

Национальный исследовательский ядерный университет «МИФИ»
Институт Ядерной Физики и Технологий
Кафедра № 40 “Физика элементарных частиц”

Выпускная квалификационная работа магистра на тему

Применение свёрточных нейронных сетей для
реконструкции положения события в эксперименте
DEAP-3600

Выполнил:
Студент гр. М19-115
Ильясов А.И.
Научный руководитель:
к.ф.-м.н.
Гробов А.В.



Москва, 2021



Оглавление

1) Введение

- a) Темная материя Вселенной
- b) Детекторы по поиску темной материи
- c) Детектор DEAP-3600
- d) Описание использующихся алгоритмов и цель работы

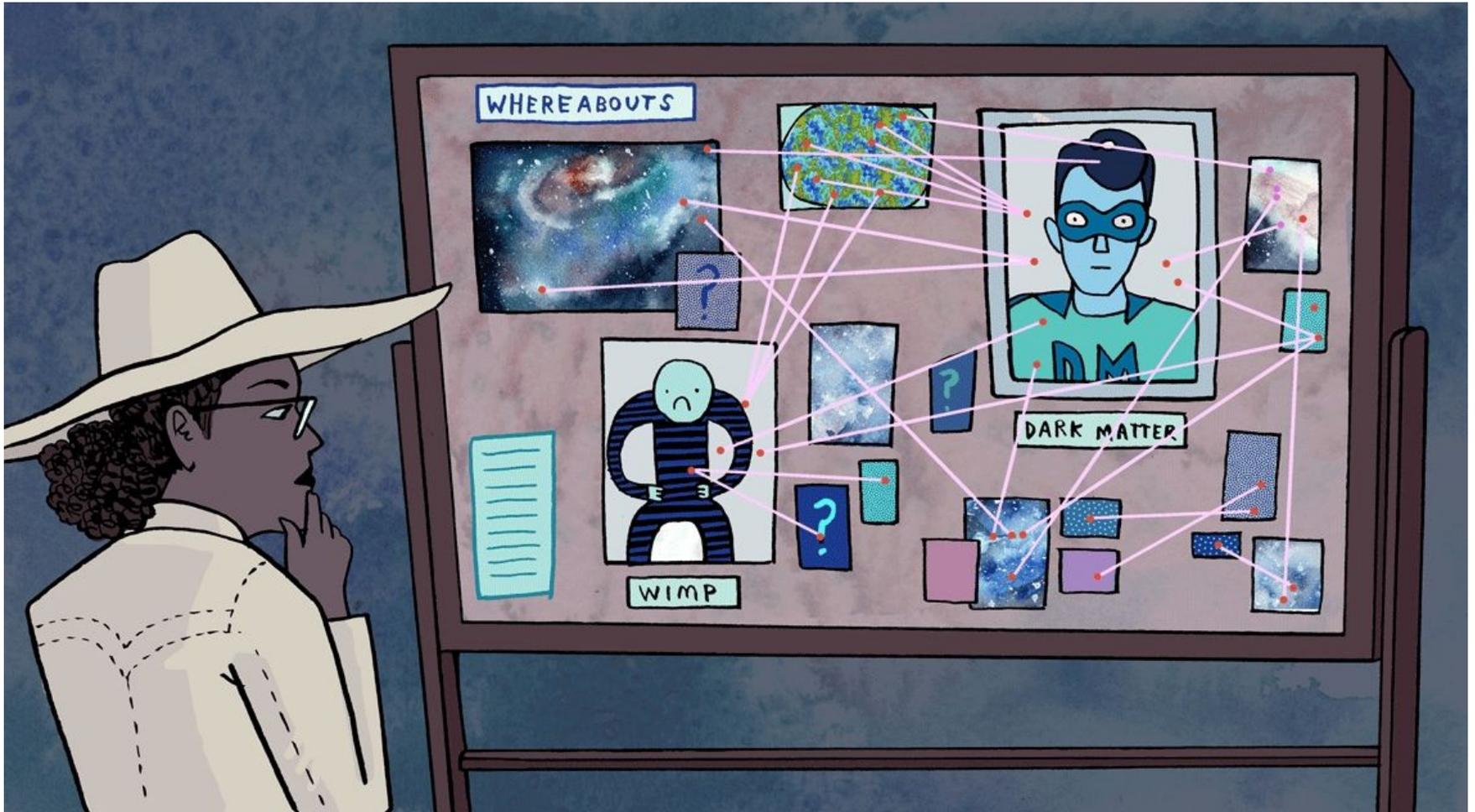
2) Исследуемые алгоритмы реконструкции

- a) Полносвязная нейронная сеть
- b) Нейронная сеть с короткими связями
- c) Нейронная сеть со сверточными слоями

3) Полученные результаты

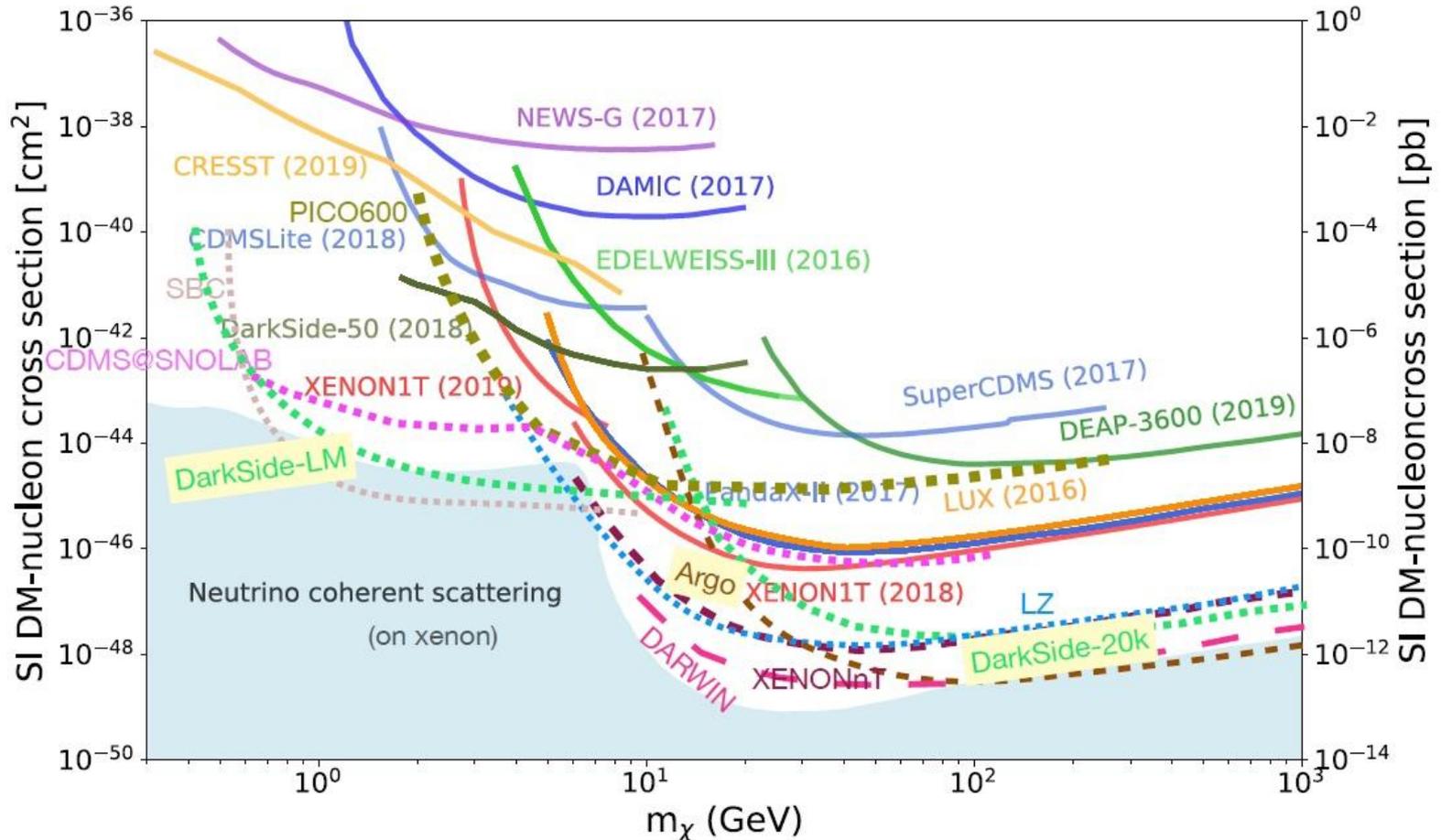
4) Заключение

Введение



Детекторы по поиску темной материи

Particle Data Group et al. Review of particle physics //Progress of Theoretical and Experimental Physics. – 2020. – T. 2020. – №. 8. – C. 083C01.



$$\Omega_{\Lambda} = (68.47 \pm 0.73)\% \quad \Omega_b = (4.92 \pm 0.03)\%$$

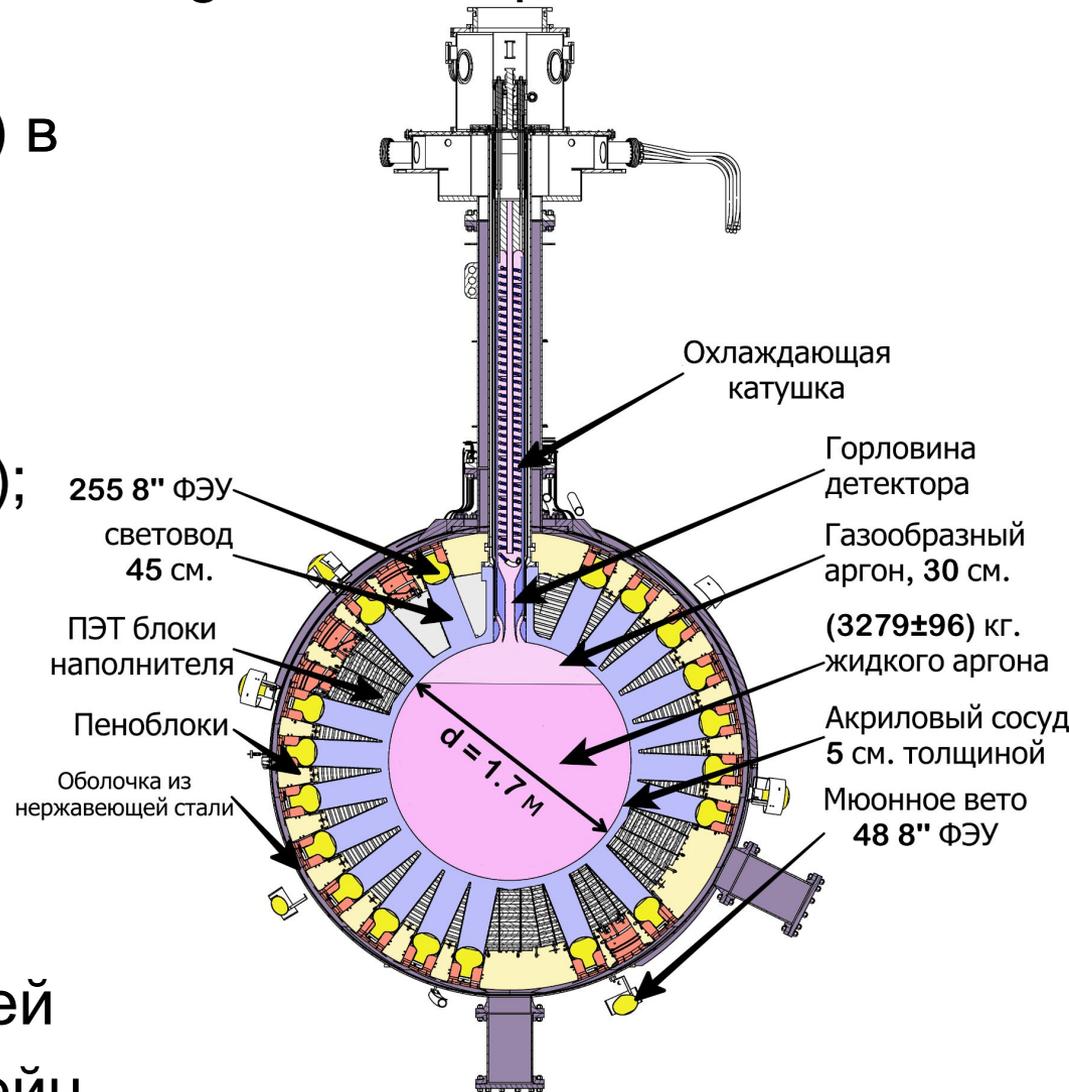
$$\Omega_c = (26.42 \pm 0.26)\%$$

Детектор DEAP-3600

The DEAP Collaboration, Design and Construction of the DEAP-3600 Dark Matter Detector, *Astropart. Phys.* 108, 1 (2019).

Dark matter Experiment in Argon using Pulse-shape discrimination

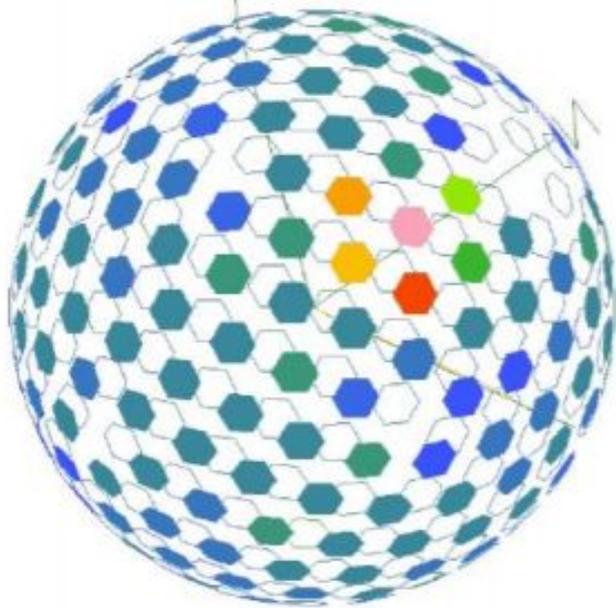
- 2 км. под землей;
- (3279 ± 96) кг. LAr (84 K) в UVA акриловом сосуде толщиной 5 см. и диаметром 1.7 м.;
- 255 + 48 ФЭУ (240-290 K);
- 45 см световоды
- **Нейтронная и черенковская мюонная защита;**
- Детектор заключен в оболочку из нержавеющей стали и погружен в бассейн 300 т. воды.



