

# Актуальные примеры использования машинного обучения в ФЭЧ

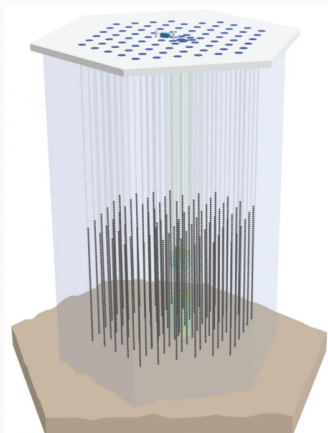
1. Наблюдение  $\tau$ -нейтрино с детектором IceCube с помощью CNN сети VGG16
  2. Новый тип архитектуры нейронной сети KAN
  3. Первое применение KAN в НЕР
- 

Семенов А.В.

Июль 21, 2025

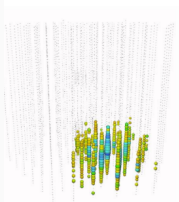
# IceCube: Краткое описание

- [Ссылка](#) на статью: Observation of Seven Astrophysical Tau Neutrino Candidates with IceCube
- Детектор на южном полюсе километровых размеров.
- 5,160 оптических модулей (DOMs) на 86 струнах.
- Позволяет регистрировать  $\nu_\tau$ .



## “Casacdes”

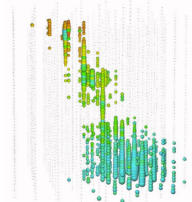
induced by  $\nu_e$  or NC interactions



Common events

## “Tracks”

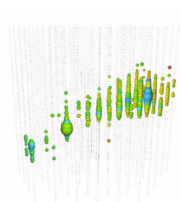
induced by Atm.  $\mu$  (or  $\nu_\mu$ )



Ubiquitous events

## “Starting Tracks”

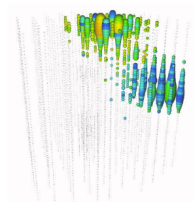
induced by  $\nu_\mu$



Common events

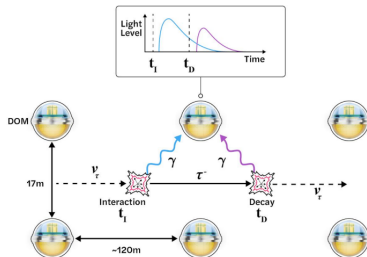
## “Double Cascades”

induced by  $\nu_\tau$

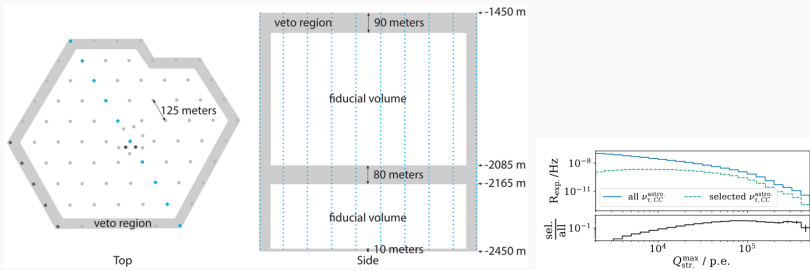


Very rare events

1. Одиночные каскады
2. Вертикальные треки
3. Горизонтальные треки
4. Двойные каскады



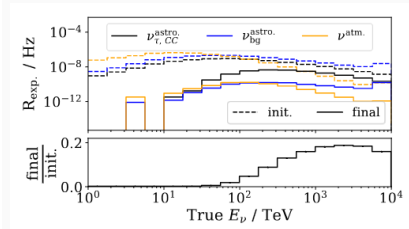
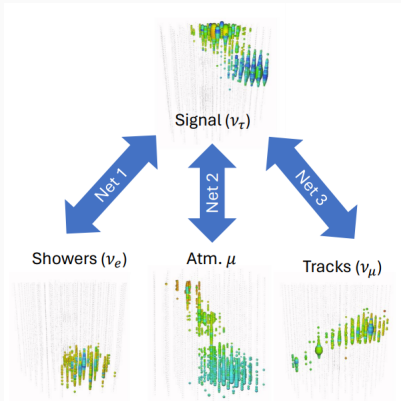
- Применяется вето техника для подавления атмосферных мюонов и нейтрино.
- Дополнительные критерии на морфологию ливней.
- Отбираются события с двойным каскадом.



