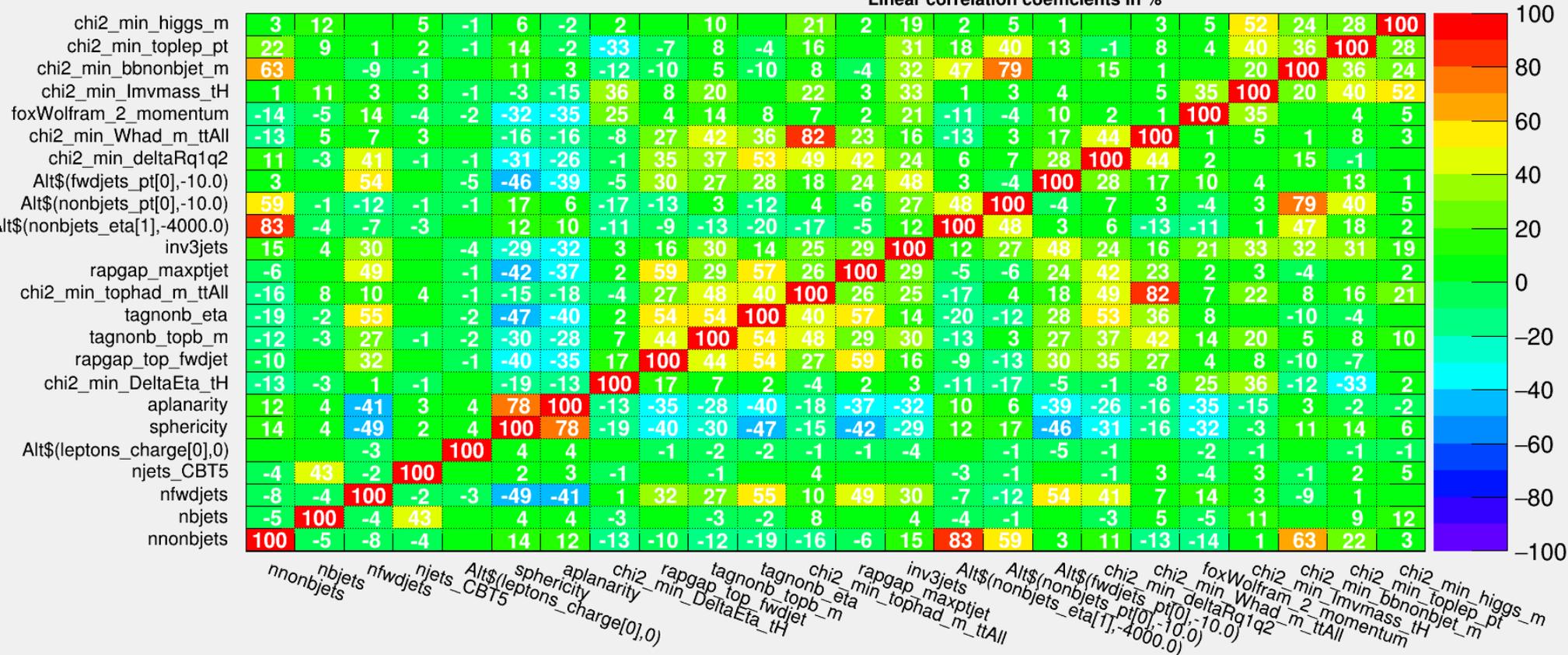
## Заряд лидирующего лептона



# Задача разделения сигнала и фонов. Выбор набора переменных. Матрица корреляции

## Correlation Matrix (signal)

Linear correlation coefficients in %

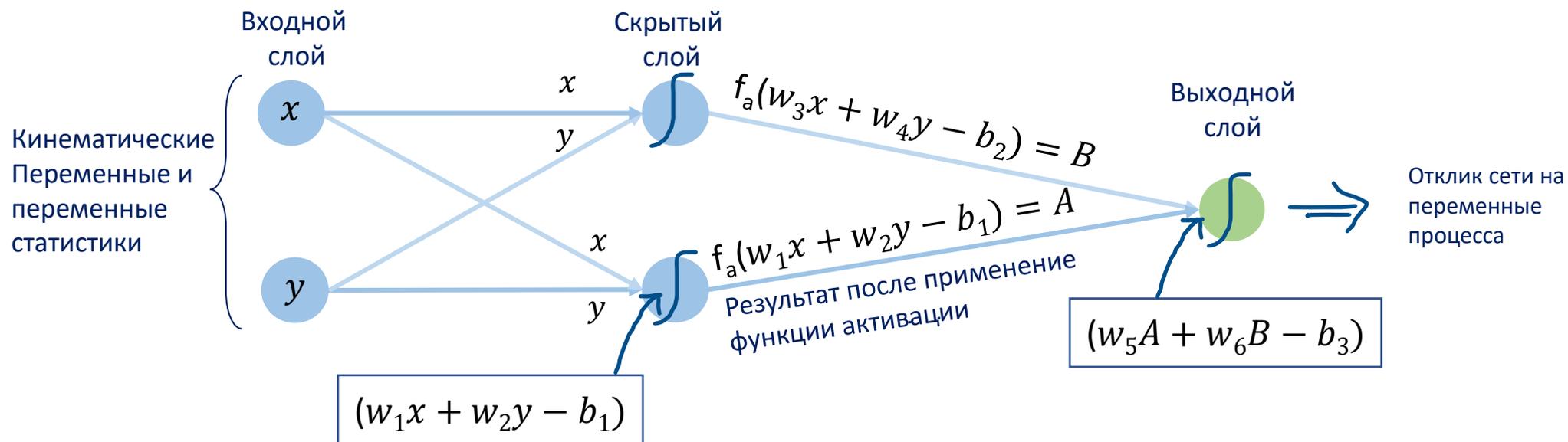


В анализе используется до 30 переменных, наиболее чувствительных к основным процессам. Корреляция < 88 %

**Нейронная сеть** – функция с большим числом варьируемых параметров. Оптимальные значения этих параметров обеспечивают наилучшее разделение сигнала и фона.

**Варьируемые (свободные) параметры:** веса ( $w_1 - w_6$ ) и смещения ( $b_1 - b_3$ ).

Пример простой нейронной сети:



Есть различные алгоритмы поиска значений параметров – **алгоритмы обучения**.

Реально применяемые алгоритмы пытаются комбинировать достоинства детерминистических и стохастических методов. В большинстве задач удовлетворительным решением является достаточно глубокий локальный минимум.

Алгоритм минимизирует меру отличия между «истинным» значением целевой переменной и предсказанным нейронной сетью значением.



Профиль оптимизируемой функции

## Внешние параметры сети

Количество слоев

Число обучающих итераций

Количество нейронов в слое

Размер обучаемой за раз партии параметров

Входные переменные

Функции активации и оптимизации

## Алгоритмы обучения сети

Абсолютно случайный поиск

Градиентный спуск (производная минимизируемой функции)

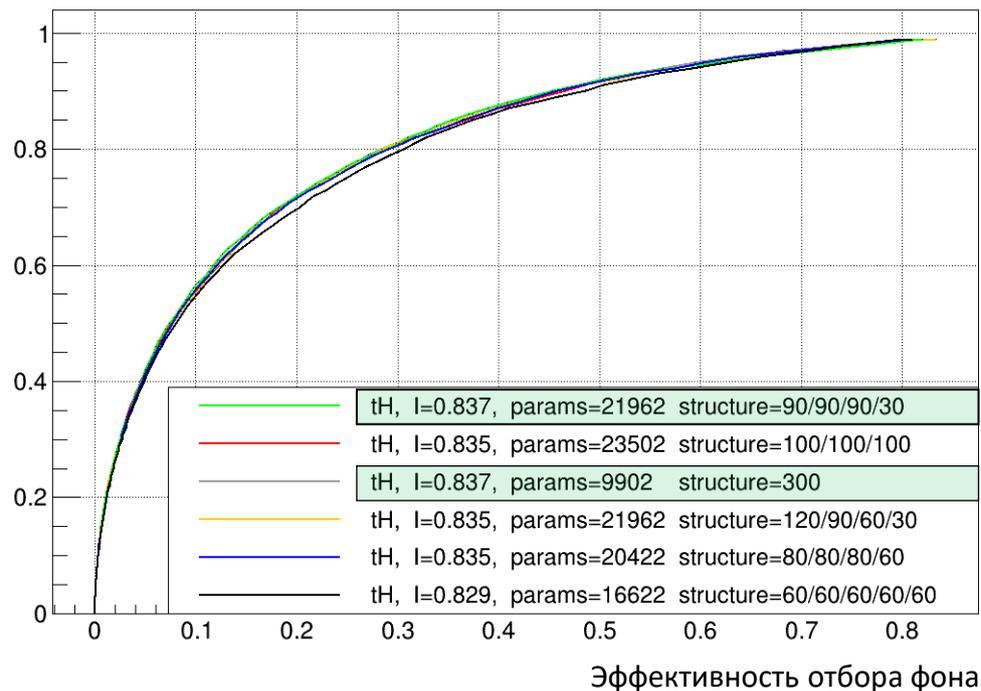
# Графики эффективности отделения сигнала от фона. Выбор оптимальной структуры сети.

Тестировались неглубокие сети (например 300), а также глубокие сети (например 150-150 и 90-90-90-30).

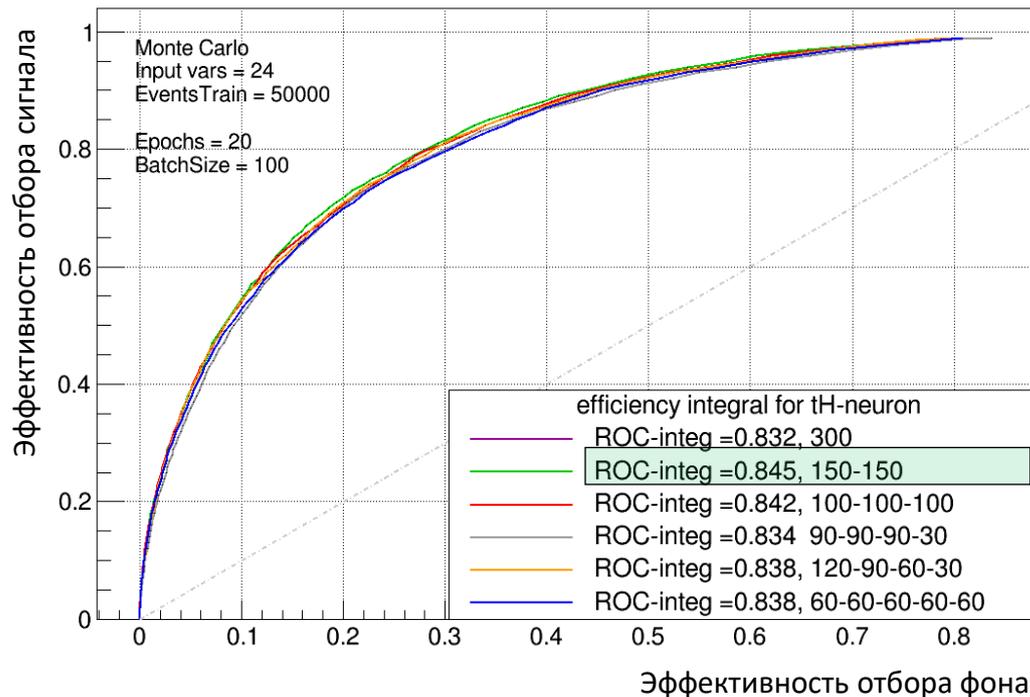
В зависимости от набора входных переменных оптимальная структура сети отличается.