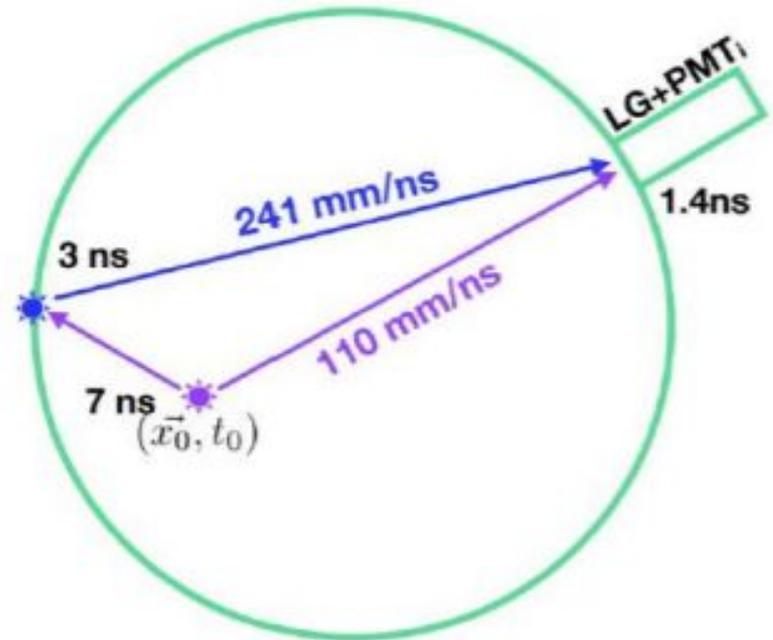
Имеющиеся алгоритмы

The DEAP Collaboration, Search for dark matter with a 231-day exposure of liquid argon using DEAP-3600 at SNOLAB, Physical Review D 100.2 (2019)

1. Метод, основанный на количестве выделившихся фотоэлектронов (MBLikelihood - MBL)



2. Метод, основанный на временном распределении регистрации фотоэлектронов (TimeFit2 - TF)



I. После **фидуциализации** ошибка первого алгоритма около **13%**;

II. Алгоритмы согласуются по Z-координате с **точностью** до 35 мм. для 50% событий;

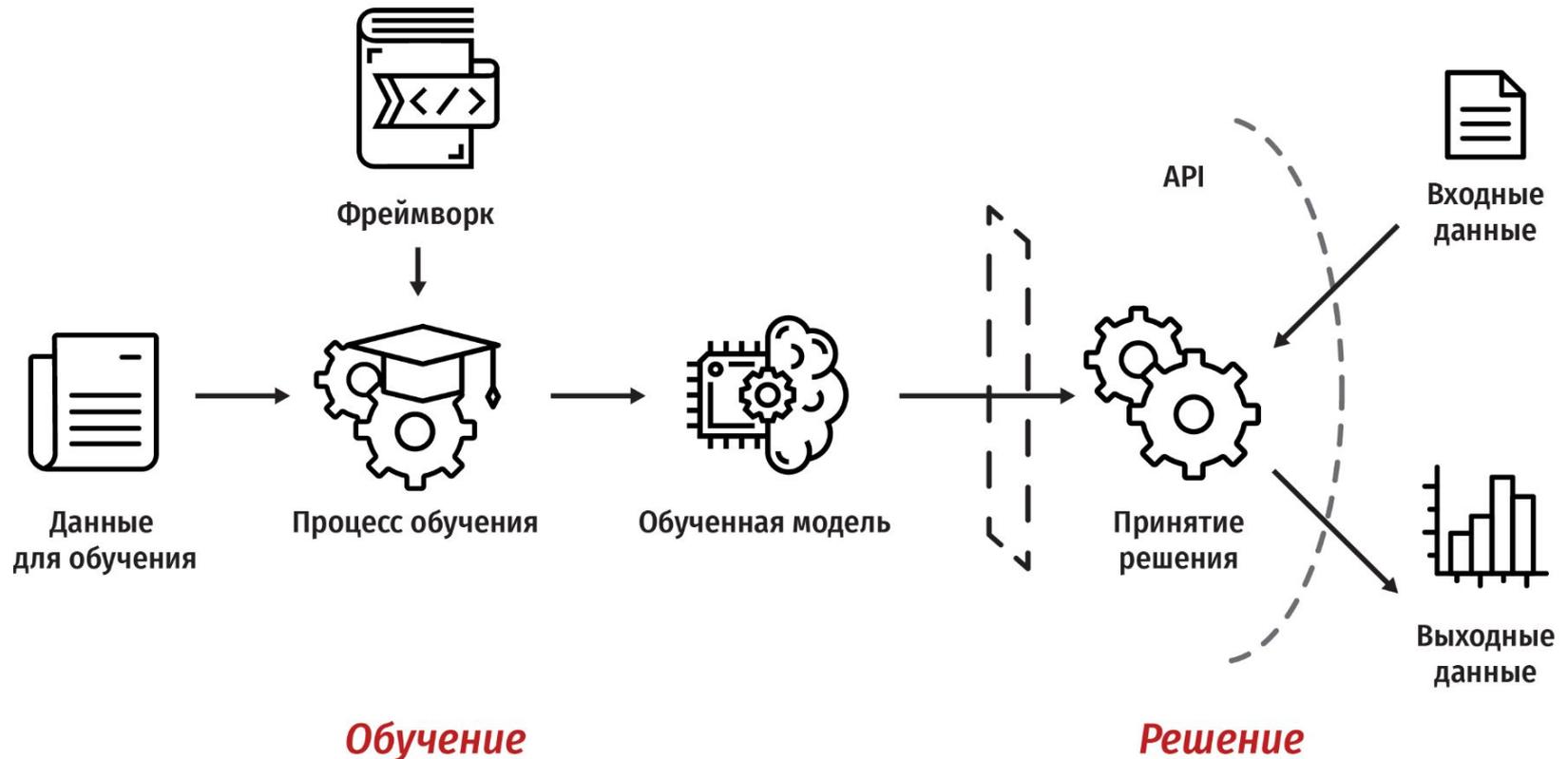
III. **Второй** алгоритм для **верхней части** детектора, **первый** алгоритм - для **нижней части** детектора

Цель работы

Создание алгоритма на основе алгоритмов машинного обучения, который будет:

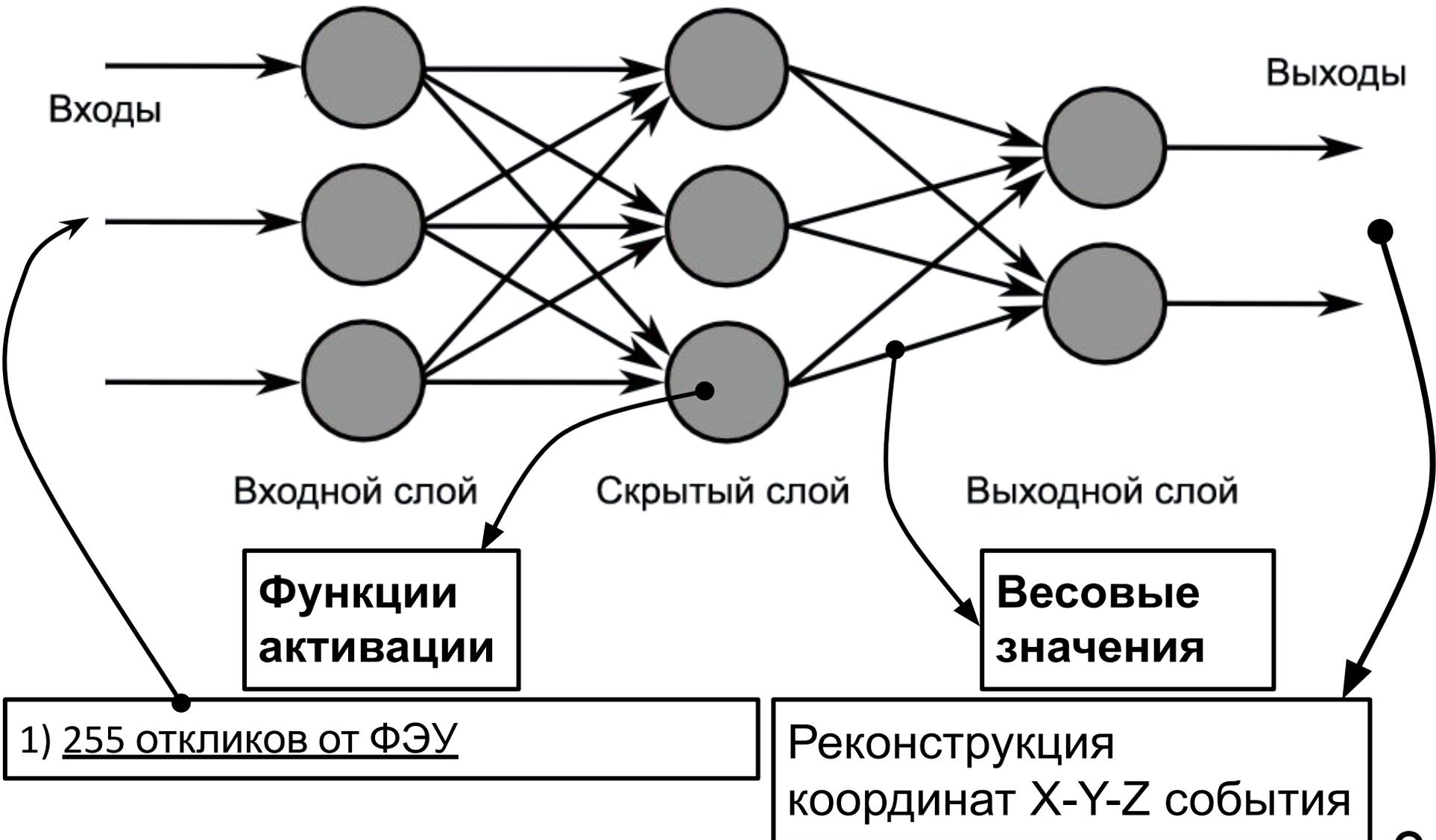
- 1) **использоваться** во всём объеме детектора $R < 850$ мм. Это позволит улучшить реконструкцию во всем детекторе и увеличить чувствительный объем детектора;
- 2) **работать точнее**, чем существующий алгоритм реконструкции;
- 3) **дополнять и поддерживать** существующий алгоритм.

Исследуемые алгоритмы



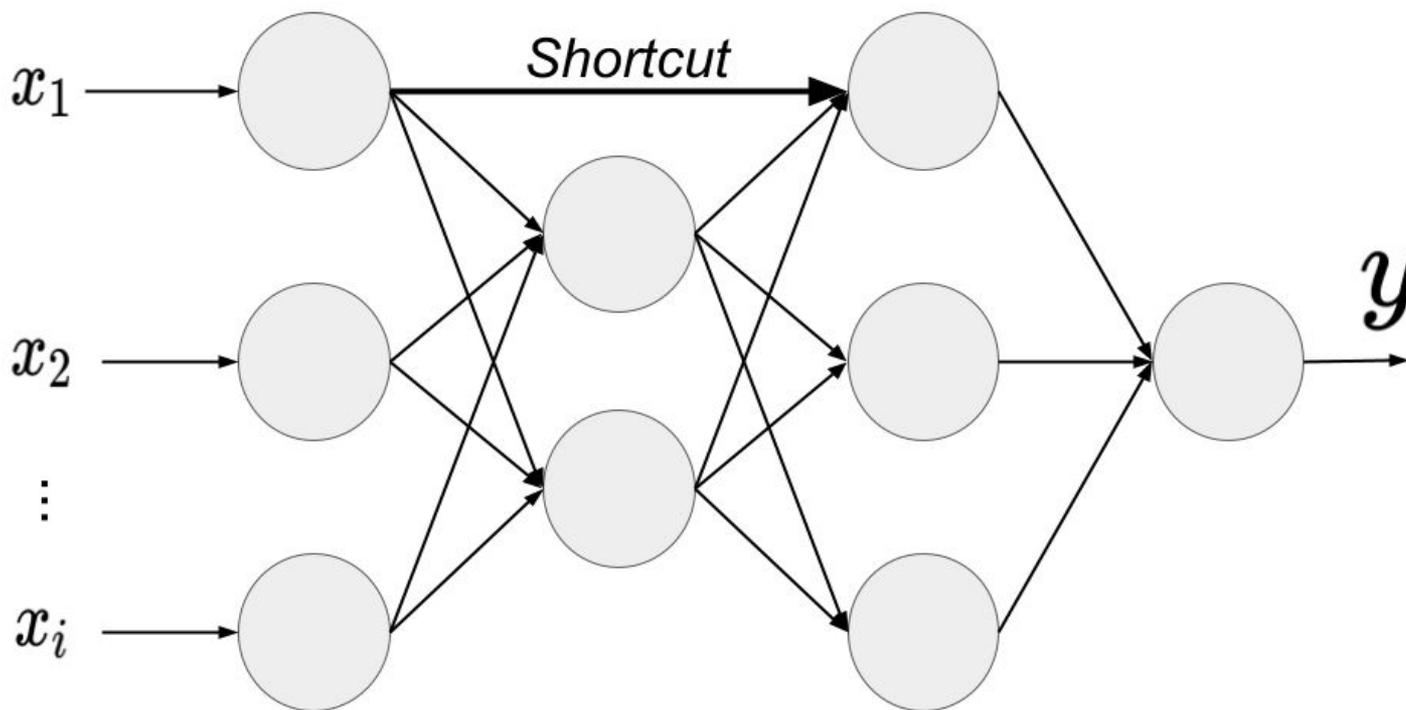
Полносвязная нейронная сеть

Hopfield J. J. *Neural networks and physical systems with emergent collective computational abilities* // *Proceedings of the national academy of sciences.* – 1982. – Т. 79. – №. 8. – С. 2554-2558.



Нейронная сеть с короткими связями

Li Y. et al. A concatenating framework of shortcut convolutional neural networks //arXiv preprint arXiv:1710.00974. – 2017.