Области вето.

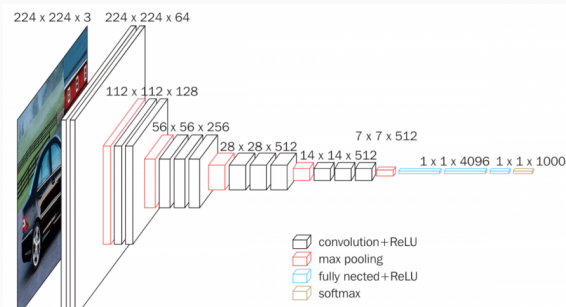
# События $\nu_\tau$

- События с  $\nu_\tau$  создают двойные каскады из-за распада  $\tau$ .
- Используется зависимость энергия-время для анализа сигналов в DOMs.
- Для классификации используется свёрточная нейронная сеть VGG16.



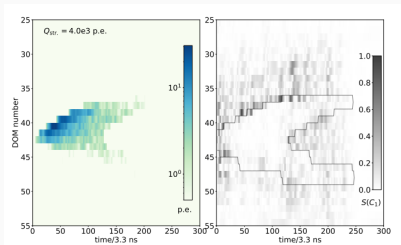
# VGG16: Архитектура

- Разработана Visual Geometry Group (Oxford) для классификации изображений. ImageNet 2014: Достигла 7.3% ошибочности (топ 5) на ILSVRC датасете.
- 16 слоёв: 13 свёрточных (фильтры 3x3) + 3 слоя обычной сети.
- Пять свёрточных блоков по 2-3 слоя сжимаются 2x2 max-pooling'ами (берётся максимум).
- Сделана для классификации 1000 типов изображений (softmax — вероятности).



# VGG16: IceCube

- Перенос обучения: Предтренированная VGG16 уже может выделять признаки.
- Три отдельные сети для классификации  $\nu_\tau$  двойных каскадов от фоновых событий.
- Обучение на моделирование.

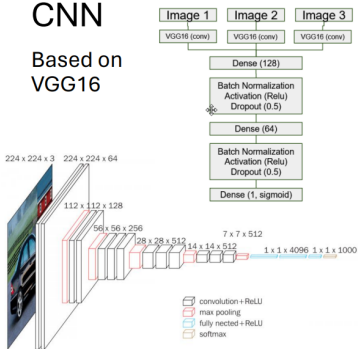


SIM одной струны.

Для классификации события берётся три соседних струны с наибольшей амплитудой

## CNN

Based on VGG16

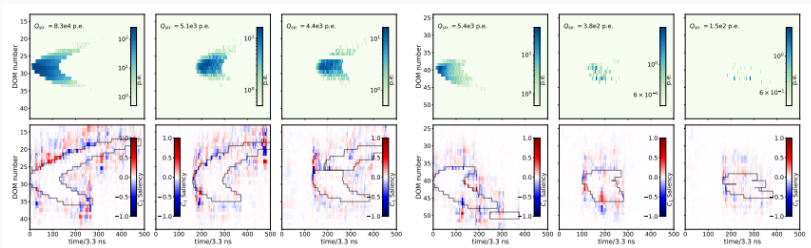


# Результаты отбора

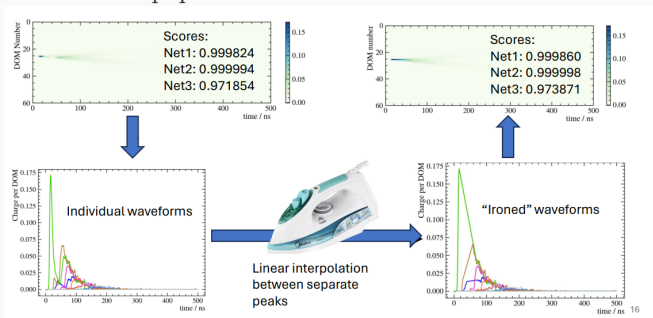
- Получено 7 событий  $\nu_\tau$  (3 события уже рассматривались, где 1 из них было кандидатом в прошлом анализе).
- Нулевая гипотеза (отсутствие астрофизических  $\nu_\tau$ ) отвергается на уровне  $5\sigma$ .