Качество отделения сигнала можно оценивать по интегральному значению кривой эффективности отделения.

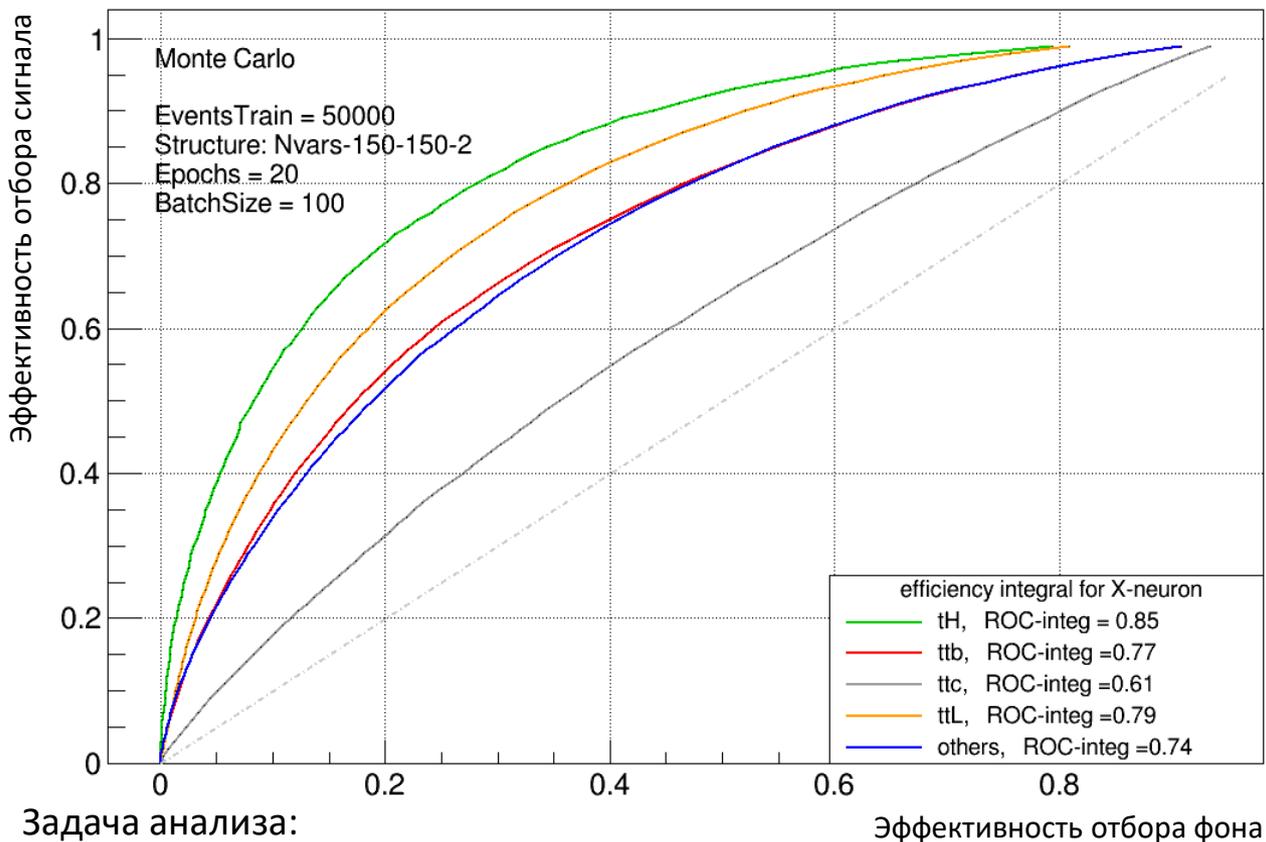
### 300 нейронов для 30 переменных



### 300 нейронов для 24 переменных



# График эффективности для мультиклассификатора. Структура сети n->150/150->2



Задача анализа:

- отделить сигнал от фона,
- отделить фоновые процессы между собой.

\* Сечение ttc, ttb, ttL в смоделированных данных не достаточно точное.

## Discrimination power for tH

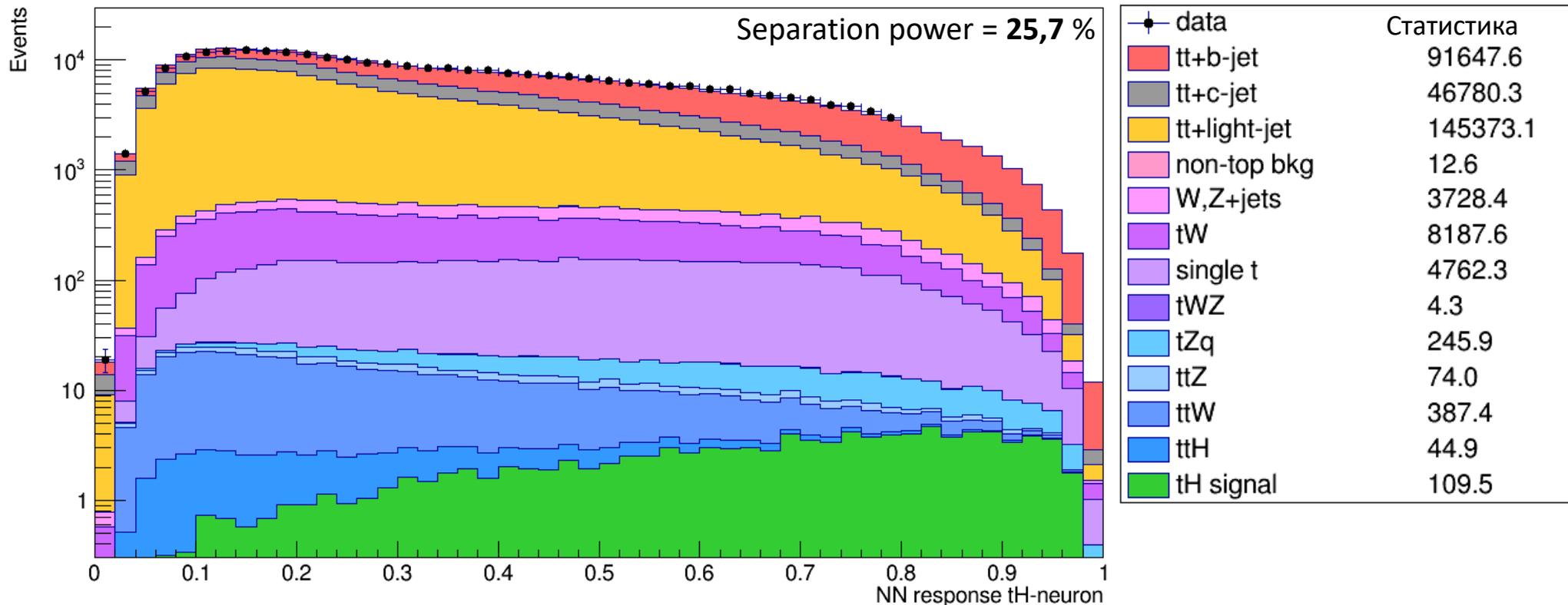
1	: njets_CBT5	: 4.364e-02
2	: nnonbjets	: 4.207e-02
3	: sphericity	: 4.114e-02
4	: aplanarity	: 3.456e-02
5	: Alt\$(nonbjets_eta[1],-4000.0)	: 3.117e-02
6	: raggap_top_fwdjet	: 2.753e-02
7	: Alt\$(fwdjets_pt[0],-10.0)	: 2.486e-02
8	: chi2_min_DeltaEta_tH	: 2.240e-02
9	: tagnonb_eta	: 2.143e-02
10	: tagnonb_topb_m	: 1.951e-02
11	: nfwdjets	: 1.895e-02
12	: chi2_min_tophad_m_ttAll	: 1.756e-02
13	: raggap_maxptjet	: 1.724e-02
14	: inv3jets	: 1.692e-02
15	: nbjets	: 1.338e-02
16	: chi2_min_toplep_pt	: 9.522e-03
17	: Alt\$(nonbjets_pt[0],-10.0)	: 9.229e-03
18	: chi2_min_deltaRq1q2	: 9.041e-03
19	: chi2_min_Whad_m_ttAll	: 8.451e-03
20	: Alt\$(leptons_charge[0],0)	: 8.298e-03
21	: foxWolfram_2_momentum	: 6.888e-03
22	: chi2_min_Invmass_tH	: 4.725e-03
23	: chi2_min_bbnonbjet_m	: 2.361e-03
24	: chi2_min_higgs_m	: 9.671e-04

ROC-integ (TMVA pyKeras)	0.85
Метод BDT	0.86

# Отклик сети на события данных и МС: сигнальный нейрон

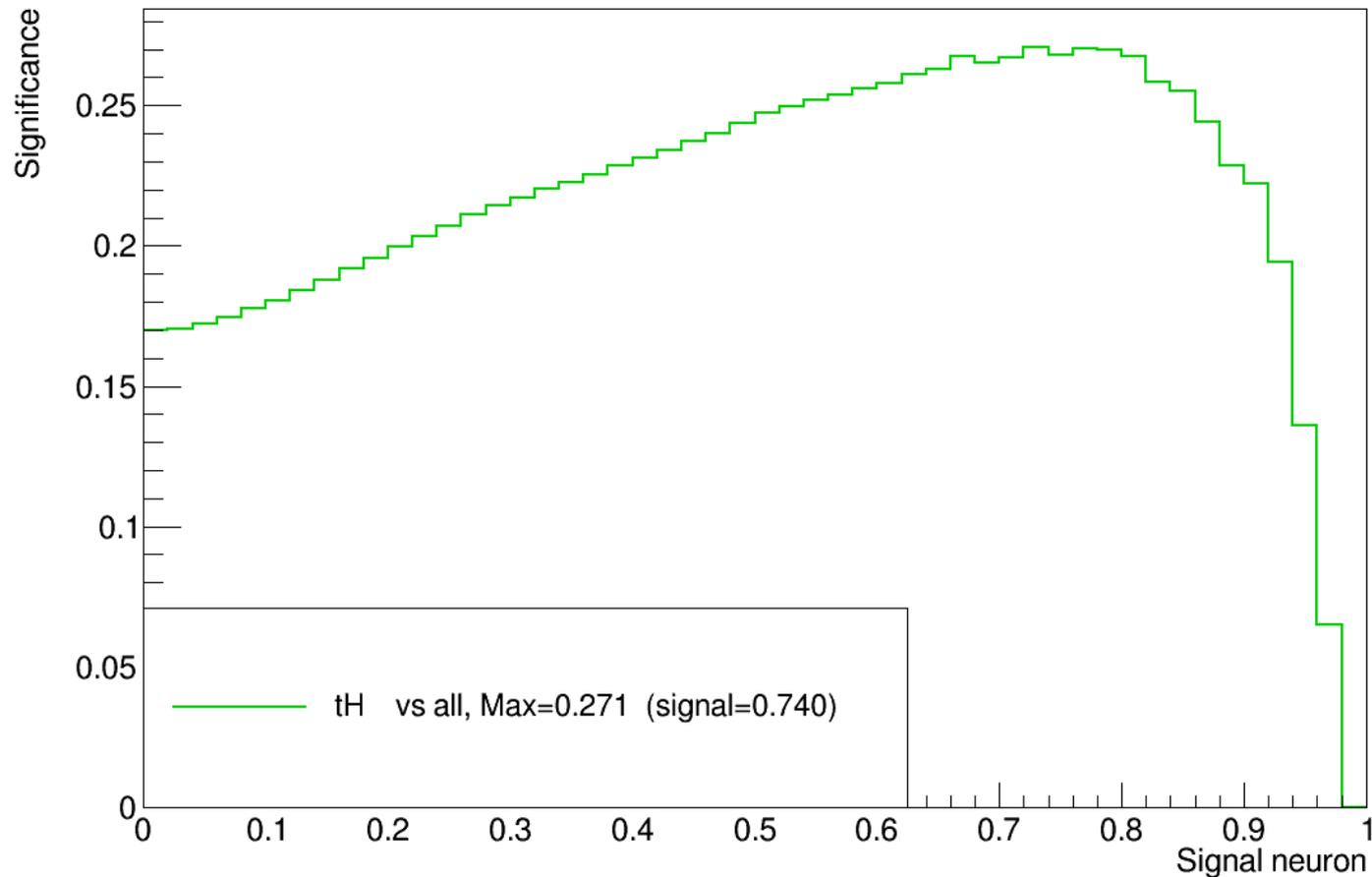
Сеть получает на вход 24 переменные и выдает 1 переменную (отклик сети), в которой аккумулированы различия между сигналом и фоном, содержащиеся во всех 24х входных переменных.