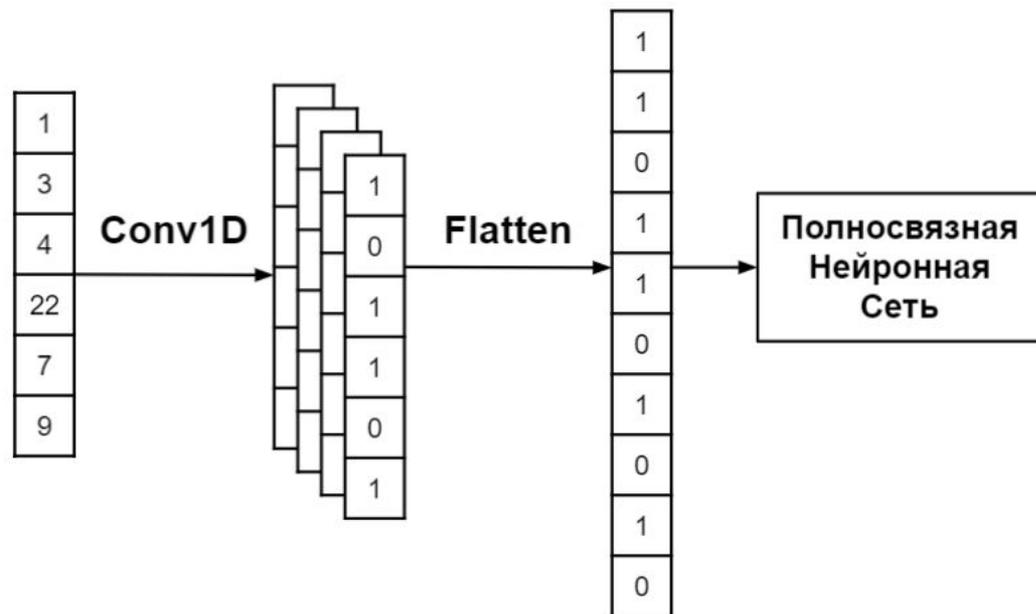


Преимущество:
Меньше теряется
первоначальная
информация о данных

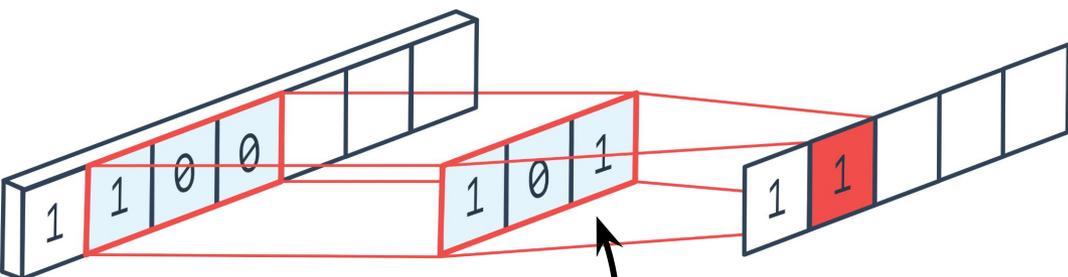
Недостатки:
1) Необходимость соответствия
длин векторов начала и конца
shortcut-связи
2) Длительность тренировки

1D Сверточная нейронная сеть

LeCun Y. et al. Backpropagation applied to handwritten zip code recognition //Neural computation. – 1989. – Т. 1. – №. 4. – С. 541-551.



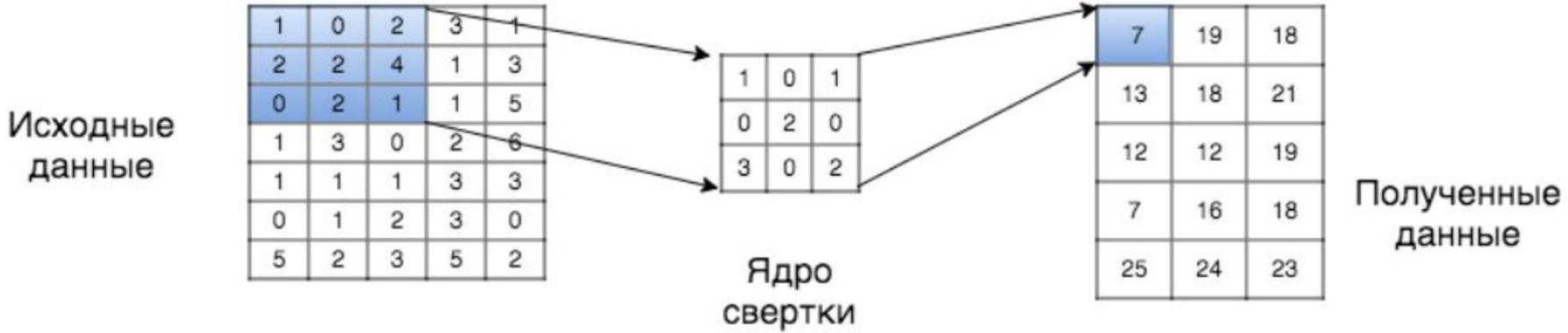
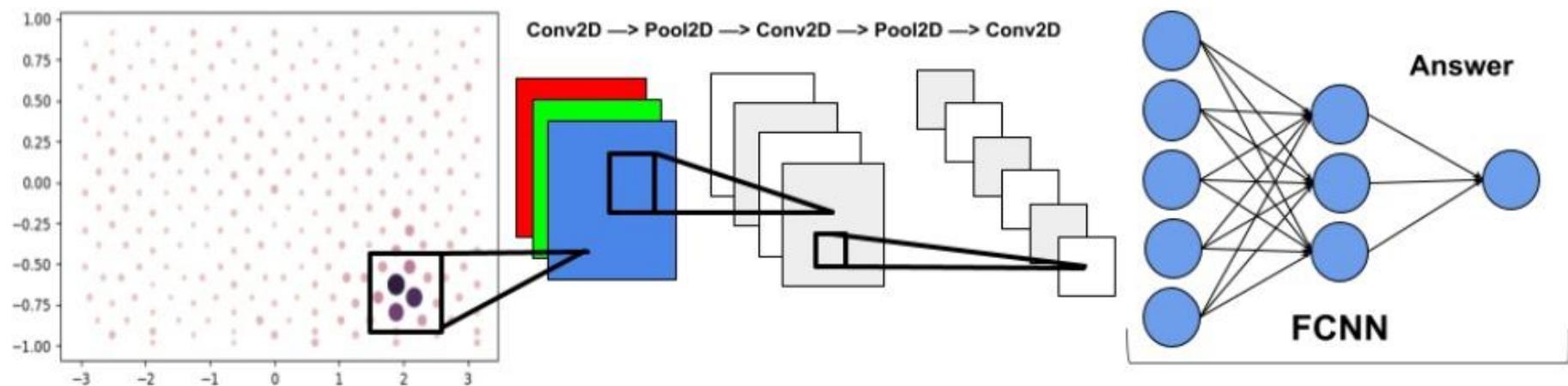
- 1) Слой **Conv1D** - свёртка входного вектора с “ядром”
- 2) **flatten** - разглаживание многомерной матрицы в одномерный вектор
- 3) **Dense** - полносвязная нейронная сеть для окончательного определения координаты



Ядро свёртки

2D Сверточная нейронная сеть

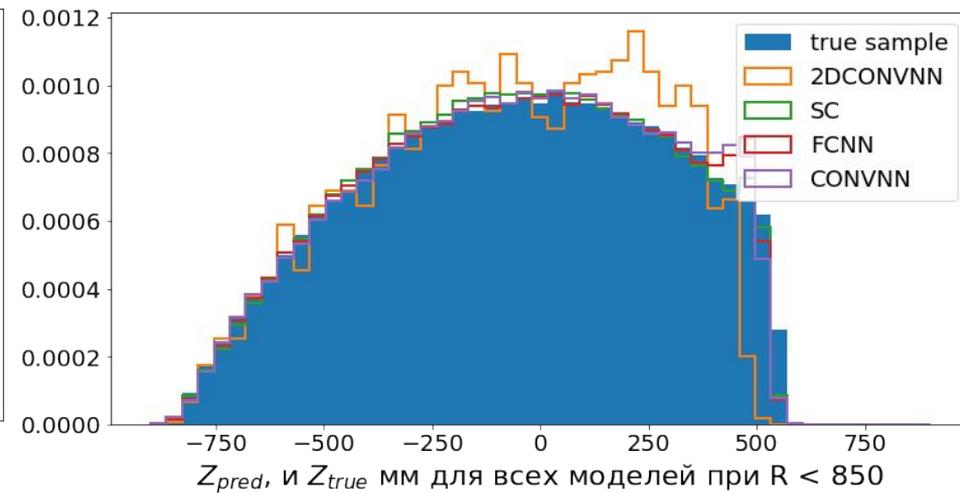
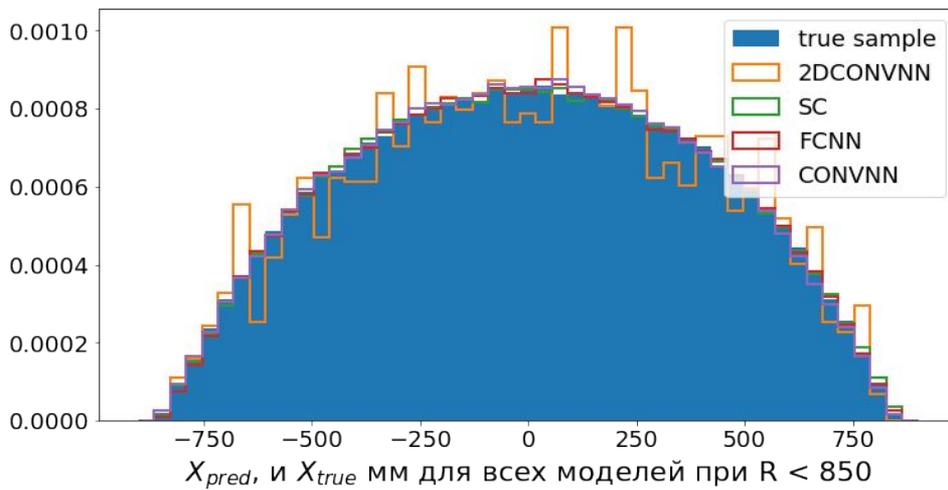
Matsugu M. et al. Subject independent facial expression recognition with robust face detection using a convolutional neural network //Neural Networks. – 2003. – Т. 16. – №. 5-6. – С. 555-559.



Полученные результаты

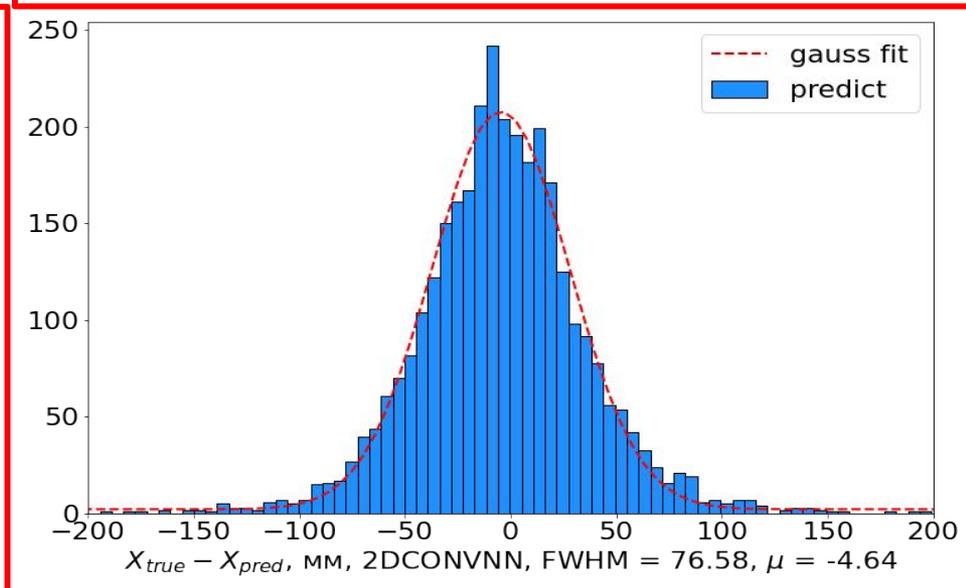
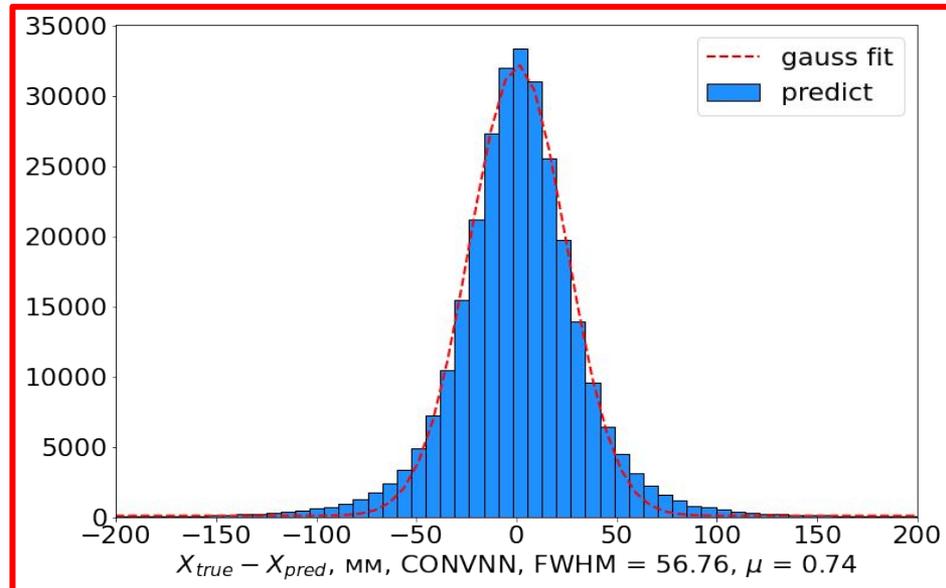
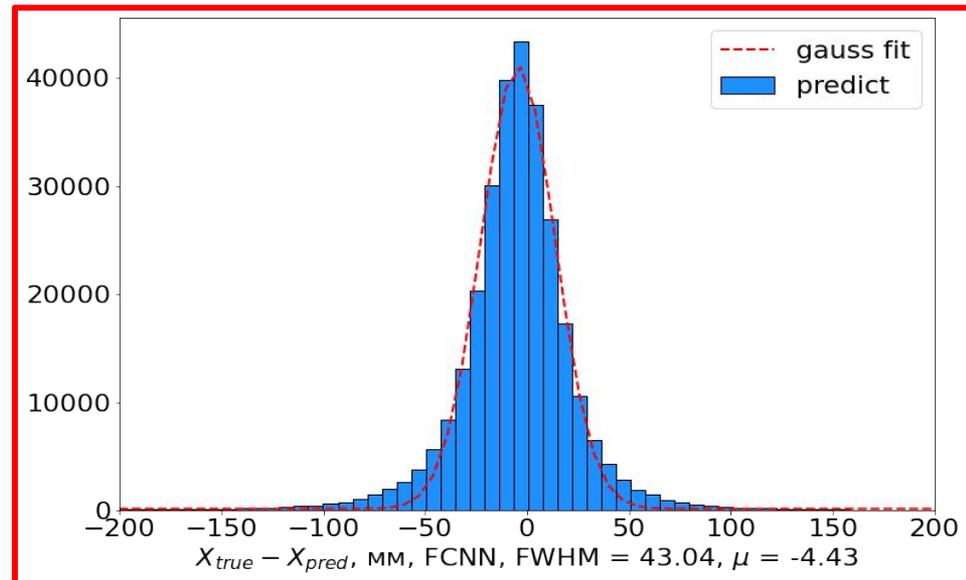
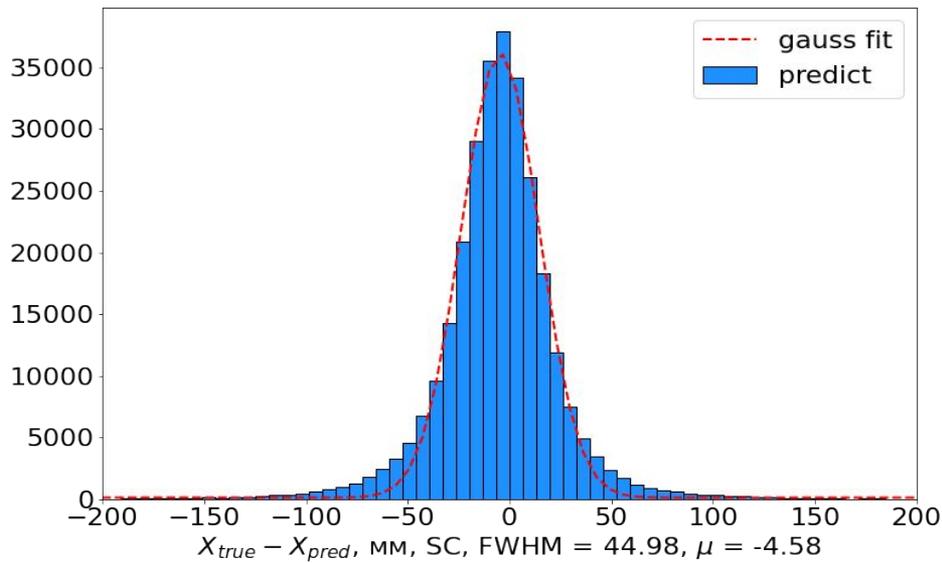