|         | $\nu_{\tau,CC}^{\text{astro}}$ [59] | $\nu_{\text{other}}^{\text{astro}}$ [59] | $\nu_{\text{conv.}}^{\text{atm}}$ [60–63] | $\nu_{\text{prompt}}^{\text{atm}}$ [56, 64–66] | $\mu_{\text{conv.}}^{\text{atm}}$ [67–70] | all background                    |
|---------|-------------------------------------|------------------------------------------|-------------------------------------------|------------------------------------------------|-------------------------------------------|-----------------------------------|
| initial | $160 \pm 0.2$ ( $190 \pm 0.3$ )     | $400 \pm 0.7$ ( $490 \pm 0.8$ )          | $580 \pm 7$                               | $72 \pm 0.1$                                   | $8400 \pm 110$                            | $9450 \pm 110$ ( $9540 \pm 110$ ) |
| final   | $6.4 \pm 0.02$ ( $4.0 \pm 0.02$ )   | $0.3 \pm 0.02$ ( $0.2 \pm 0.01$ )        | $0.1 \pm 0.008$                           | $0.1 \pm 0.001$                                | $0.01 \pm 0.008$                          | $0.5 \pm 0.02$ ( $0.4 \pm 0.02$ ) |

## Предсказания моделирования

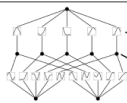
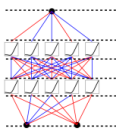
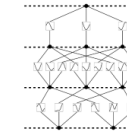


- Добавление: некоррелированных шумов в амплитуды DOM'ов, также коррелированных с глубиной и для целых струн, — во всех случаях миграция типа события мало.
- Атаки с добавлением шума, а также внесение специальных изменений до подмены класса (сигнал  $\rightarrow$  фон 1 из 7; фон  $\rightarrow$  сигнал 1 из 634  $\nu_\tau$ ) не дали существенного влияния.
- Разглаживание форм сигналов с DOM'ов не меняет отклик.



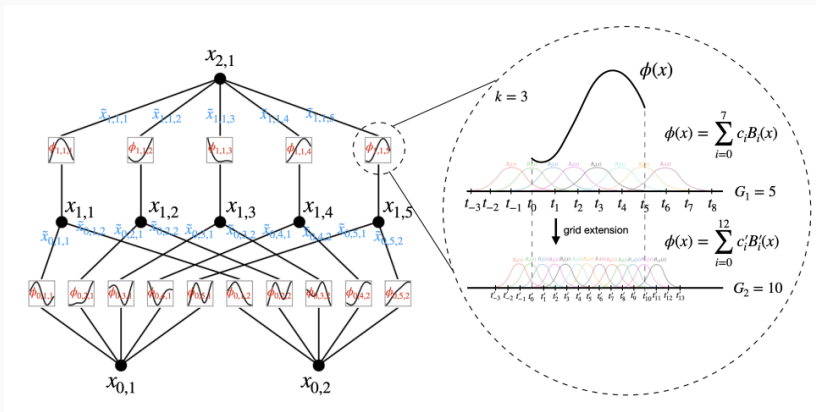
# Kolmogorov-Arnold Networks (KAN)

- [Ссылка](#) на статью: KAN: Kolmogorov–Arnold Networks
- $f(\vec{x}) = \sum_{q=0}^{2n} \Phi(\sum_p^n \phi_{q,p}(x_p))$ .
- KAN заменяет фиксированную функцию активации на обучаемую.
- Главное достоинство: интерпретируемость.