Смоделированные данные МС хорошо описываются экспериментальными. Некоторые процессы не были включены в обучение (например, single t). Экспериментальные данные в области  $> 0.8$  не показаны.



# Значимость для процесса tH

Максимальная значимость сигнала достигается для отбора на отклик tH-нейрона > **0.74**



$$Signif_i = \frac{S_i}{\sqrt{S_i + b_i}}$$

Для pp-столкновений при 13 ТэВ  
интегральной светимости  $139 \text{ fb}^{-1}$ :

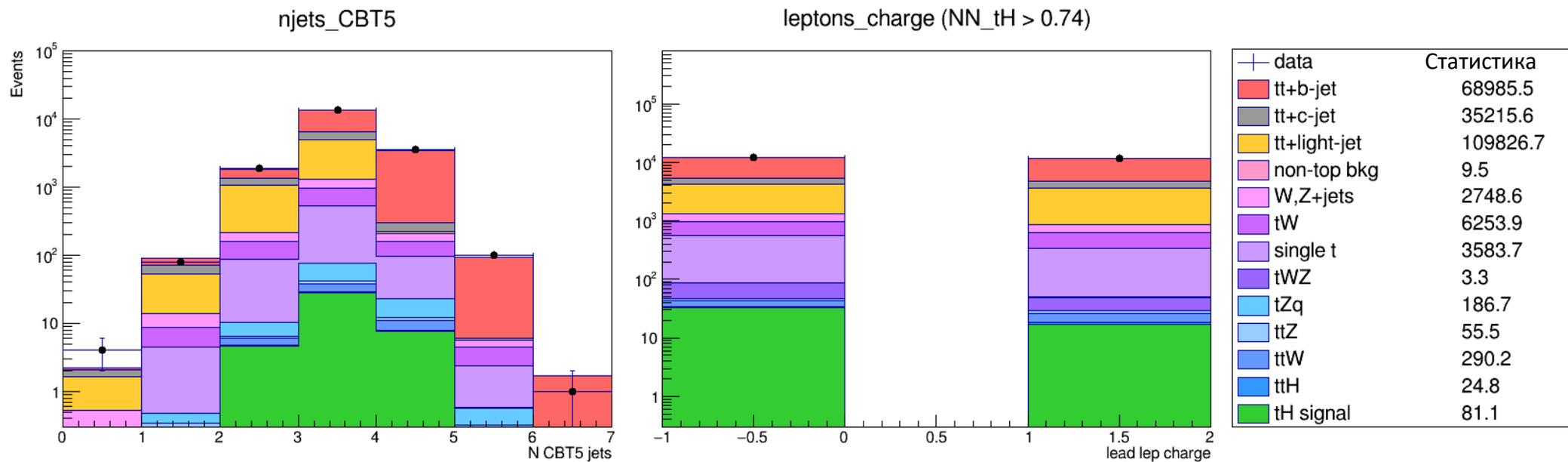
- Без применения сети значимость сигнала **0.17  $\sigma$**
- После применения сети **0.27  $\sigma$**

Таким образом, вне зависимости от количества экспериментальных данных нейронная сеть позволяет увеличить значимость сигнала в **1,6** раз.

То же самое касается чувствительности данных к потенциальным отклонениям от SM

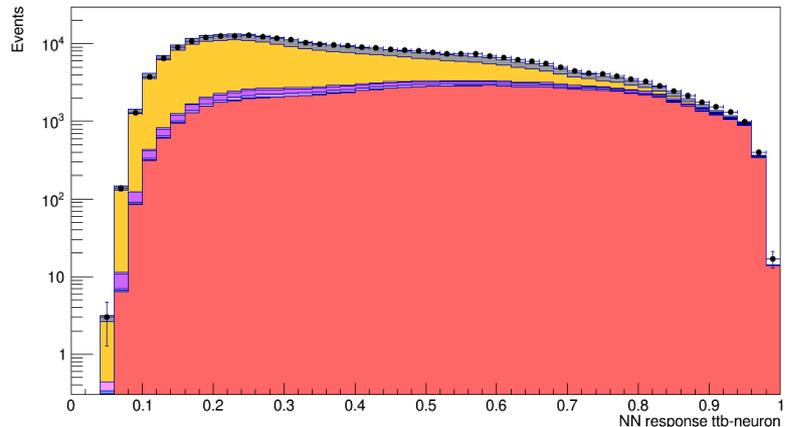
# Кинематика событий в сигнальной области (для NNresponse tH > 0.74)

Применение отбора по отклику сети не нарушает согласие экспериментальных данных с моделированием

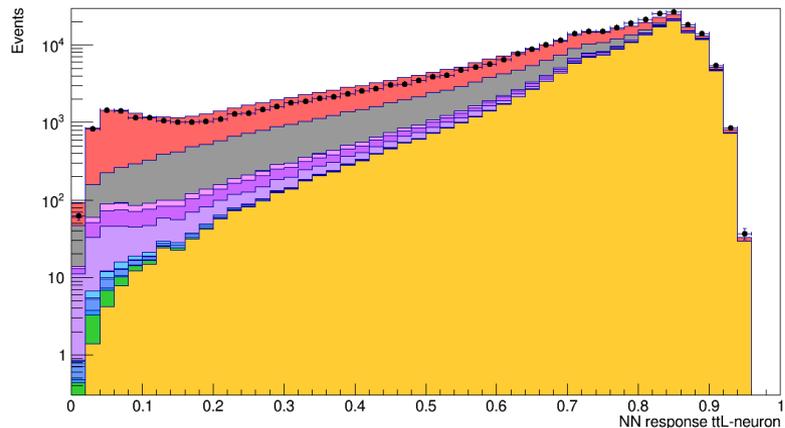


# Отклик сети на события данных и МС: нейроны ttb, ttc, ttL

NNresponse\_ttb



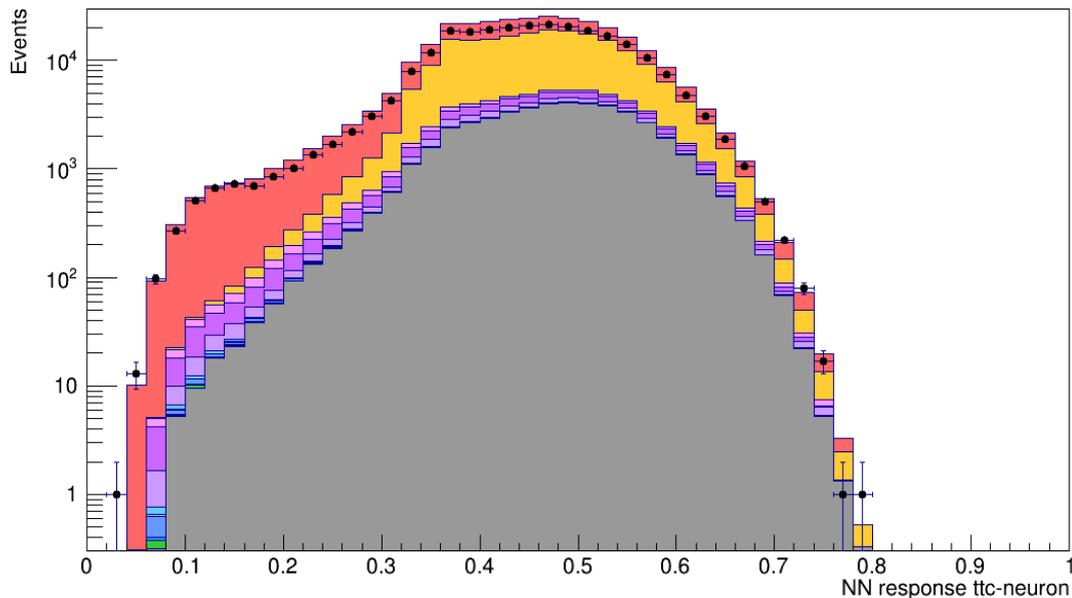
NNresponse\_ttL



Отклонения по откликам нейронов между данными и МС свидетельствуют о неточности моделирования сечений ttb, ttc, ttL.

Благодаря применению сети, поправки к этим сечениям могут быть извлечены из данных путем фитирования.

NNresponse\_ttc



## Выводы и планы

---

- Изучены существующие экспериментальные и теоретические результаты;
- Исследована кинематика сигнального и фоновых процессов. Выбран оптимальный набор кинематических переменных для каждого типа процесса; проведено сравнение МС с данными по всем используемым переменным;
- Изучены приложения для программной реализации DNN (TMVA, Keras);
- Найден рабочий набор настроек нейронной сети. Получены результаты мульти-классификации в задаче поиска событий tH;
- определены сигнальная область и области, где максимизирован вклад каждого из фоновых процессов. Применение обученной сети позволяет увеличить значимость сигнала в 1,6 раз;
- Проведено сравнение с другими приложениями, реализующими глубокое машинное обучение.

### **Основные планы**

- Оптимизация определения сигнальной и фоновых областей на основе откликов нейронных сетей. Использование с этой целью нейронной сети "второго" уровня.
- Завершение оптимизации нейронной сети с точки зрения скорости обучения и эффективности классификации

---

---

Спасибо за внимание!