Полученные результаты: работоспособность

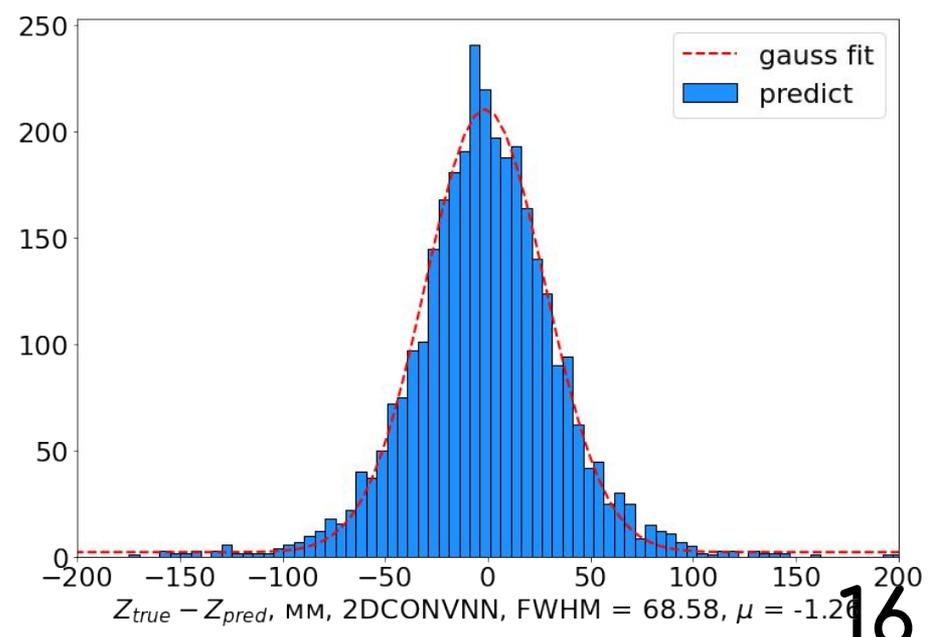
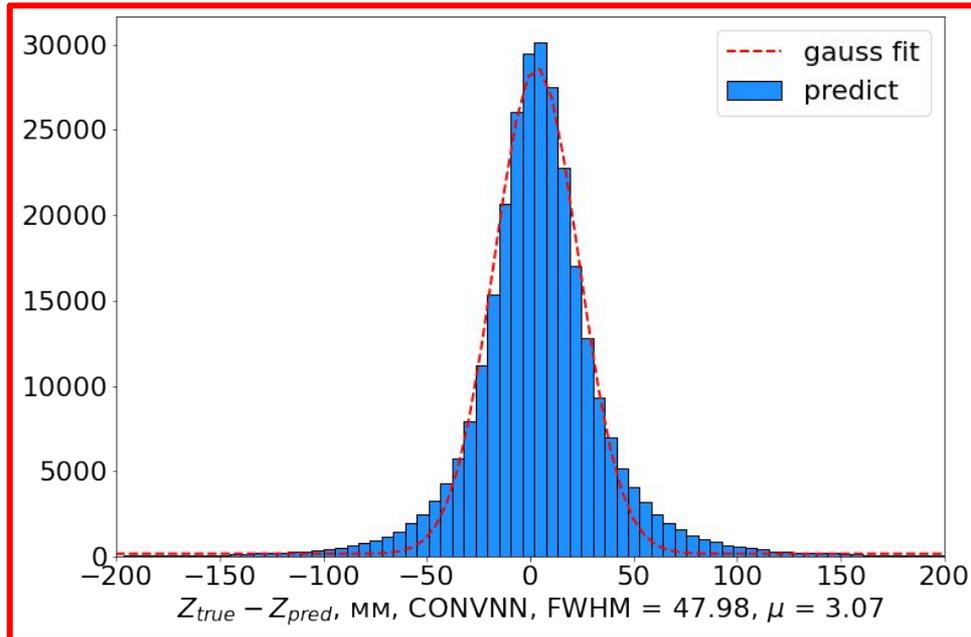
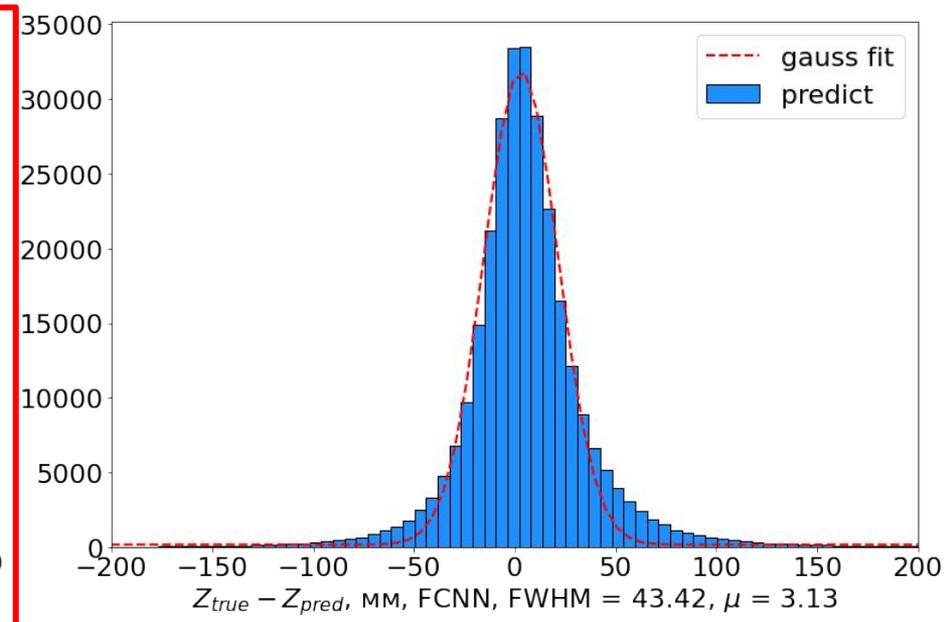
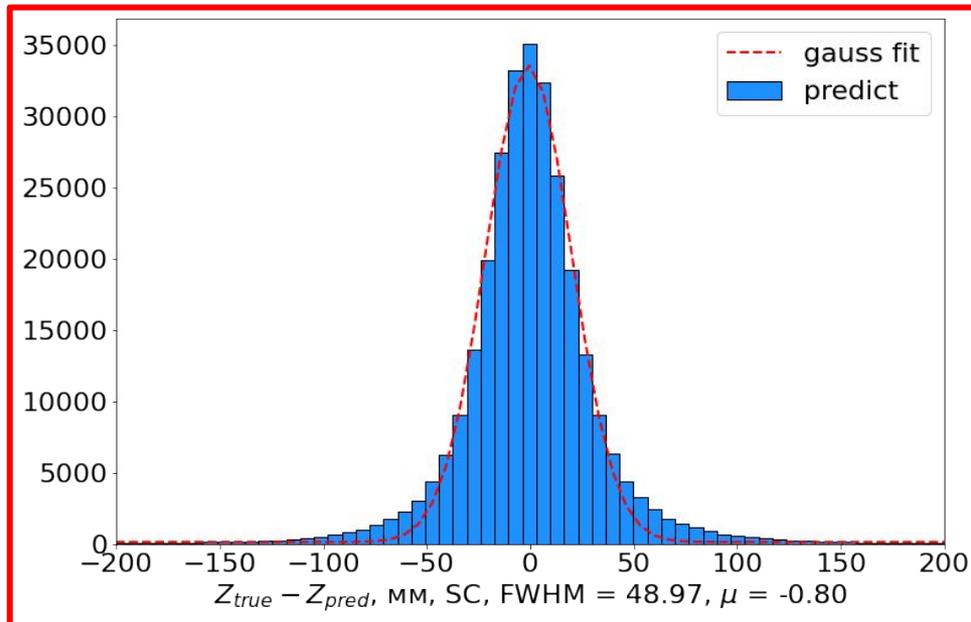


	Точность	X_{FWHM} , мм	Y_{FWHM} , мм	Z_{FWHM} , мм
$R < 850$	0.959 ± 0.003	47 ± 4	49 ± 3	51 ± 7
$R < 720$	0.944 ± 0.001	64 ± 3	64 ± 7	66 ± 2
$R < 630$	0.933 ± 0.002	75 ± 3	76 ± 6	78 ± 2
$R < 430$	0.899 ± 0.007	91 ± 7	92 ± 8	105 ± 15
$720 < R < 850$	0.983 ± 0.003	41 ± 9	42 ± 8	42 ± 9
$630 < R < 720$	0.966 ± 0.005	55 ± 5	55 ± 8	42 ± 5
$430 < R < 630$	0.952 ± 0.002	79 ± 10	77 ± 8	58 ± 2
$200 < R < 430$	0.913 ± 0.004	90 ± 5	88 ± 3	98 ± 7

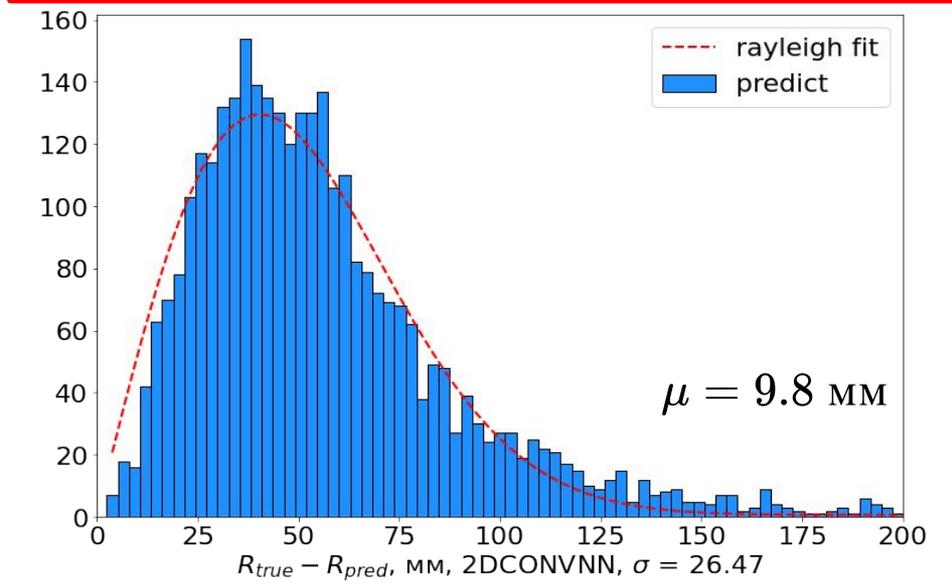
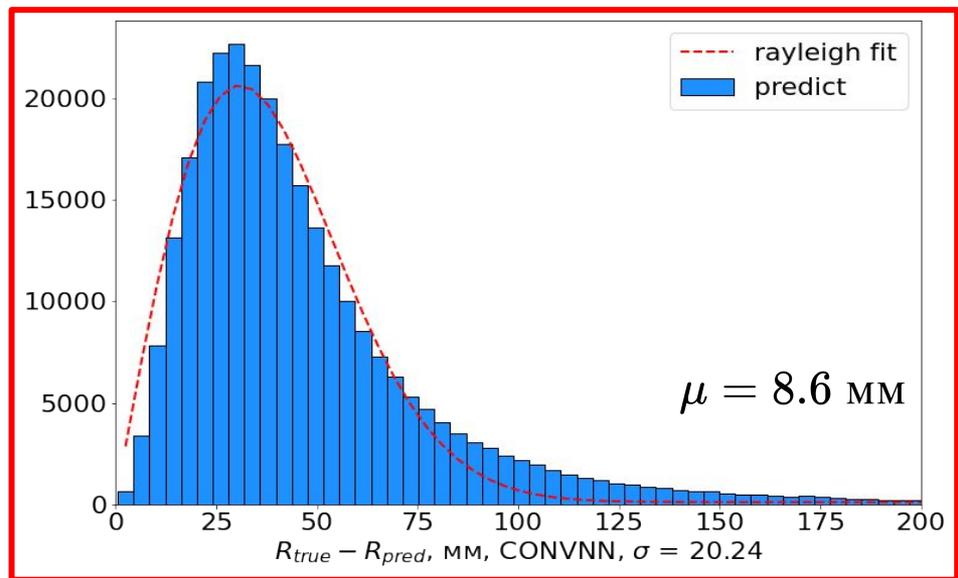
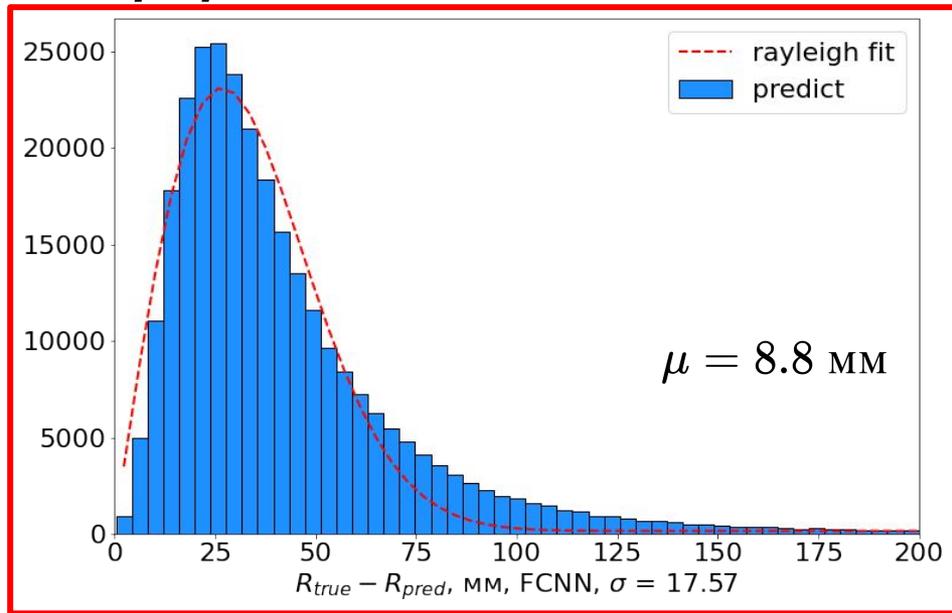
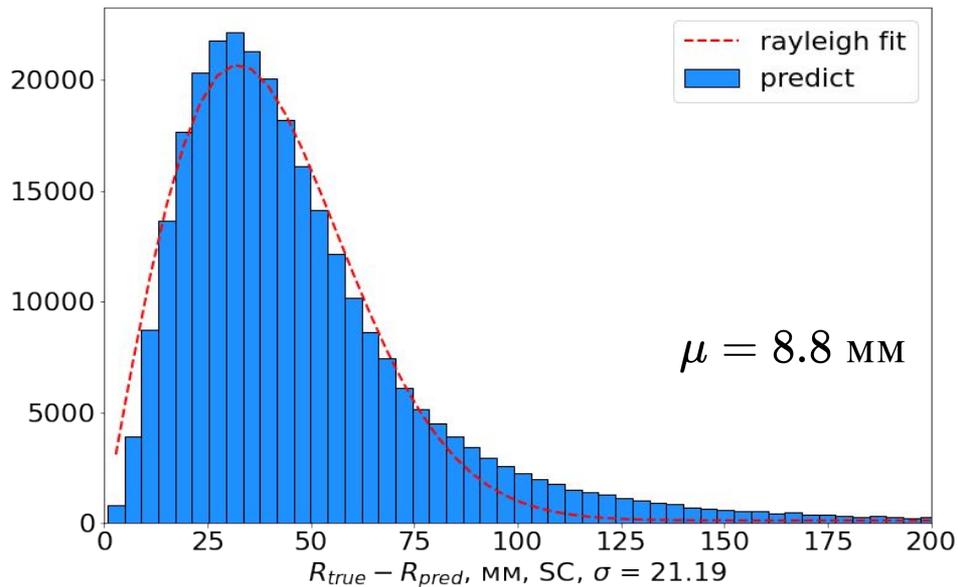
Полученные результаты: эффективность