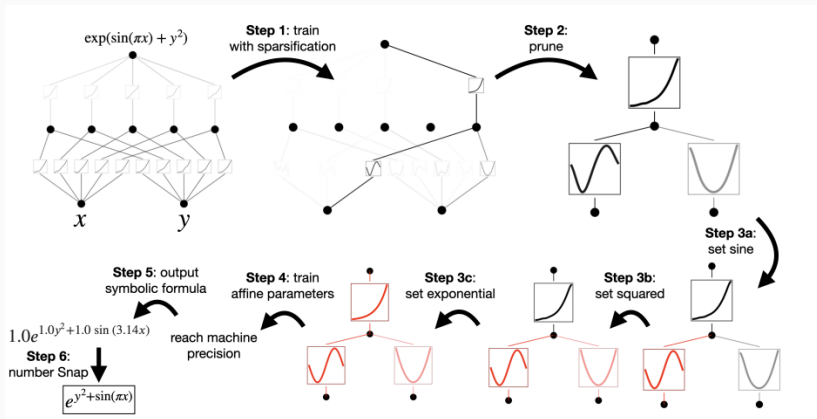
| Model             | Multi-Layer Perceptron (MLP)                                                                                                                                                                                                                                                                                                                  | Kolmogorov-Arnold Network (KAN)                                                                                                                                                                                                                |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Theorem           | Universal Approximation Theorem                                                                                                                                                                                                                                                                                                               | Kolmogorov-Arnold Representation Theorem                                                                                                                                                                                                       |
| Formula (Shallow) | $f(\mathbf{x}) \approx \sum_{i=1}^{N(c)} a_i \sigma(\mathbf{w}_i \cdot \mathbf{x} + b_i)$                                                                                                                                                                                                                                                     | $f(\mathbf{x}) = \sum_{q=1}^{2n+1} \Phi_q \left( \sum_{p=1}^n \phi_{q,p}(x_p) \right)$                                                                                                                                                         |
| Model (Shallow)   | <p>(a)  <i>fixed activation functions on nodes</i><br/><i>learnable weights on edges</i></p>                                                                                                                                                                 | <p>(b)  <i>learnable activation functions on edges</i><br/><i>sum operation on nodes</i></p>                                                                  |
| Formula (Deep)    | $\text{MLP}(\mathbf{x}) = (\mathbf{W}_3 \circ \sigma_2 \circ \mathbf{W}_2 \circ \sigma_1 \circ \mathbf{W}_1)(\mathbf{x})$                                                                                                                                                                                                                     | $\text{KAN}(\mathbf{x}) = (\Phi_3 \circ \Phi_2 \circ \Phi_1)(\mathbf{x})$                                                                                                                                                                      |
| Model (Deep)      | <p>(c)  <i>MLP(x)</i><br/><math>\mathbf{W}_3</math><br/><math>\sigma_2</math><br/><math>\mathbf{W}_2</math><br/><math>\sigma_1</math><br/><math>\mathbf{W}_1</math><br/><math>\mathbf{x}</math><br/><i>nonlinear, fixed</i><br/><i>linear, learnable</i></p> | <p>(d)  <i>KAN(x)</i><br/><math>\Phi_3</math><br/><math>\Phi_2</math><br/><math>\Phi_1</math><br/><math>\mathbf{x}</math><br/><i>nonlinear, learnable</i></p> |

# KAN архитектура

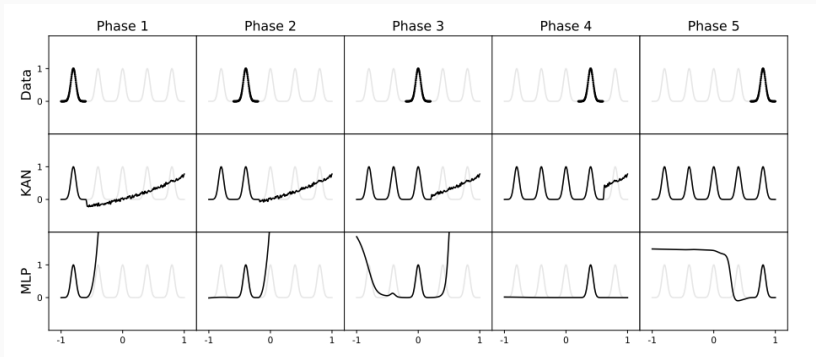
- Никаких весов: каждое ребро использует одномерную функцию.
- Функции активации параметризуются В-сплайнами.



# Символьная регрессия



Аппроксимация выражения  $e^{y^2 + \sin(\pi x)}$ .



Демонстрация невоприимчивости к забыванию. 5 последовательных фаз обучения.

# KANs vs. MLPs

- Точность: малая KAN достигает сравнимой или лучшей точности чем MLP.
- Масштабирование: KAN быстрее масштабируется (добавление одного узла сильнее влияет) теоретически и экспериментально.
- Интерпретируемость: Понятное представление для человека.

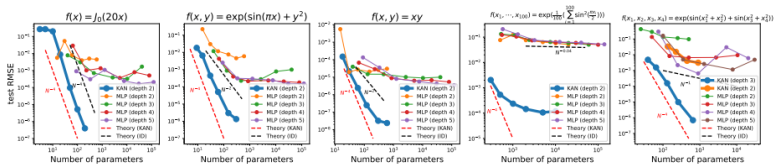


Figure 3.1: Compare KANs to MLPs on five toy examples. KANs can almost saturate the fastest scaling law predicted by our theory ( $\alpha = 4$ ), while MLPs scales slowly and plateau quickly.