---

---

# Благодарности

---

---

## Научному руководителю

И.В. Елецких

## Рецензенту

И.Р. Бойко

## Научной группе

Н. Гусейнов

А. Дарибаева

А. Тропина

О. Коваль

## Кафедре ФЭЧ

Зав. Кафедрой В.А. Матвееву

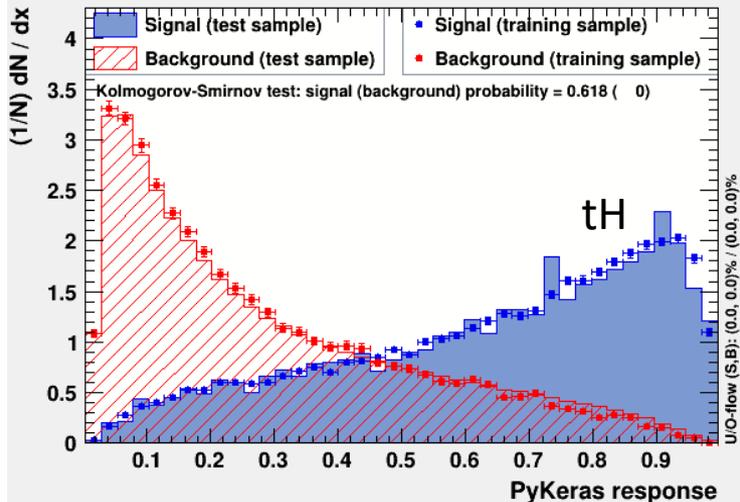
Всему преподавательскому составу

И всем сотрудникам, помогавшим  
организовывать учебный процесс

tH		ttb		ttc		ttl		Others (ttH, ttZ, ttW, tZq)	
1 : njets_CBT5	1 : njets_CBT5	1 : njets_CBT5	1 : njets_CBT5	1 : njets_CBT5	1 : njets				
2 : nnonbjets	2 : njets_CBT4	2 : n_nontophad_jets_CBT4_ttAll	2 : n_nontophad_jets_CBT4_ttAll	2 : n_nontophad_jets_CBT4_ttAll	2 : Alt\$(nonbjets_eta[1],-4000.0)	2 : Alt\$(nonbjets_eta[1],-4000.0)	2 : Alt\$(nonbjets_eta[1],-4000.0)	2 : Alt\$(nonbjets_eta[1],-4000.0)	2 : Alt\$(nonbjets_eta[2],-4000.0)
3 : sphericity	3 : nbjets	3 : nbjets	3 : nbjets	3 : nbjets	3 : Alt\$(nonbjets_eta[2],-4000.0)	3 : Alt\$(nonbjets_eta[2],-4000.0)	3 : Alt\$(nonbjets_eta[2],-4000.0)	3 : Alt\$(nonbjets_eta[2],-4000.0)	3 : Alt\$(nonbjets_pt[0],-10.0)
4 : aplanarity	4 : njets	4 : n_tophad_jets_CBT123_ttAll	4 : n_tophad_jets_CBT123_ttAll	4 : n_tophad_jets_CBT123_ttAll	4 : chi2_min_bbnonbjet_m	4 : chi2_min_bbnonbjet_m	4 : chi2_min_bbnonbjet_m	4 : chi2_min_bbnonbjet_m	4 : Alt\$(nonbjets_eta[1],-4000.0)
5 : Alt\$(nonbjets_eta[1],-4000.0)	5 : Alt\$(nonbjets_eta[1],-4000.0)	5 : top1_m	5 : top1_m	5 : top1_m	5 : Alt\$(nonbjets_pt[0],-10.0)	5 : Alt\$(nonbjets_pt[0],-10.0)	5 : Alt\$(nonbjets_pt[0],-10.0)	5 : chi2_min_bbnonbjet_m	5 : chi2_min_bbnonbjet_m
6 : rappgap_top_fwdjet	6 : chi2_min_bbnonbjet_m	6 : n_tophad_jets_CBT5_ttAll	6 : n_tophad_jets_CBT5_ttAll	6 : n_tophad_jets_CBT5_ttAll	6 : njets_CBT5	6 : njets_CBT5	6 : njets_CBT5	6 : chi2_min_tophad_pt_ttAll	6 : chi2_min_tophad_pt_ttAll
7 : Alt\$(fwdjets_pt[0],-10.0)	7 : chi2_min_Imvmass_tH	7 : top2_m	7 : top2_m	7 : top2_m	7 : inv3jets				
8 : chi2_min_DeltaEta_tH	8 : chi2_min_DeltaEta_tH	8 : bbs_top_m	8 : bbs_top_m	8 : bbs_top_m	8 : chi2_min_Imvmass_tH	8 : chi2_min_Imvmass_tH	8 : chi2_min_Imvmass_tH	8 : chi2_min_toplep_pt	8 : chi2_min_toplep_pt
9 : tagnonb_eta	9 : inv3jets	9 : njets_CBT123	9 : njets_CBT123	9 : njets_CBT123	9 : nbjets	9 : nbjets	9 : nbjets	9 : chi2_min_tophad_m_ttAll	9 : chi2_min_tophad_m_ttAll
10 : tagnonb_topb_m	10 : Alt\$(nonbjets_eta[2],-4000.0)	10 : chi2_min_deltaRq1q2	10 : chi2_min_deltaRq1q2	10 : chi2_min_deltaRq1q2	10 : bbs_top_m	10 : bbs_top_m	10 : bbs_top_m	10 : chi2_min_Whad_m_ttAll	10 : chi2_min_Whad_m_ttAll
11 : nfwjsets	11 : chi2_min_deltaRq1q2	11 : chi2_min_DeltaEta_tH	11 : chi2_min_DeltaEta_tH	11 : chi2_min_DeltaEta_tH	11 : chi2_min_toplep_pt	11 : chi2_min_toplep_pt	11 : chi2_min_toplep_pt	11 : Alt\$(fwdjets_pt[0],-10.0)	11 : Alt\$(fwdjets_pt[0],-10.0)
12 : chi2_min_tophad_m_ttAll	12 : Alt\$(nonbjets_pt[0],-10.0)	12 : Alt\$(leptons_charge[0],0)	12 : Alt\$(leptons_charge[0],0)	12 : Alt\$(leptons_charge[0],0)	12 : chi2_min_tophad_pt_ttAll	12 : chi2_min_tophad_pt_ttAll	12 : chi2_min_tophad_pt_ttAll	12 : chi2_min_deltaRq1q2	12 : chi2_min_deltaRq1q2
13 : rappgap_maxptjet	13 : chi2_min_tophad_m_ttAll	13 : chi2_min_deltaR_Wq1q2	13 : chi2_min_deltaR_Wq1q2	13 : chi2_min_deltaR_Wq1q2	13 : chi2_min_deltaRq1q2	13 : chi2_min_deltaRq1q2	13 : chi2_min_deltaRq1q2	13 : chi2_min_Imvmass_tH	13 : chi2_min_Imvmass_tH
14 : inv3jets	14 : nfwjsets	14 : n_tophad_jets_CBT0_ttAll	14 : n_tophad_jets_CBT0_ttAll	14 : n_tophad_jets_CBT0_ttAll	14 : chi2_min_tophad_m_ttAll	14 : chi2_min_tophad_m_ttAll	14 : chi2_min_tophad_m_ttAll	14 : Alt\$(leptons_charge[0],0)	14 : Alt\$(leptons_charge[0],0)
15 : nbjets	15 : tagnonb_topb_m	15 : chi2_min_Imvmass_tH	15 : chi2_min_Imvmass_tH	15 : chi2_min_Imvmass_tH	15 : Alt\$(fwdjets_pt[0],-10.0)	15 : Alt\$(fwdjets_pt[0],-10.0)	15 : Alt\$(fwdjets_pt[0],-10.0)	15 : tagnonb_topb_m	15 : tagnonb_topb_m
16 : chi2_min_toplep_pt	16 : rappgap_maxptjet	16 : chi2_min_toplep_pt	16 : chi2_min_toplep_pt	16 : chi2_min_toplep_pt	16 : tagnonb_topb_m	16 : tagnonb_topb_m	16 : tagnonb_topb_m	16 : nfwjsets	16 : nfwjsets
17 : Alt\$(nonbjets_pt[0],-10.0)	17 : aplanarity	17 : Alt\$(nonbjets_eta[2],-4000.0)	17 : Alt\$(nonbjets_eta[2],-4000.0)	17 : Alt\$(nonbjets_eta[2],-4000.0)	17 : nfwjsets	17 : nfwjsets	17 : nfwjsets	17 : chi2_min_DeltaEta_tH	17 : chi2_min_DeltaEta_tH
18 : chi2_min_deltaRq1q2	18 : Alt\$(fwdjets_pt[0],-10.0)	18 : chi2_min_bbnonbjet_m	18 : chi2_min_bbnonbjet_m	18 : chi2_min_bbnonbjet_m	18 : chi2_min_Whad_m_ttAll	18 : chi2_min_Whad_m_ttAll	18 : chi2_min_Whad_m_ttAll	18 : nbjets	18 : nbjets
19 : chi2_min_Whad_m_ttAll	19 : sphericity	19 : chi2_min_higgs_m	19 : chi2_min_higgs_m	19 : chi2_min_higgs_m	19 : tagnonb_eta				
20 : Alt\$(leptons_charge[0],0)	20 : rappgap_top_fwdjet	20 : chi2_min_tophad_pt_ttAll	20 : chi2_min_tophad_pt_ttAll	20 : chi2_min_tophad_pt_ttAll	20 : rappgap_maxptjet	20 : rappgap_maxptjet	20 : rappgap_maxptjet	20 : aplanarity	20 : aplanarity
21 : foxWolfram_2_momentum	21 : chi2_min_toplep_pt	21 : foxWolfram_2_momentum	21 : foxWolfram_2_momentum	21 : foxWolfram_2_momentum	21 : rappgap_maxptjet				
22 : chi2_min_Imvmass_tH	22 : chi2_min_tophad_pt_ttAll	22 : nfwjsets	22 : nfwjsets	22 : nfwjsets	22 : chi2_min_DeltaEta_tH	22 : chi2_min_DeltaEta_tH	22 : chi2_min_DeltaEta_tH	22 : rappgap_top_fwdjet	22 : rappgap_top_fwdjet
23 : chi2_min_bbnonbjet_m	23 : tagnonb_eta	23 : inv3jets	23 : inv3jets	23 : inv3jets	23 : rappgap_top_fwdjet	23 : rappgap_top_fwdjet	23 : rappgap_top_fwdjet	23 : njets_CBT5	23 : njets_CBT5
24 : chi2_min_higgs_m	24 : Alt\$(leptons_charge[0],0)	24 : rappgap_top_fwdjet	24 : rappgap_top_fwdjet	24 : rappgap_top_fwdjet	24 : sphericity	24 : sphericity	24 : sphericity	24 : foxWolfram_2_momentum	24 : foxWolfram_2_momentum
	25 : chi2_min_Whad_m_ttAll	25 : chi2_min_tophad_m_ttAll	25 : chi2_min_tophad_m_ttAll	25 : chi2_min_tophad_m_ttAll	25 : Alt\$(leptons_charge[0],0)	25 : Alt\$(leptons_charge[0],0)	25 : Alt\$(leptons_charge[0],0)	25 : sphericity	25 : sphericity
	26 : chi2_min_tophad_eta_ttAll	26 : Alt\$(fwdjets_eta[0],-4000.0)	26 : Alt\$(fwdjets_eta[0],-4000.0)	26 : Alt\$(fwdjets_eta[0],-4000.0)	27 : rappgap_maxptjet	27 : rappgap_maxptjet	27 : rappgap_maxptjet		
		27 : rappgap_maxptjet	27 : rappgap_maxptjet	27 : rappgap_maxptjet	28 : W1_m	28 : W1_m	28 : W1_m		
		28 : W1_m	28 : W1_m	28 : W1_m	29 : tagnonb_topb_m	29 : tagnonb_topb_m	29 : tagnonb_topb_m		
		29 : tagnonb_topb_m	29 : tagnonb_topb_m	29 : tagnonb_topb_m	30 : sphericity_t	30 : sphericity_t	30 : sphericity_t		
		30 : sphericity_t	30 : sphericity_t	30 : sphericity_t					
ROC-integ		ROC-integ		ROC-integ		ROC-integ		ROC-integ	
TMVA pyKeras	0.85	TMVA pyKeras	0.77	TMVA pyKeras	0.61	TMVA pyKeras	0.79	TMVA pyKeras	0.74
Old BDT	0.86	Old BDT	0.75	Old BDT	0.60	Old BDT	0.78	Old BDT	0.64

# Тренировка и тестирование результатов

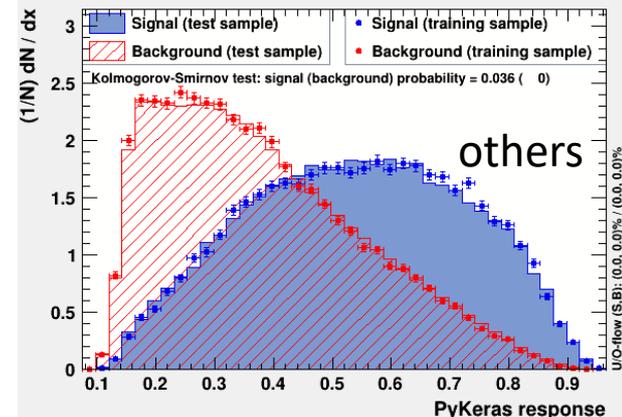
TMVA overtraining check for classifier: PyKeras



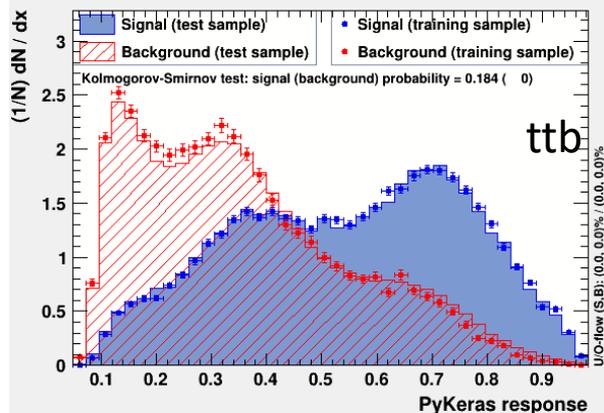
Для тестирования используется независимый набор событий, не использовавшийся для обучения.