Полученные результаты: эффективность

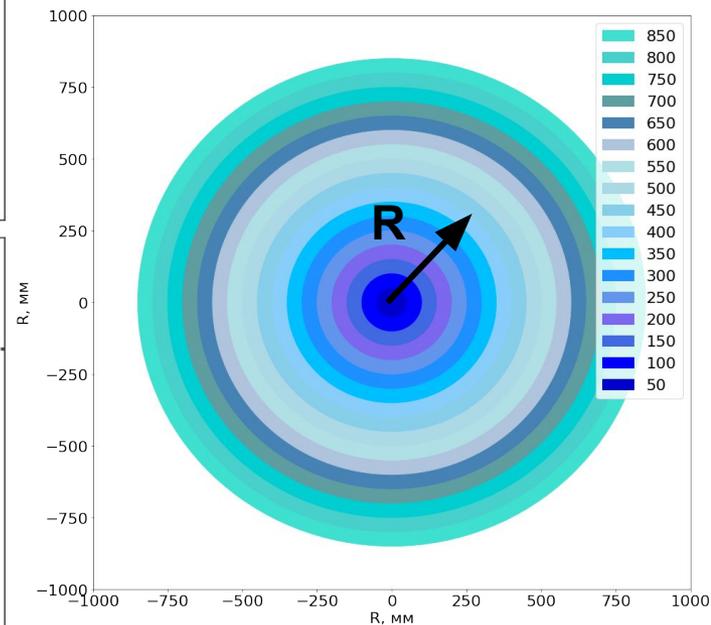
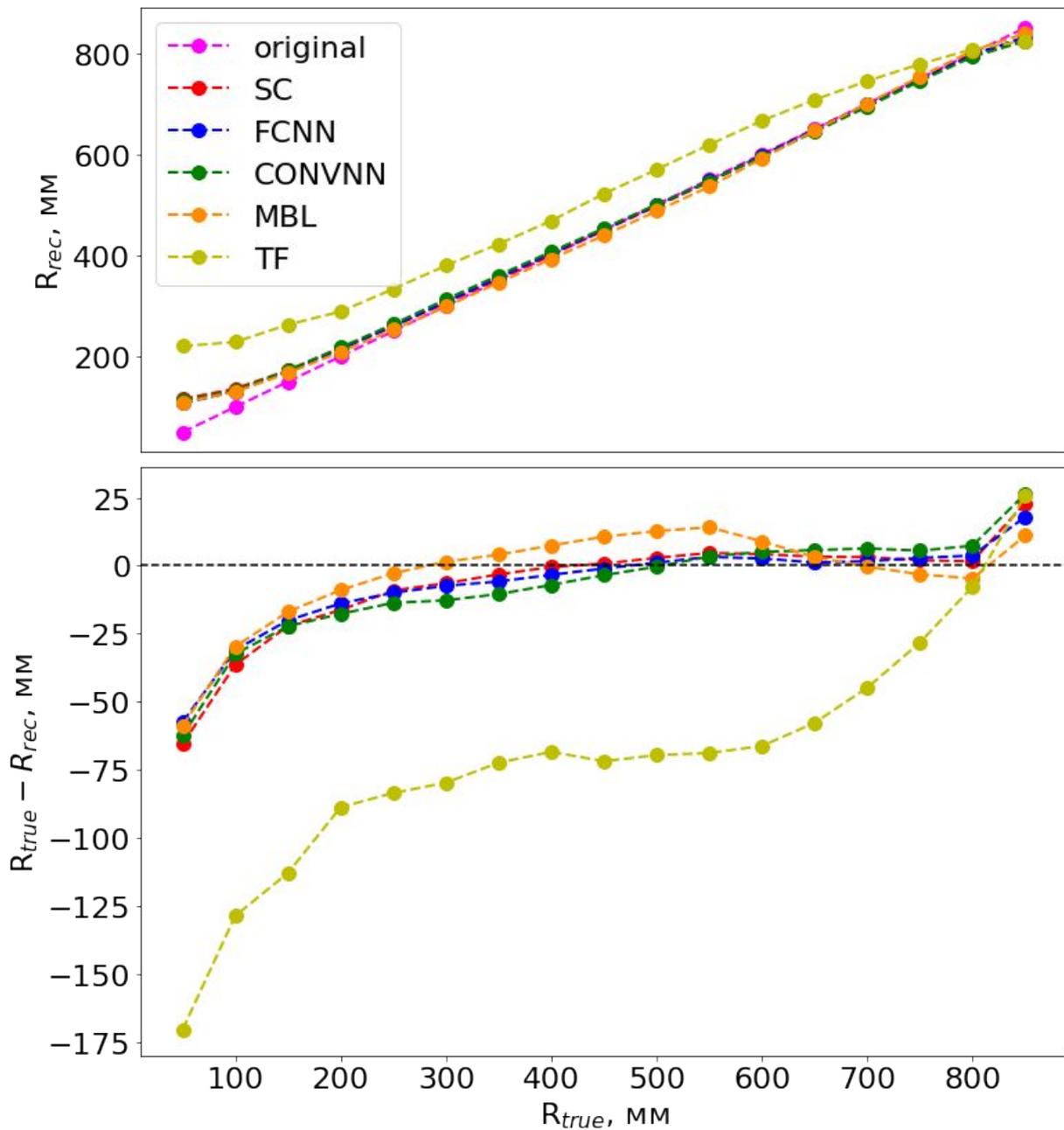


Полученные результаты: эффективность

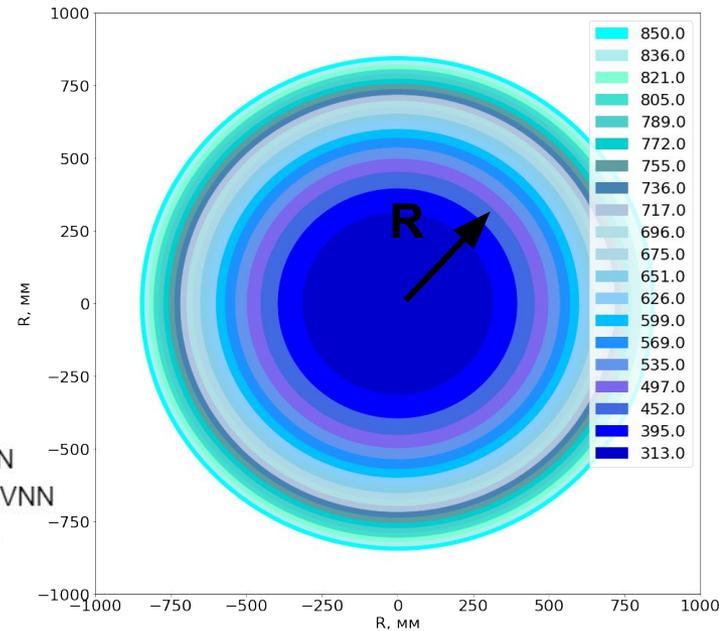
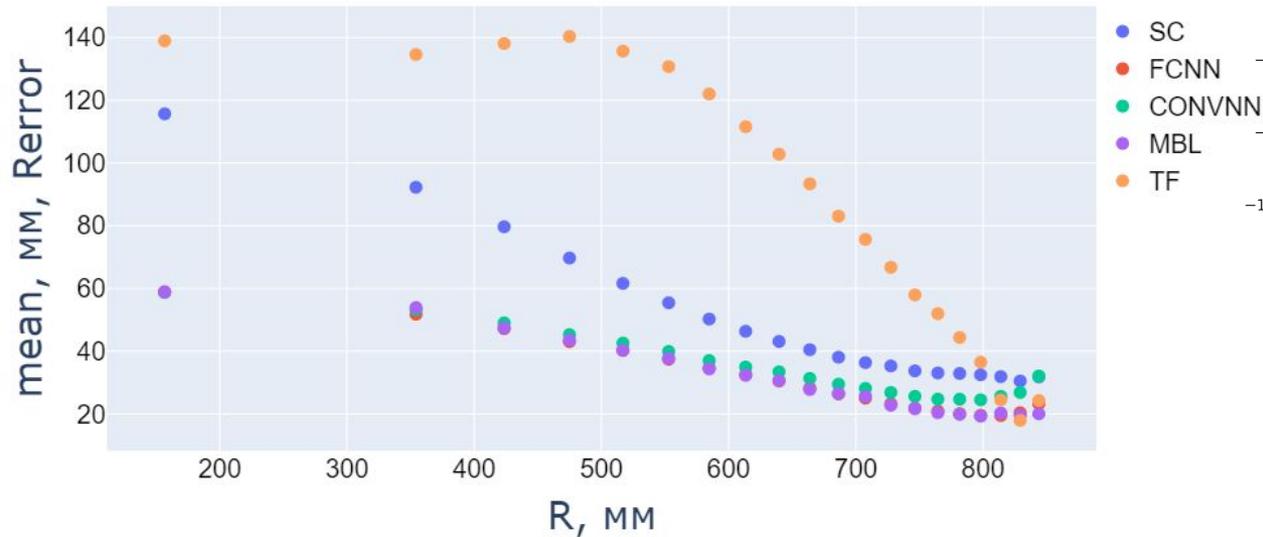
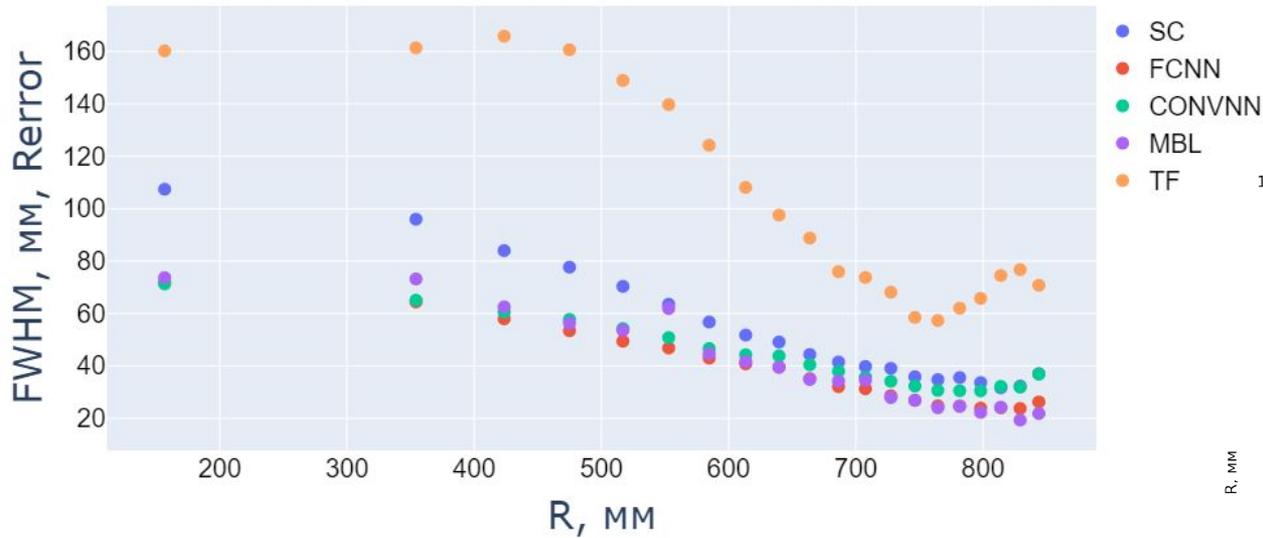


$$R_{true} - R_{pred} = \sqrt{(X_{true} - X_{pred})^2 + (Y_{true} - Y_{pred})^2 + (Z_{true} - Z_{pred})^2} \quad 17$$