# Некоторые примеры KAN

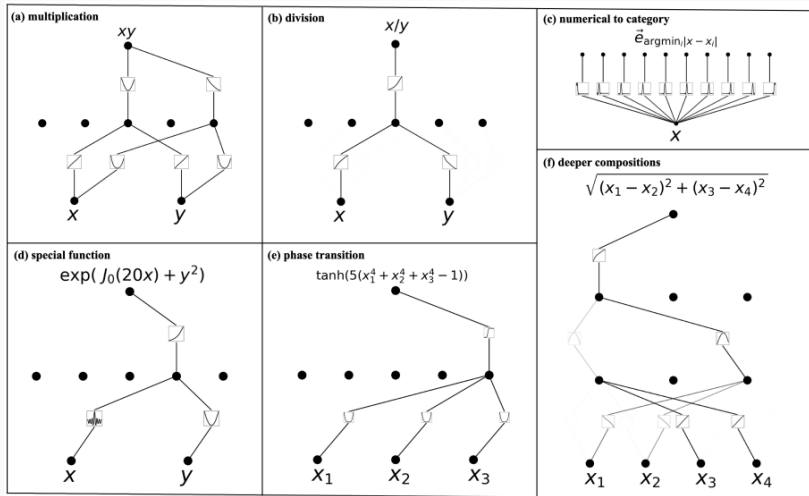
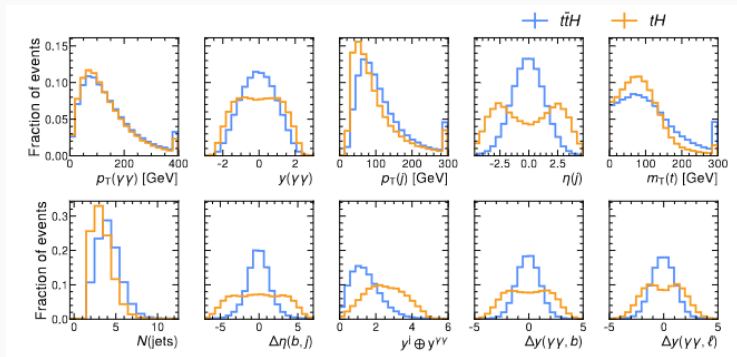


Figure 4.1: KANs are interpretable for simple symbolic tasks

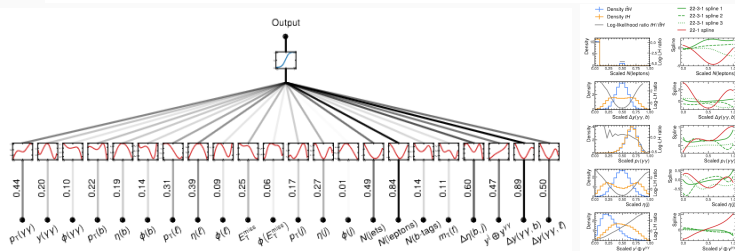
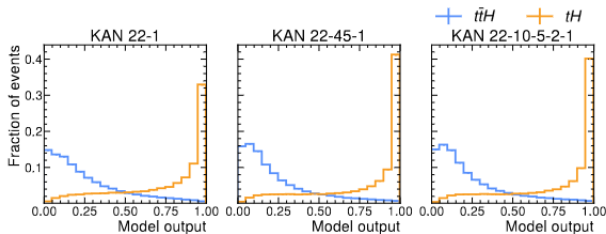
# Пример использования KAN

- [Ссылка](#) на статью: KAN we improve on HEP classification tasks? Kolmogorov-Arnold Networks applied to an LHC physics example
- Классификация событий  $t\bar{t}H$  и  $tH$  ( $H \rightarrow \gamma\gamma$ ) с детектором CMS.
- Использовалось 22 входных параметра.



# Результаты обучения

- Обучались три сети разной структуры: 22-1, 22-45-1, 22-45-10-5-2-1.



# Сравнение с MLP

- Значительного улучшения нет.
- Есть интерпретируемость.
- Но как понимать более глубокие KAN сети?