Ожидается одинаковый отклик сети на события тренировочные и события тестовые.

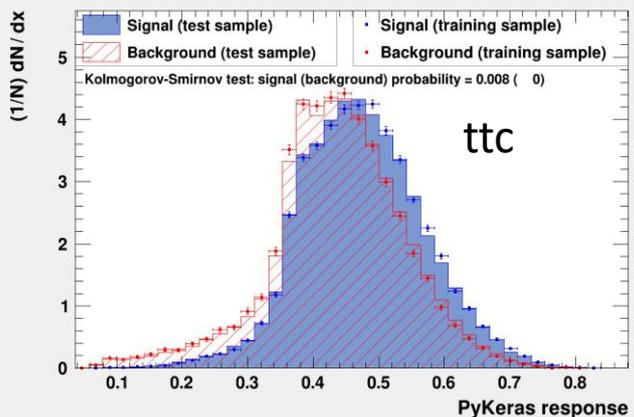
TMVA overtraining check for classifier: PyKeras



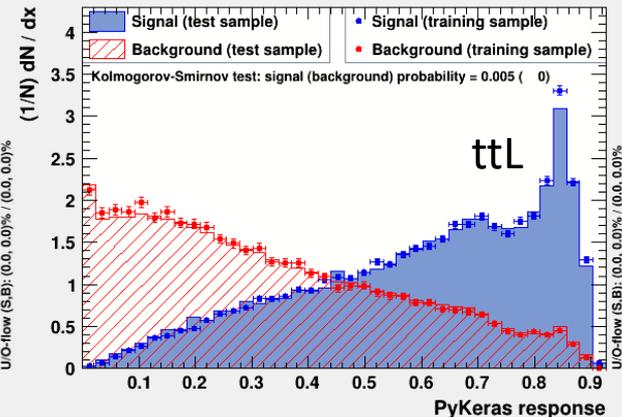
TMVA overtraining check for classifier: PyKeras



TMVA overtraining check for classifier: PyKeras

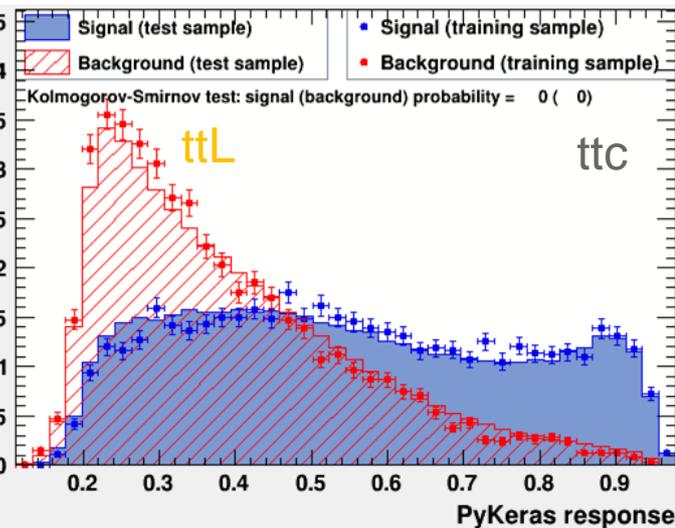
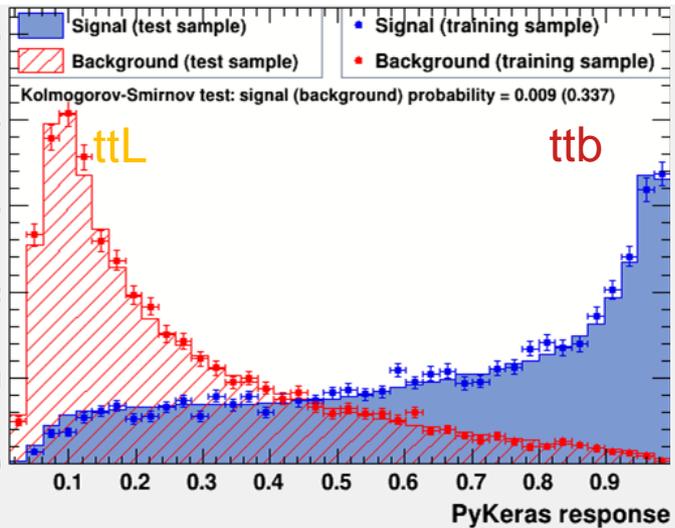
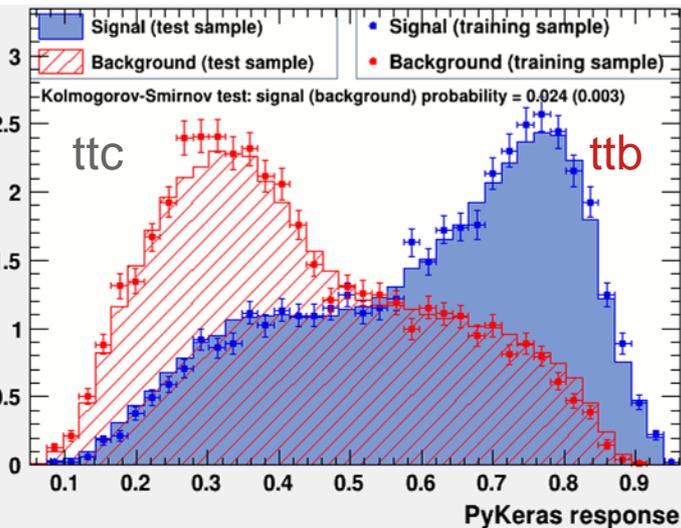
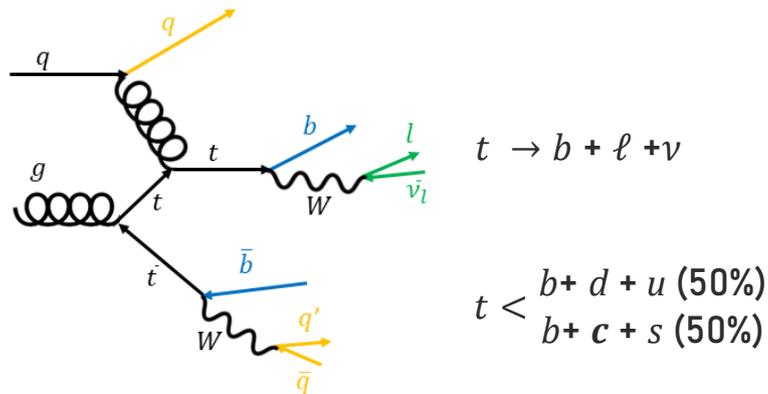


TMVA overtraining check for classifier: PyKeras



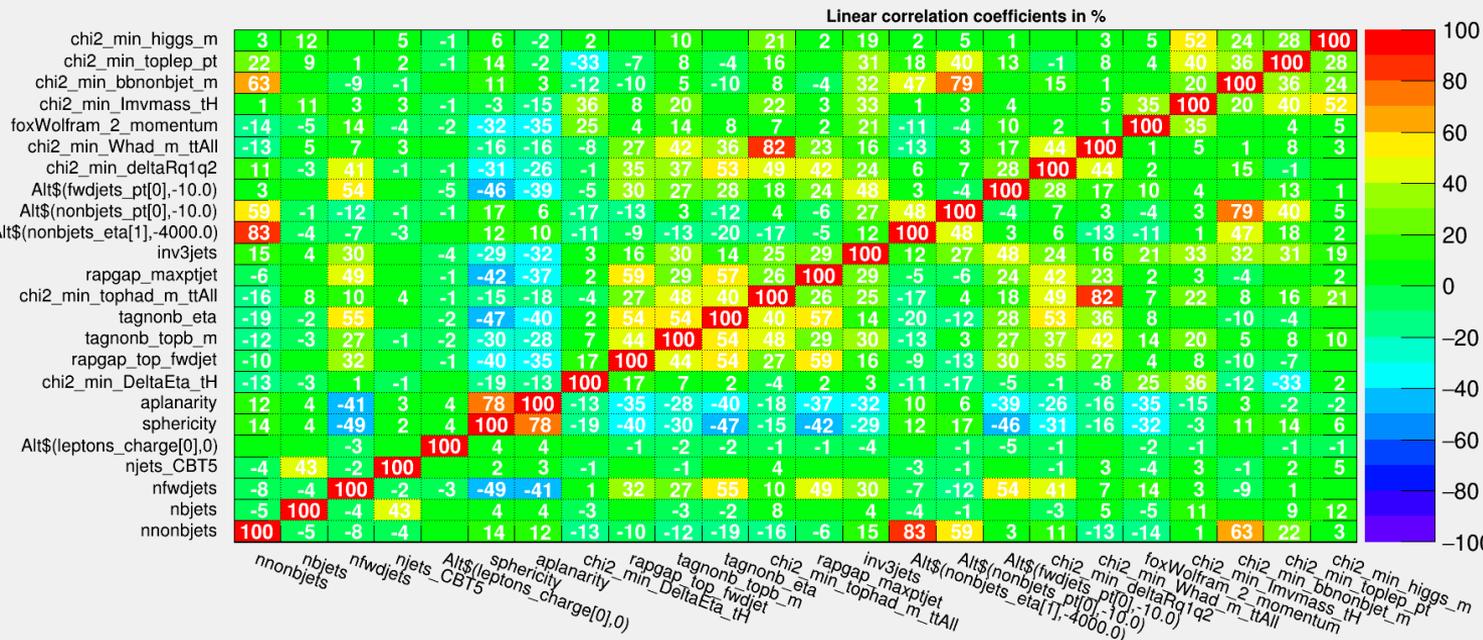
# Разделение $t\bar{t}c$ , $t\bar{t}b$ и $t\bar{t}+\text{light-flavour}$ струй.