Полученные результаты: равные радиусы



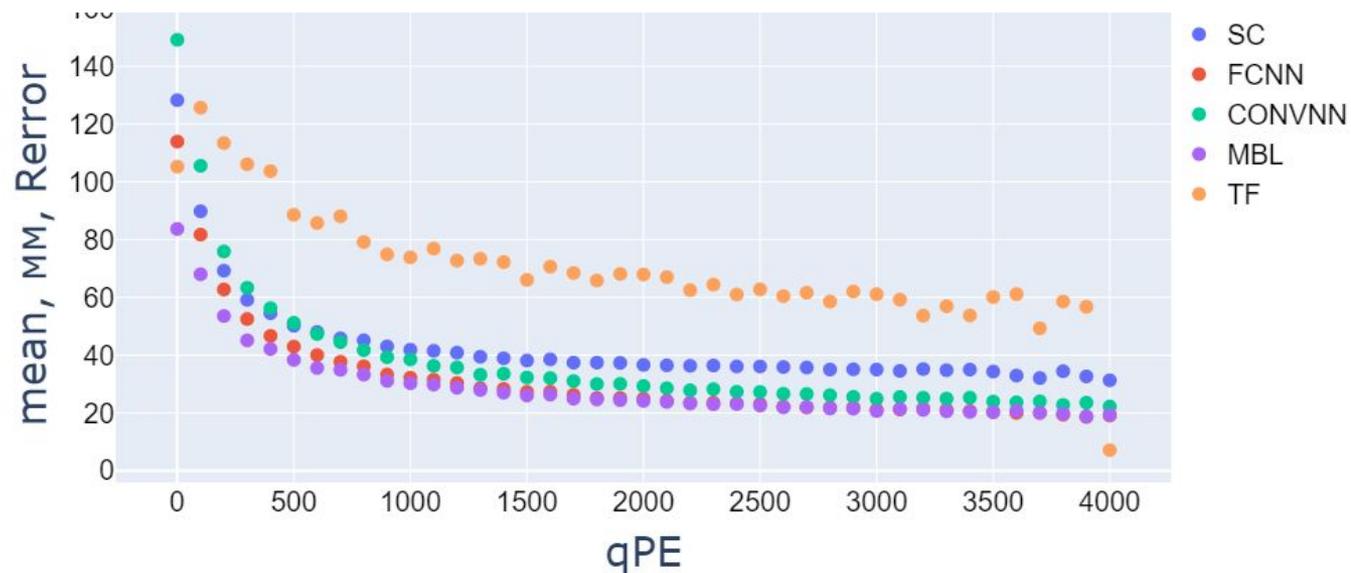
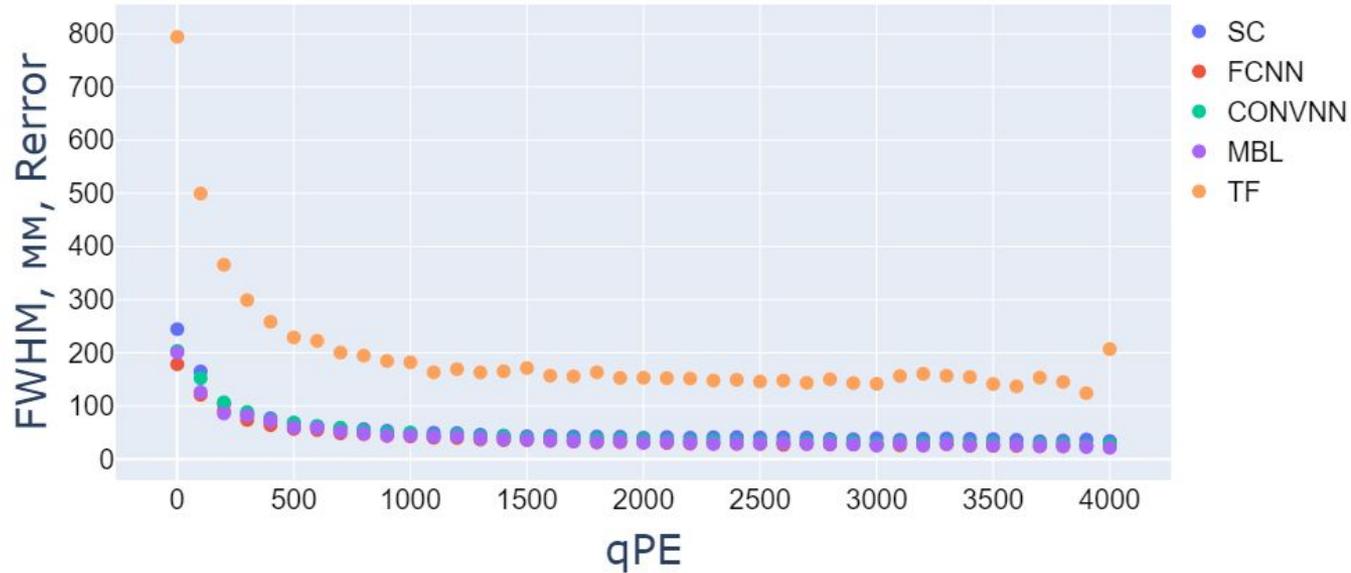
Полученные результаты: равные объемы



$$R_{true} - R_{pred} = \sqrt{(X_{true} - X_{pred})^2 + (Y_{true} - Y_{pred})^2 + (Z_{true} - Z_{pred})^2} \quad 19$$

Полученные результаты: равные энергии

qPE - Общее количество зарегистрированных фотоэлектронов

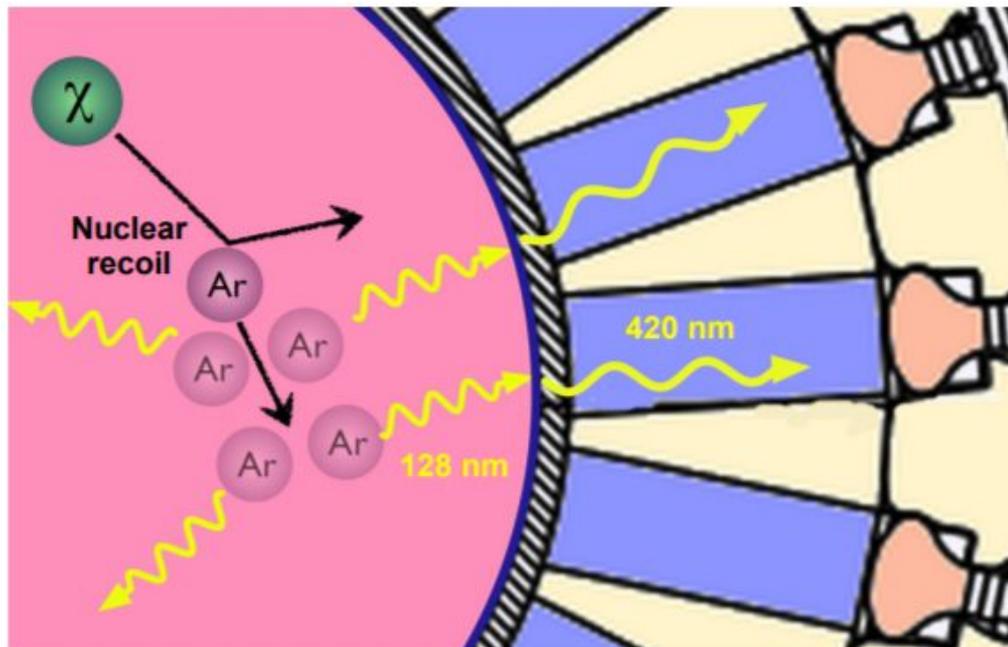
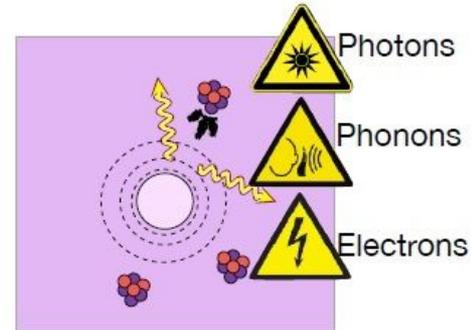
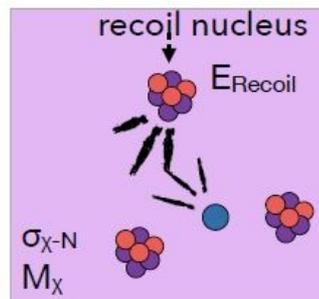
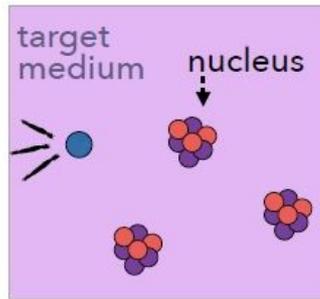
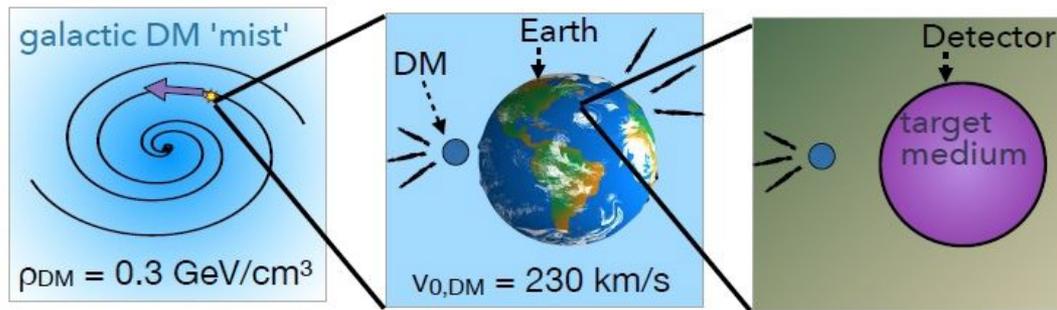


$$R_{true} - R_{pred} = \sqrt{(X_{true} - X_{pred})^2 + (Y_{true} - Y_{pred})^2 + (Z_{true} - Z_{pred})^2} \quad 20$$

Заключение

- 1) Получены данные Монте-Карло моделирования естественной радиоактивности аргона-39;
- 2) Было протестировано четыре модели машинного обучения: полносвязная нейронная сеть, нейронная сеть с короткими связями, одномерная и двумерная сверточная нейронная сеть. Наилучшим образом работает сверточная нейронная сеть;
- 3) Все алгоритмы выполняют реконструкцию положения во всём объеме детектора $R < 850$ мм;
- 4) Средняя ошибка по всем трем координатам составляет не более 6 мм, с разрешением FWHM около 50 мм для 76% всех событий
- 5) Сравнение моделей с используемыми алгоритмами показало, что во всех областях детектора по R и по Q алгоритм работает эффективно и даже лучше в области $300 < R < 600$ мм.

*Дополнительные картинки
графики
цифры
то есть
Back-up слайды*



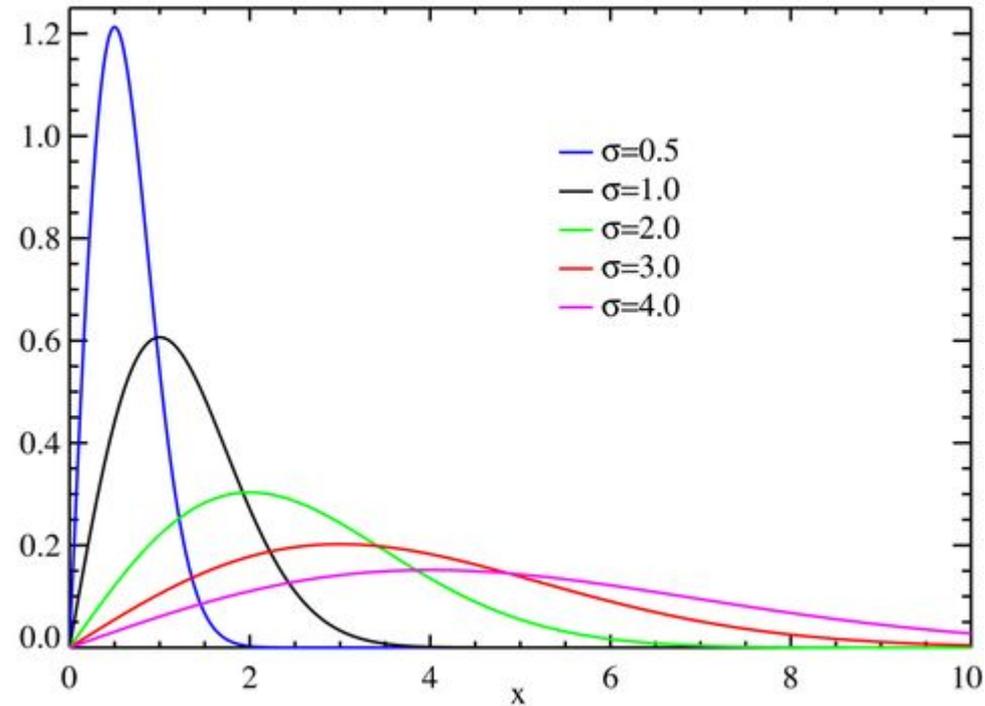
Распределение Рэля

$$\sigma_g = (2 - \pi/2)\sigma^2$$

$$\mu_g = \sqrt{\pi/2}\sigma$$

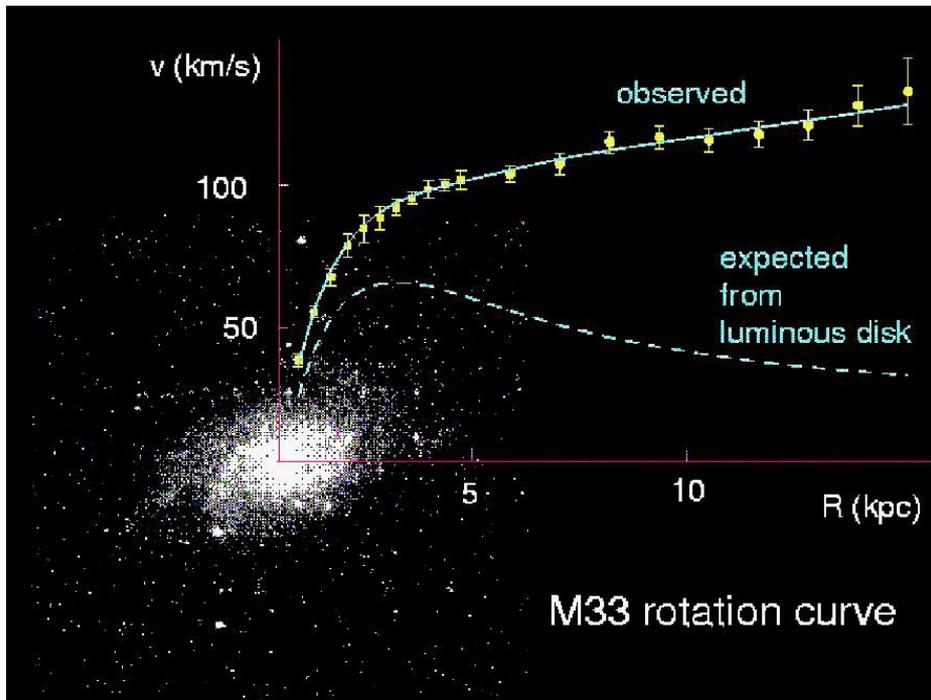
$$f(x; \sigma) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

$$x \geq 0, \sigma > 0$$



Тёмная материя Вселенной

Zwicky F. On the Masses of Nebulae and of Clusters of Nebulae //The Astrophysical Journal. – 1937. – Т. 86. – С. 217.