“Плохая” эффективность для  $t\bar{t}c$  связана с его кинематическими характеристиками, а не с обучением сети



# Задача разделения сигнала и фонов. Выбор набора переменных. Матрица корреляции

## Correlation Matrix (signal)



## Discrimination power for tH

1	: njets_CBT5	: 4.364e-02
2	: nnonbjets	: 4.207e-02
3	: sphericity	: 4.114e-02
4	: aplanarity	: 3.456e-02
5	: Alt\$(nonbjets_eta[1],-4000.0)	: 3.117e-02
6	: rapgap_top_fwdjet	: 2.753e-02
7	: Alt\$(fwdjets_pt[0],-10.0)	: 2.486e-02
8	: chi2_min_DeltaEta_th	: 2.240e-02
9	: tagnonb_eta	: 2.143e-02
10	: tagnonb_topb_m	: 1.951e-02
11	: nfdwjets	: 1.895e-02
12	: chi2_min_tophad_m_ttAll	: 1.756e-02
13	: rapgap_maxptjet	: 1.724e-02
14	: inv3jets	: 1.692e-02
15	: nbjets	: 1.338e-02
16	: chi2_min_toplep_pt	: 9.522e-03
17	: Alt\$(nonbjets_pt[0],-10.0)	: 9.229e-03
18	: chi2_min_deltaRq1q2	: 9.041e-03
19	: chi2_min_Whad_m_ttAll	: 8.451e-03
20	: Alt\$(leptons_charge[0],0)	: 8.298e-03
21	: foxWolfram_2_momentum	: 6.888e-03
22	: chi2_min_Imvmassth	: 4.725e-03
23	: chi2_min_bnonbjet_m	: 2.361e-03
24	: chi2_min_higgs_m	: 9.671e-04

**njets\_CBT X** – количество струй, попадающих в CBT бин X ( X=5 или 4 – жесткая b-струя; X=1, 2 или 3– легкая струя).

**n non-b jets** – число центральных струй, отличных от b струй.

**nbjets** – количество b струй = njets\_CBT5 + njets\_CBT4

**Inv 3 jets** – инвариантная масса трех струй с наибольшим pt.

**chi2 min X** – величина, получаемая отбором объектов, реконструируемых методом Хи-квадрат.

**n fwd jets** – число передних струй.

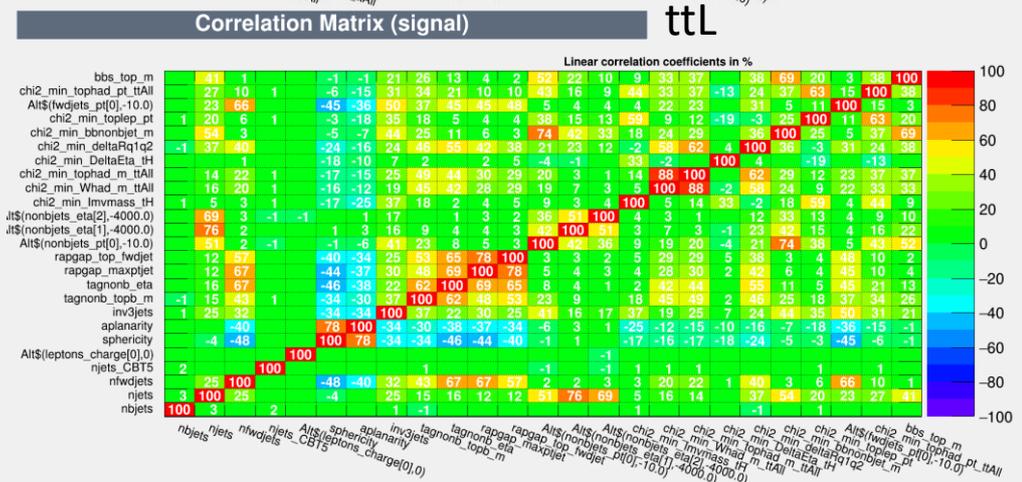
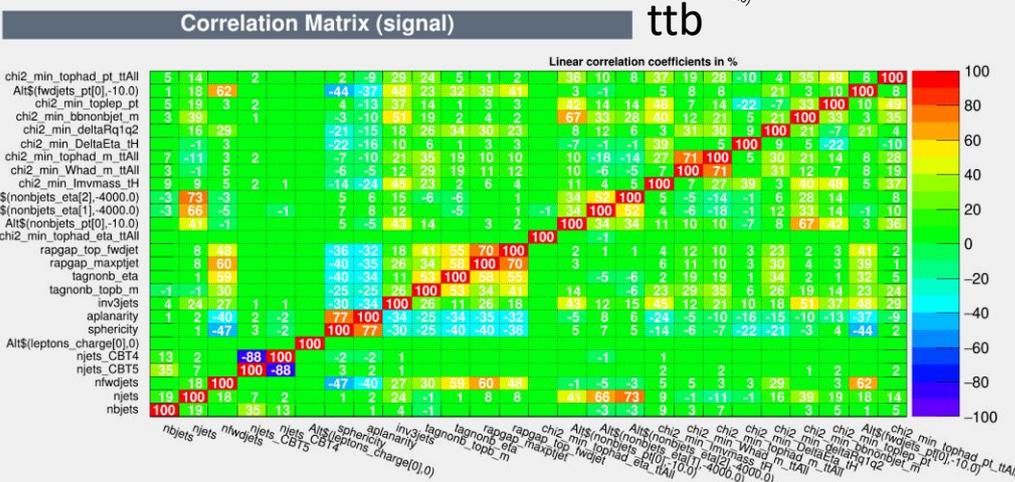
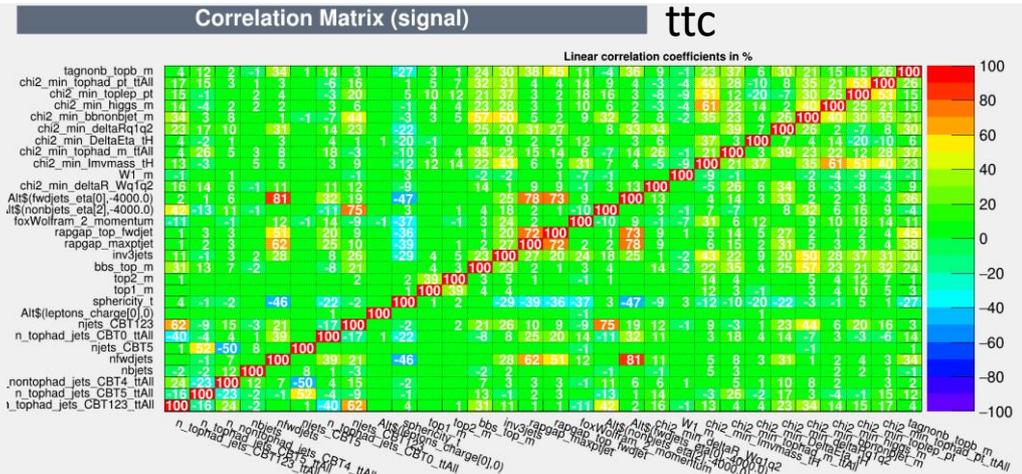
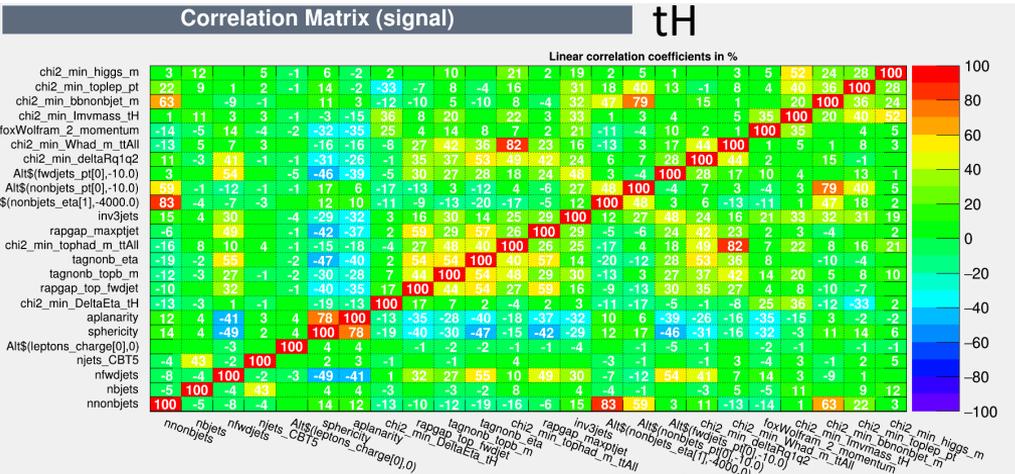
**leptons charge** – заряд лидирующего лептона

**fox-wolfram moments** – характеризует геометрические корреляции между струями.

**rapgap\_maxptjet** – разница псевдобыстрот лидирующих струй

В анализе используется до 30 переменных, наиболее чувствительных к основным процессам.  
Корреляция < 88 %

# Корреляция переменных < 88%



---

---

# Separation power

---

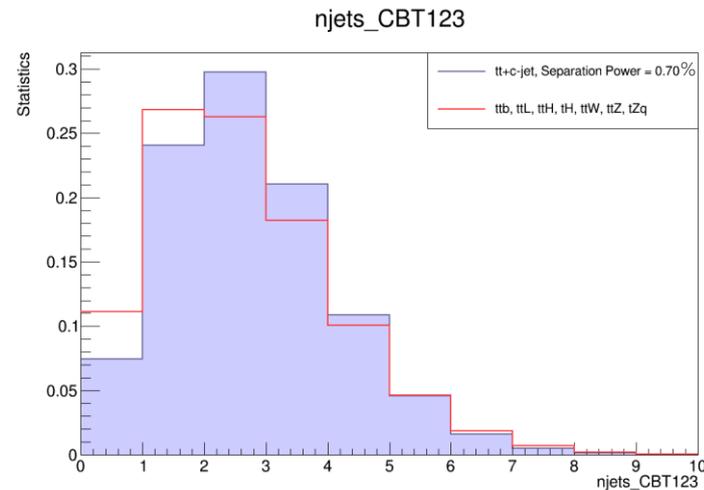
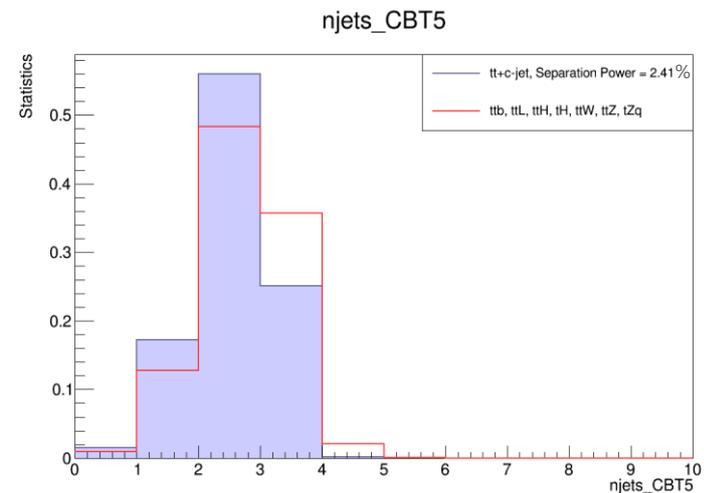
---

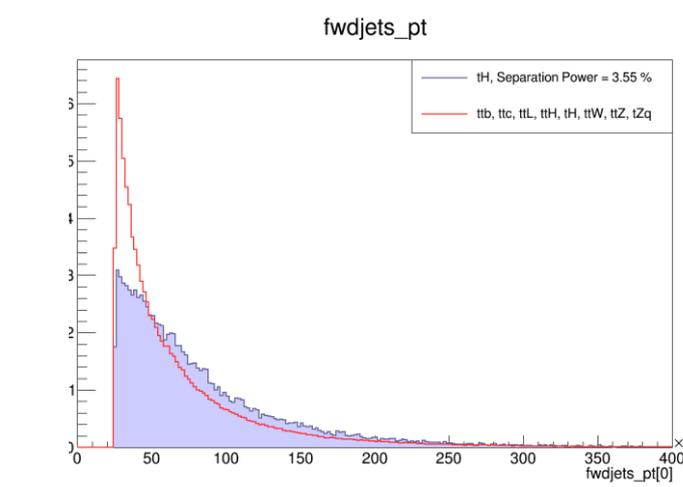
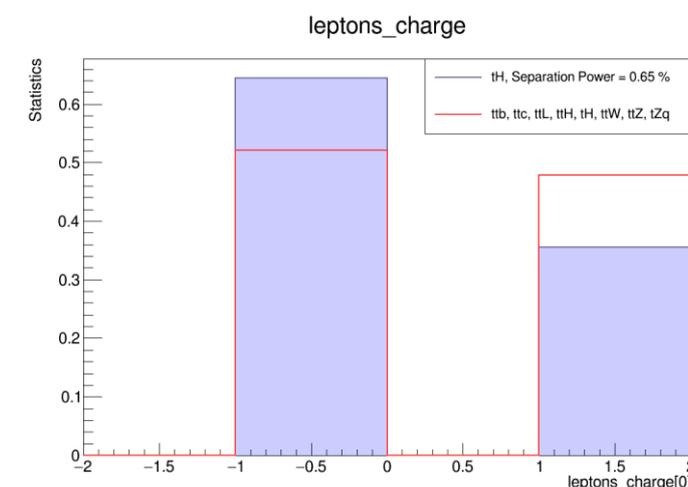
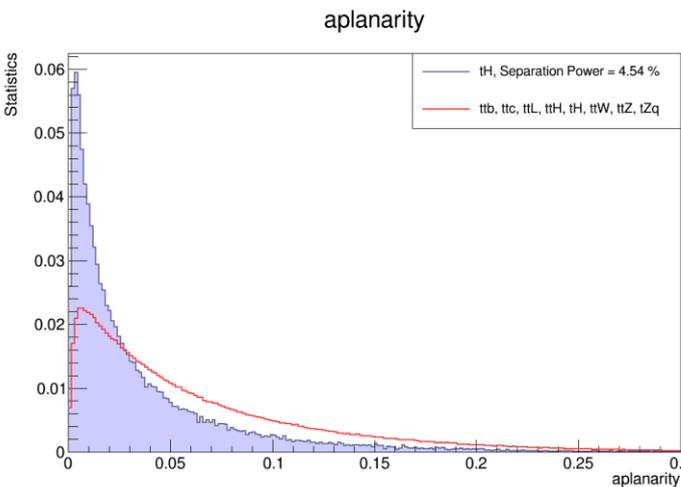
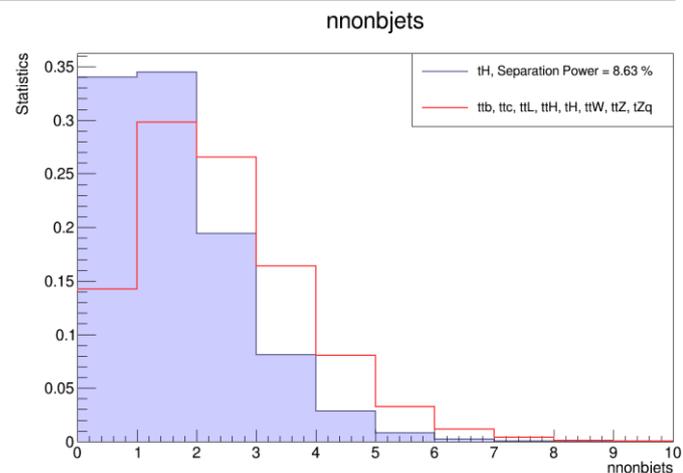
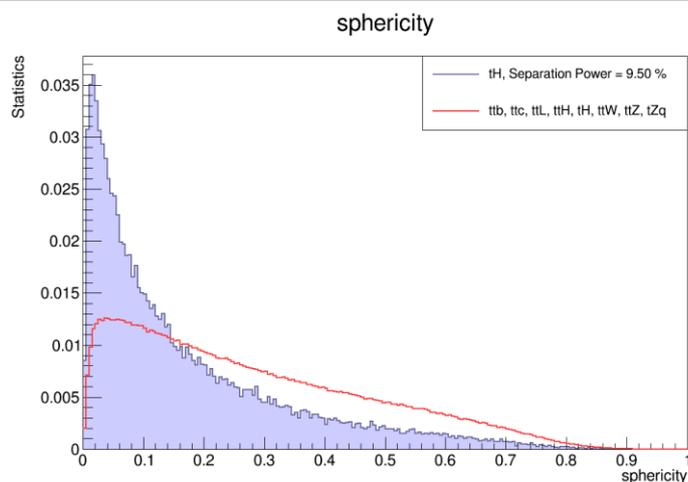
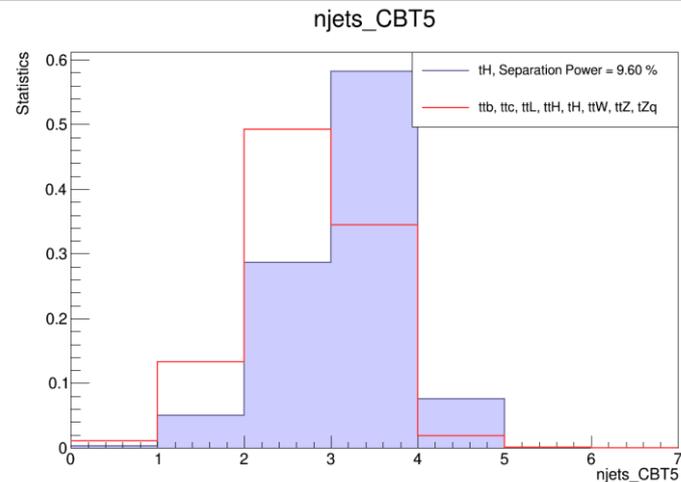
$$S = \frac{1}{2} \left( \sum_{i=0}^{nbins} \frac{(s_i - b_i)^2}{s_i + b_i} \right) \times 100$$

Variables [0]	Power %
leptons_pt	0.07
leptons_eta	0.02
leptons_e	0.07
leptons_phi	0.02
jets_pt	0.09
jets_eta	0.02
jets_e	0.01
jets_phi	0.05
bjets_pt	0.09
bjets_eta	0.01
bjets_e	0.06
bjets_phi	0.01
<b>nonbjets_pt</b>	<b>0.14</b>
nonbjets_eta	0.01
nonbjets_e	0.10
nonbjets_phi	0.02
<b>fwdjets_pt</b>	<b>0.14</b>
fwdjets_eta	0.07
fwdjets_e	0.10
fwdjets_phi	0.04

Variables [0]	Power %
njets_CBT5	2.41
njets_CBT4	1.65
njets_CBT123	0.70
nfwjets	0.01

Variables [0]	Power %
W1_m	0.03
W1_pt	0.10
W1_eta	0.02
W2_m	0.03
W2_pt	0.10
W2_eta	0.03
top1_m	0.44
top1_pt	0.08
top1_eta	0.03
top2_m	0.26
top2_pt	0.07
top2_eta	0.03





---

---

# Машинное обучение

---

---

## Искусственный интеллект:

Метод, позволяющий компьютерам имитировать человеческий интеллект.

## Машинное обучение:

Способ достижения Искусственного интеллекта.

## Нейронная сеть:

Одна из моделей машинного обучения.

## Глубокое машинное обучение:

Подмножество МО, основанное на искусственных нейронных сетях.



---

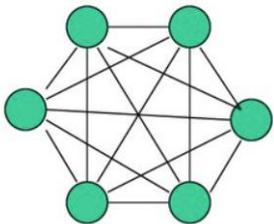
---

# Нейронные сети

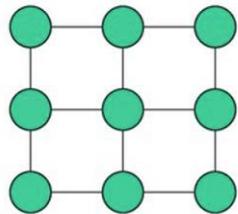
---

---

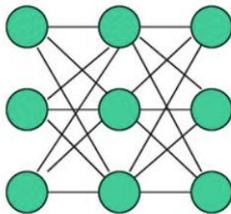
Классификация по топологии:



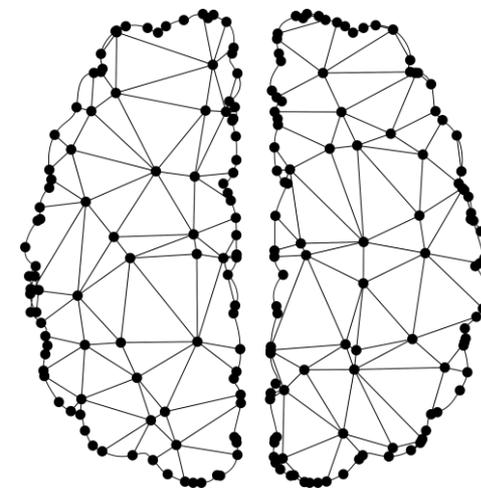
Полносвязные



Слабосвязные



Многослойные



1. **Искусственная нейронная сеть** – функция с большим числом варьируемых параметров (веса и смещения)
2. **Нейрон** – узел искусственной нейронной сети, являющийся упрощённой моделью естественного нейрона.  
Математически, искусственный нейрон обычно представляют как некоторую нелинейную функцию от единственного аргумента — линейной комбинации всех входных сигналов. Данную функцию называют функцией активации или функцией срабатывания, передаточной функцией.
3. **Слой** – группа нейронов, находящихся на одном уровне иерархии. В плоскостных сетях это группа нейронов, имеющих общий вход / выход, функцию активации и не соединены между собой.
4. **Скрытый слой** - слой нейронной сети, находящийся между её входным и выходным слоями.
5. **Веса** – это действительные числа, отражающие коэффициент связи между конкретными нейронами.
6. **Функция активации** нейрона – функция, определяющая выходной сигнал на основе трансформации входного сигнала:  $\text{выход} = \text{функция активации}(\text{вход})$ .
7. **Loss / функция ошибки / функция потерь** (Функция минимизации) – математическая дифференцируемая функция, характеризующая разницу между «истинным» значением целевой переменной и предсказанным нейронной сетью значением. В Keras – название параметра, передаваемого в метод `compile` при компиляции сети.
8. **optimizer (функция оптимизатора)** – оптимизационный алгоритм (функция), задача которого - изменение весов сети для уменьшения ошибки сети в процессе обучения.
9. **Epoch** – Число обучающих итераций - количество повторений циклов обучения для всей выборки данных. Размер обучаемой за раз партии параметров
10. **Батч / batch\_size** – количество элементов выборки, с которыми идет работа в пределах одной итерации до изменения весов.
11. **Итерация обучения** – это часть процесса работы сети с данными, продолжительность которой равна времени обработки одного батча.

**Epoch** – Число обучающих итераций - количество повторений циклов обучения для всей выборки данных. Размер обучаемой за раз партии параметров

**Батч / batch\_size** – количество элементов выборки, с которыми идет работа в пределах одной итерации до изменения весов.