$$U_G = -2\varepsilon_{kin} \quad U_G \approx \frac{-GM^2}{r}$$

$$\varepsilon_{kin} = \frac{M\bar{v}^2}{2} \quad M \approx \frac{\bar{v}^2 r}{G}$$

$$\frac{GmM}{r^2} = \frac{mv^2}{r}$$

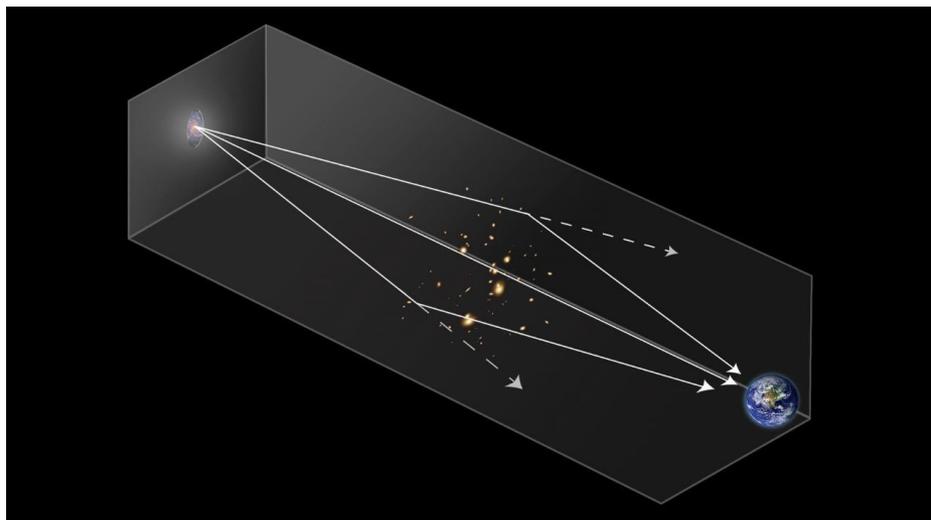
$$v(r) = \sqrt{\frac{GM}{r}} \quad M = \bar{\rho} \frac{4}{3} \pi r^3$$

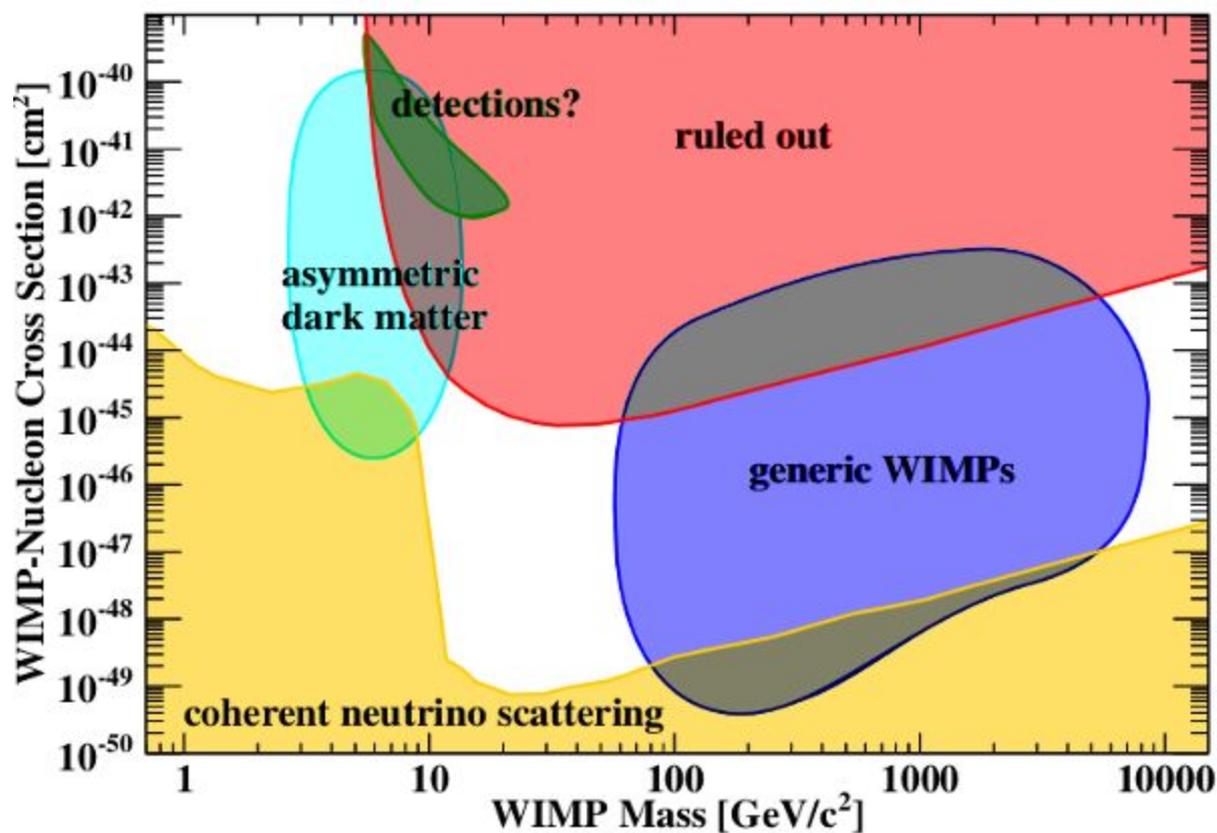
$$v(r) \sim \sqrt{\frac{M}{r}} \sim r \quad v(r) \sim \frac{1}{\sqrt{r}}$$