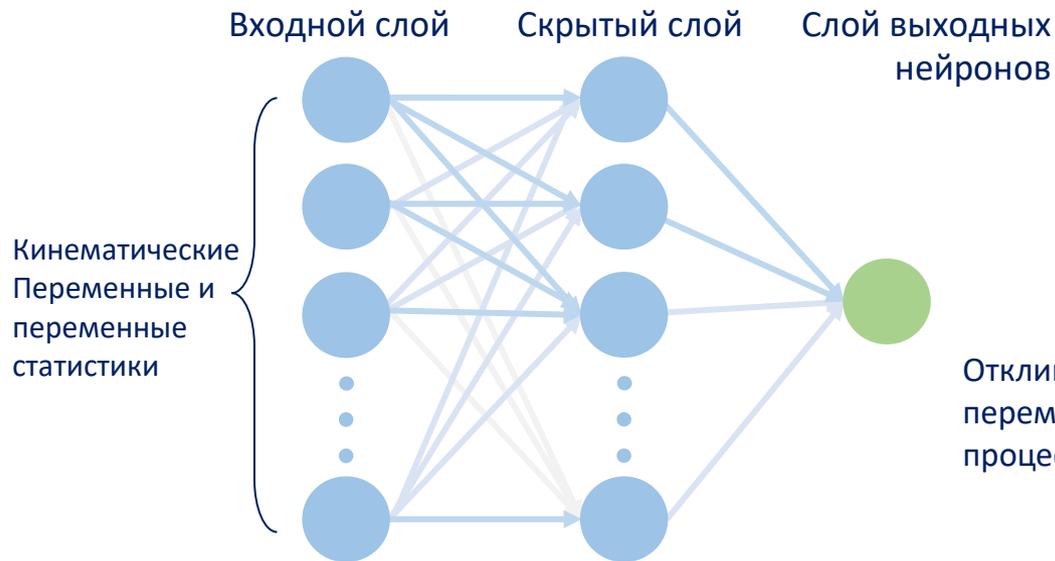
**Итерация обучения** – это часть процесса работы сети с данными, продолжительность которой равна времени обработки одного батча.

```
Train on 100000 samples, validate on 1859721 samples
Epoch 1/20
 99900/100000 [=====>.] - ETA: 0s - loss: 0.4974 - acc: 0.7657Epoch 00000: val_loss improved from inf to 0.07134, saving model to TMVAb_ttc_50k_NN_150n150n_100n50n/weights/TrainedModel_PyKeras.h5
100000/100000 [=====>.] - 33s - loss: 0.4973 - acc: 0.7657 - val_loss: 0.0713 - val_acc: 0.7852
Epoch 2/20
 99700/100000 [=====>.] - ETA: 0s - loss: 0.4669 - acc: 0.7829Epoch 00001: val_loss improved from 0.07134 to 0.06530, saving model to TMVAb_ttc_50k_NN_150n150n_100n50n/weights/TrainedModel_PyKeras.h5
100000/100000 [=====>.] - 33s - loss: 0.4669 - acc: 0.7829 - val_loss: 0.0653 - val_acc: 0.8022
Epoch 3/20
 99700/100000 [=====>.] - ETA: 0s - loss: 0.4577 - acc: 0.7876Epoch 00002: val_loss did not improve
100000/100000 [=====>.] - 31s - loss: 0.4577 - acc: 0.7877 - val_loss: 0.0679 - val_acc: 0.7926
```

**Нейронная сеть** – функция со многими параметрами – весами нейронов. Оптимальные значения этих параметров обеспечивают наилучшее разделение сигнала и фона.

Есть различные алгоритмы поиска значений параметров – алгоритмы обучения.

Пример простой нейронной сети:



## Внешние параметры сети

Количество слоев	Число обучающих итераций
Количество нейронов в слое	Размер обучаемой за раз партии параметров
Входные переменные	Функция активации и минимизируемая функция потерь

## Алгоритмы обучения сети

Абсолютно случайный поиск	Градиентный спуск (производная минимизируемой функции)
---------------------------	--

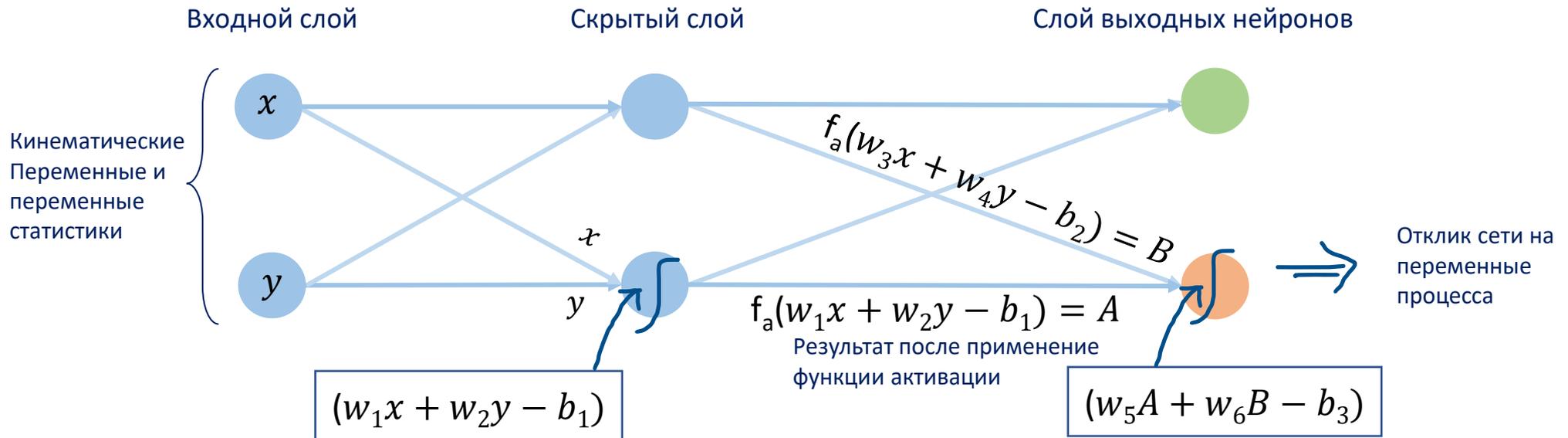
Deep Neural Networks (DNN) – нейронные сети с более чем одним скрытым слоем

Нейронная сеть – «**черный ящик**».

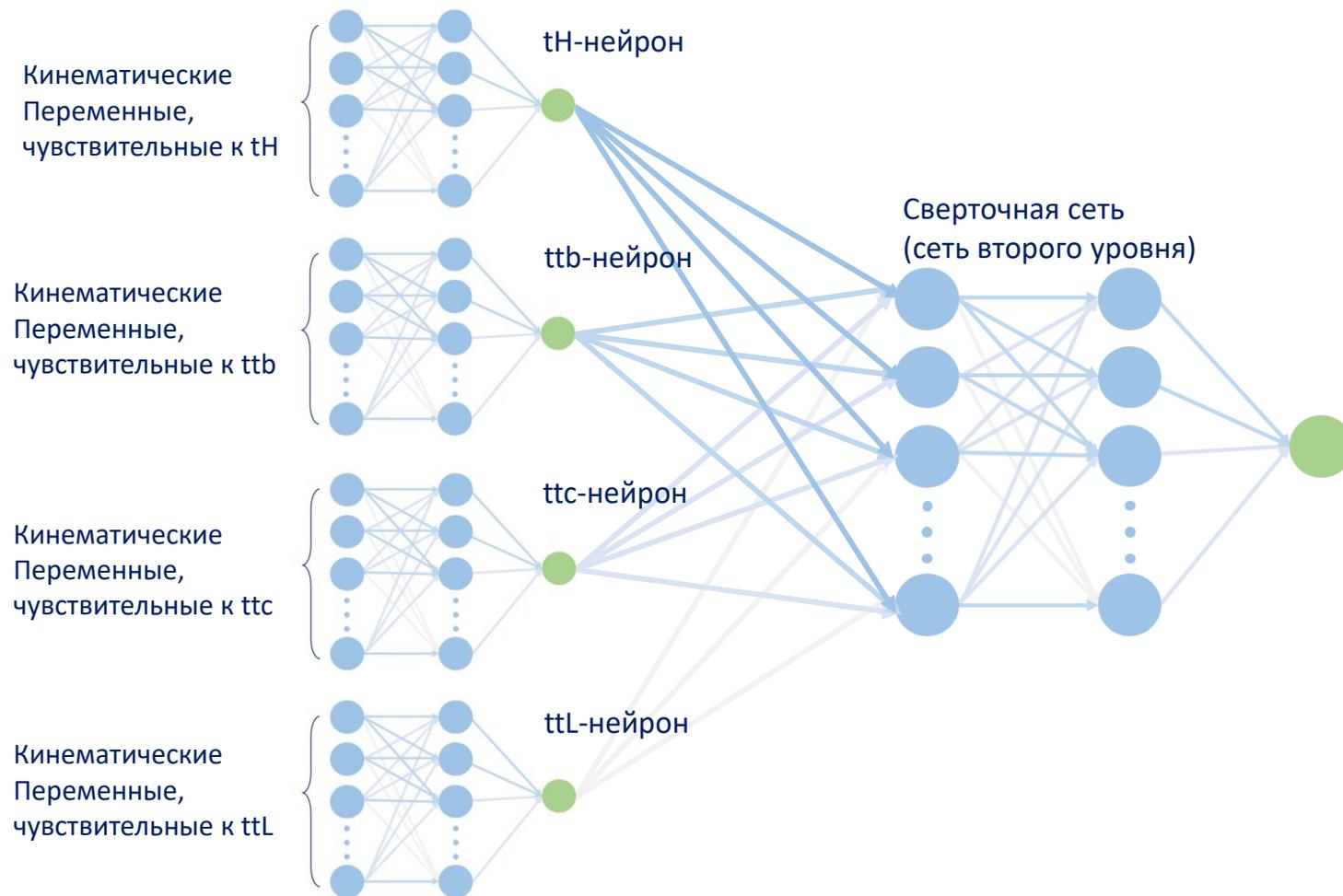
Знания накапливаются в виде весовых коэффициентов и не поддаются содержательной интерпретации

Пример простой нейронной сети:

**Варьируемые (свободные) параметры:** веса ( $w_1 - w_8$ ) и смещения ( $b_1 - b_4$ )

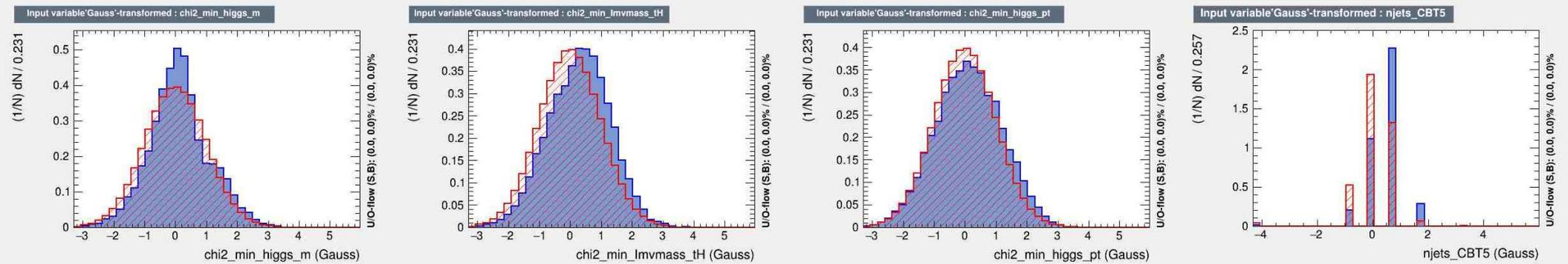


# Применение нейронной сети для классификации процессов. Структура сети



# Трансформация

До тренировки кинематические переменные необходимо обезразмерить и привести к общей области значений.



В TMVA ROOT реализованы:

- 1) нормализация (англ. Normalisation),
- 2) преобразование переменных в Равномерные распределения (англ. Uniformisation),
- 3) преобразование переменных в распределения Гаусса (англ. Gaussianisation),
- 4) декорреляция (англ. Decorrelation)

Распределения после трансформации нескольких кинематических переменных. Синим показаны распределения для сигнальных событий, красным - распределения фоновых.