$$\Omega_\Lambda = (68.47 \pm 0.73)\%$$

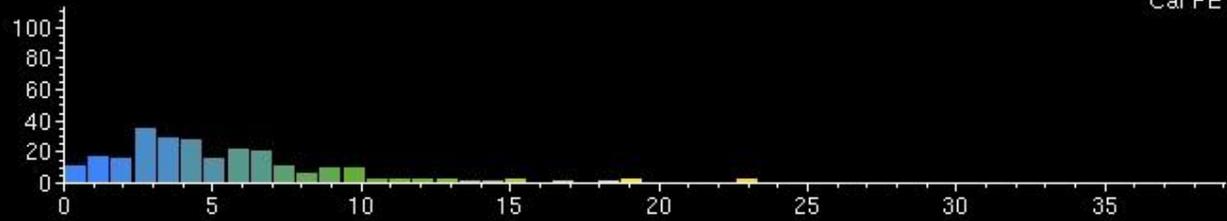
$$\Omega_c = (26.42 \pm 0.26)\%$$

$$\Omega_b = (4.92 \pm 0.03)\%$$

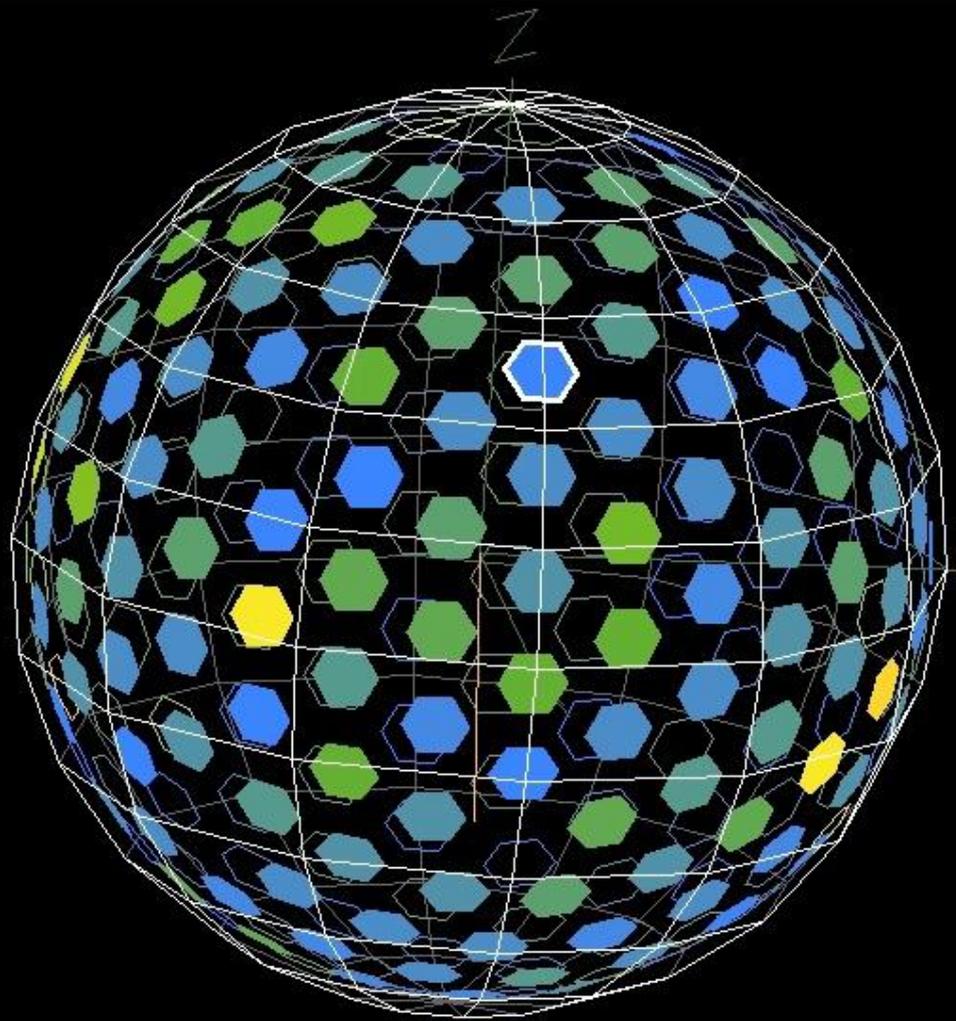


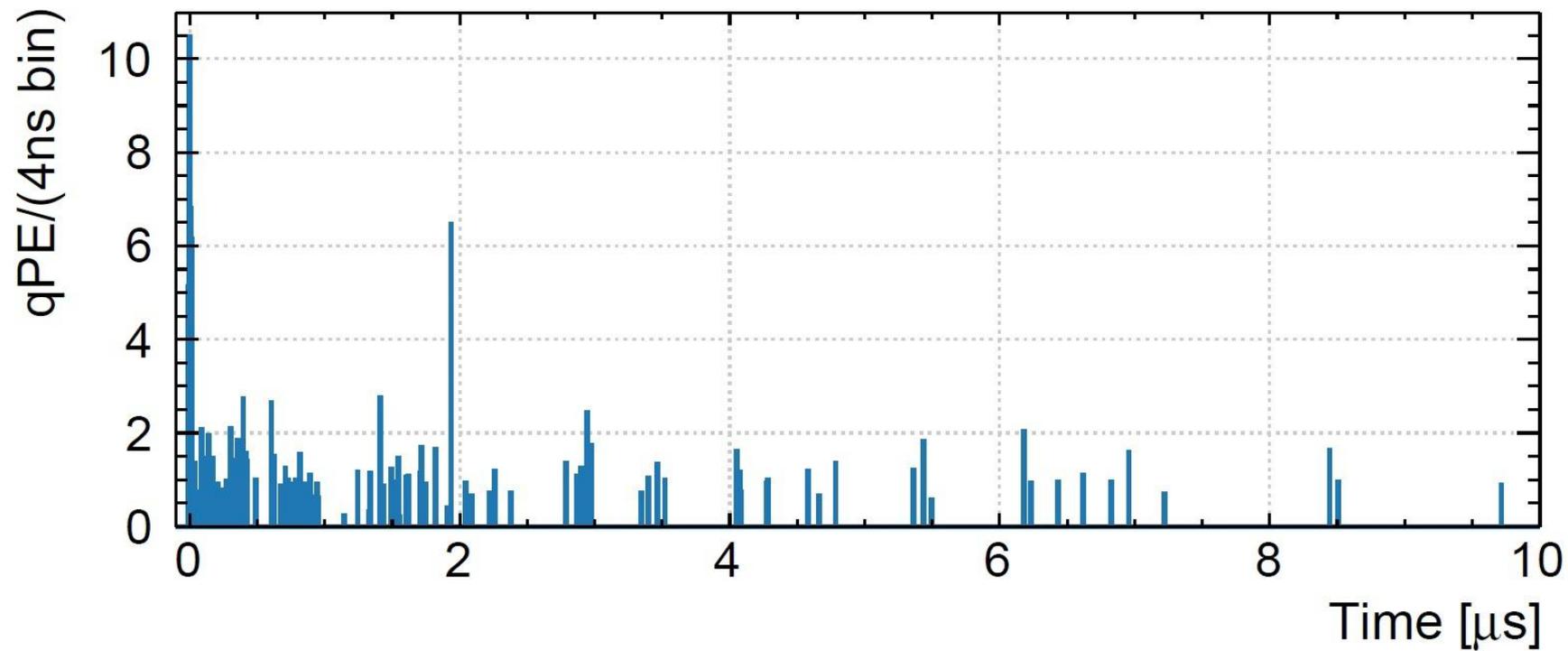


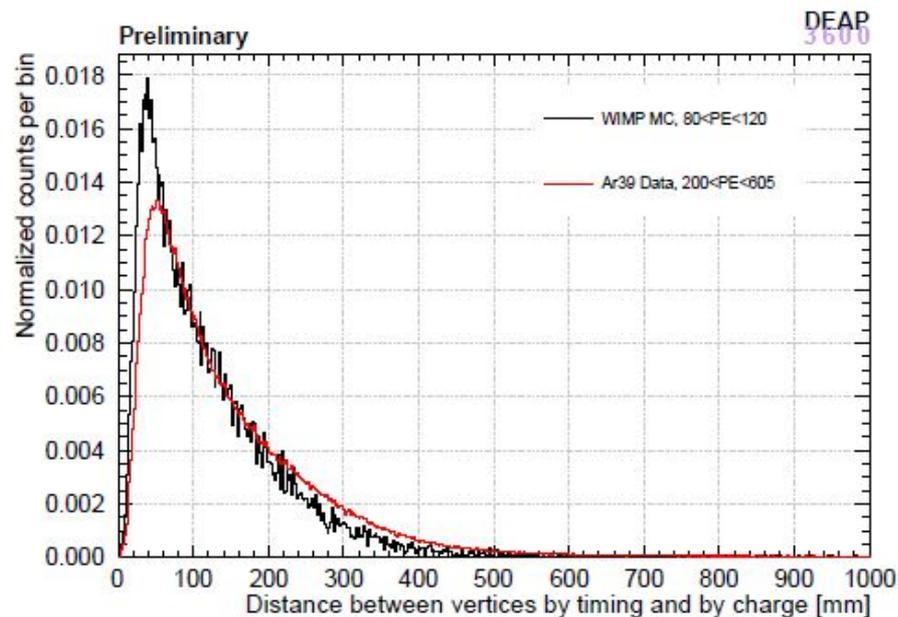
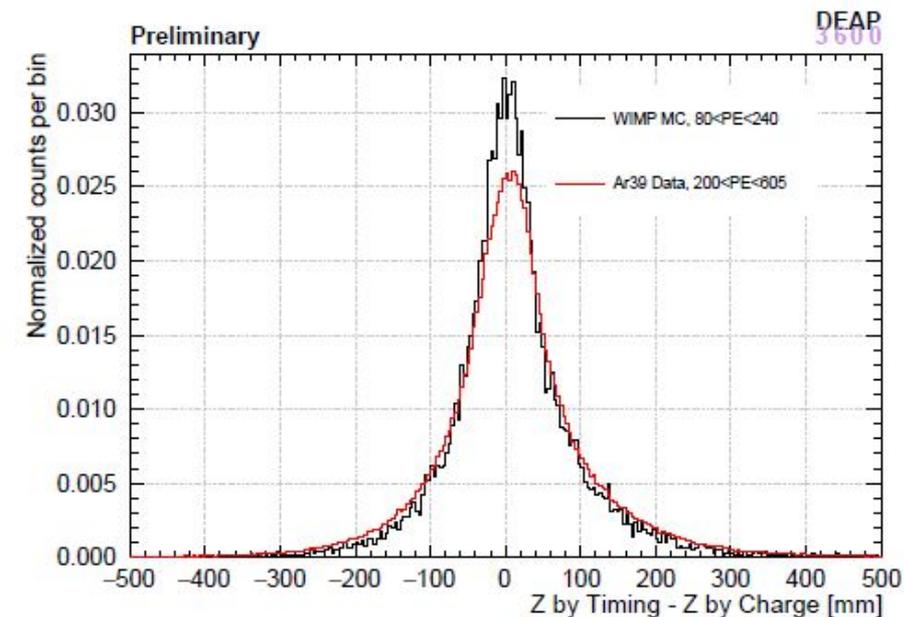
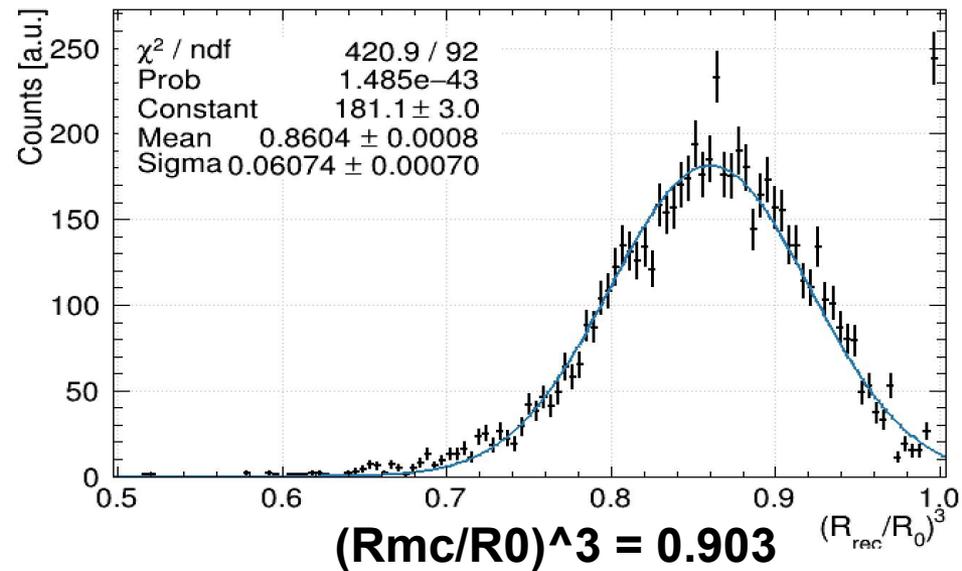
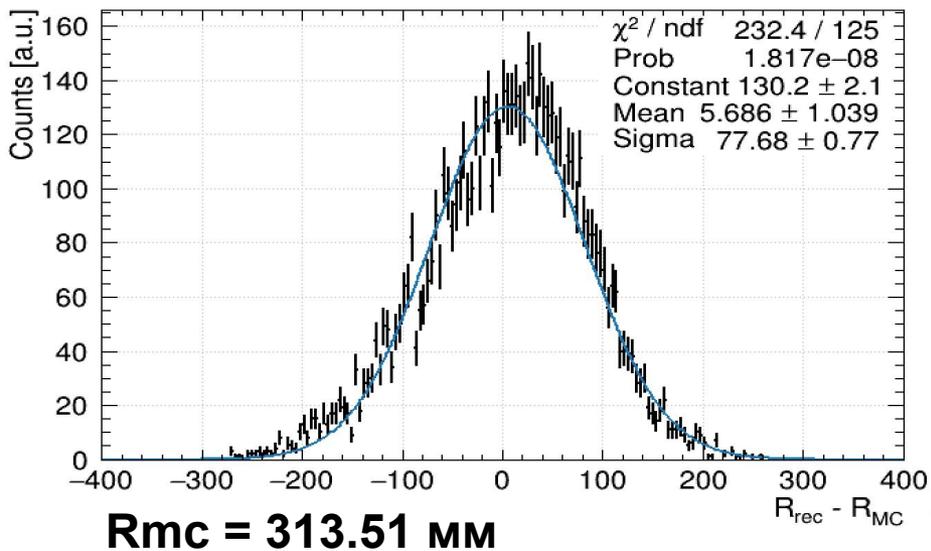
Упрощенный график, показывающий предсказанное сечение рассеяния вимпов на нуклонах как функцию массы вимпов. Асимметричные модели темной материи (голубая область) предсказывают WIMP с массой в несколько ГэВ / с²; общие модели WIMP (темно-синяя область) предсказывают большие массы в сотни ГэВ / с² или более. Части этих областей параметров были исследованы и исключены текущими экспериментами (красная область). Большая часть ожидаемого пространства параметров может быть исследована в ближайшем будущем, пока не станет актуальным фон когерентного нейтрино-ядерного рассеяния (желтая область), мешающий возможным сигналам темной материи.

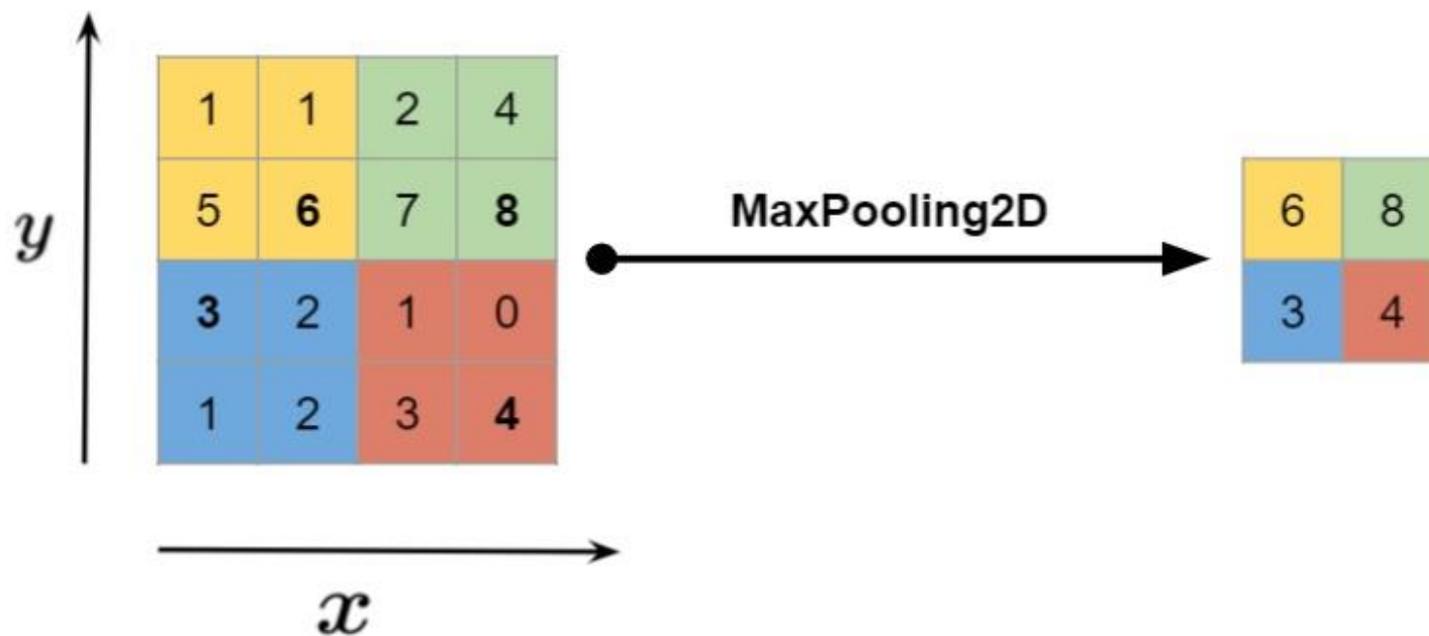
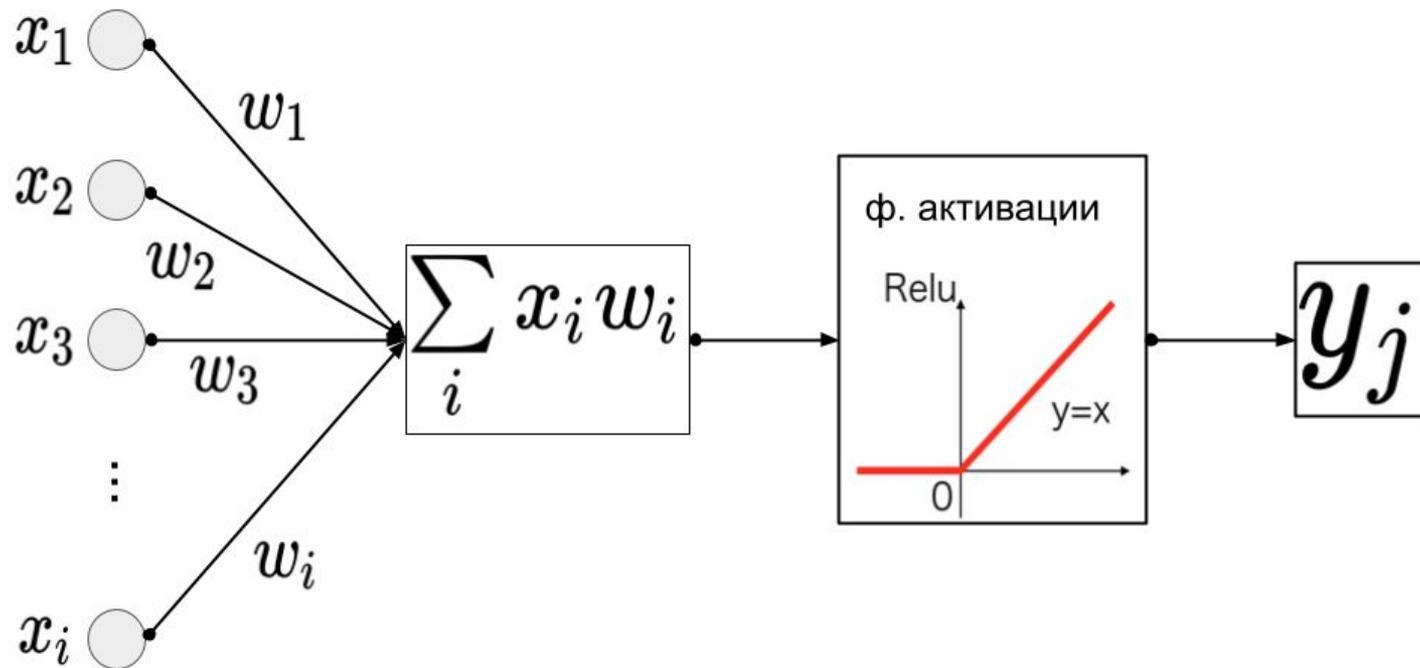


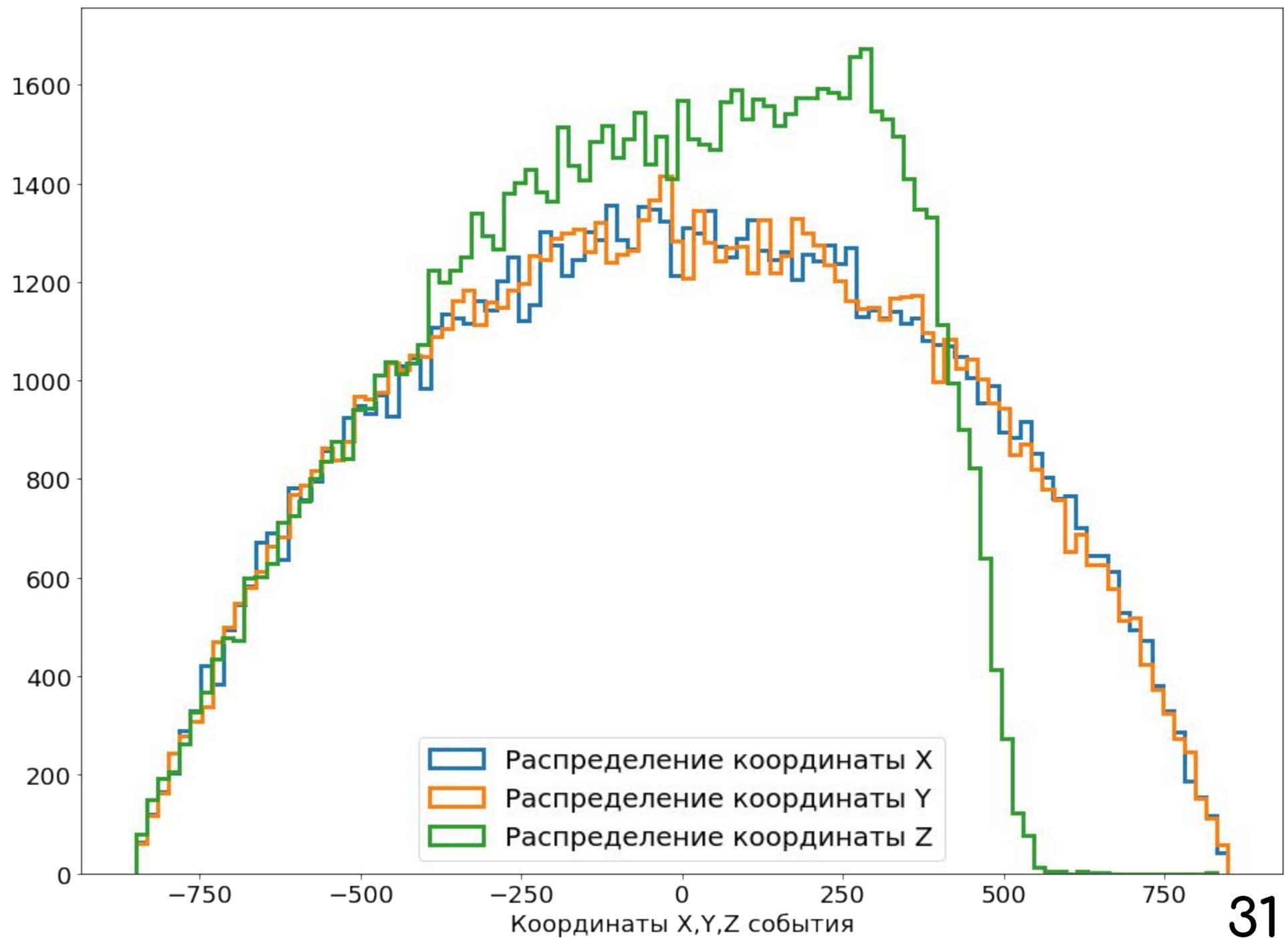
T=75.7°
P=-8.0°
G=4.0°

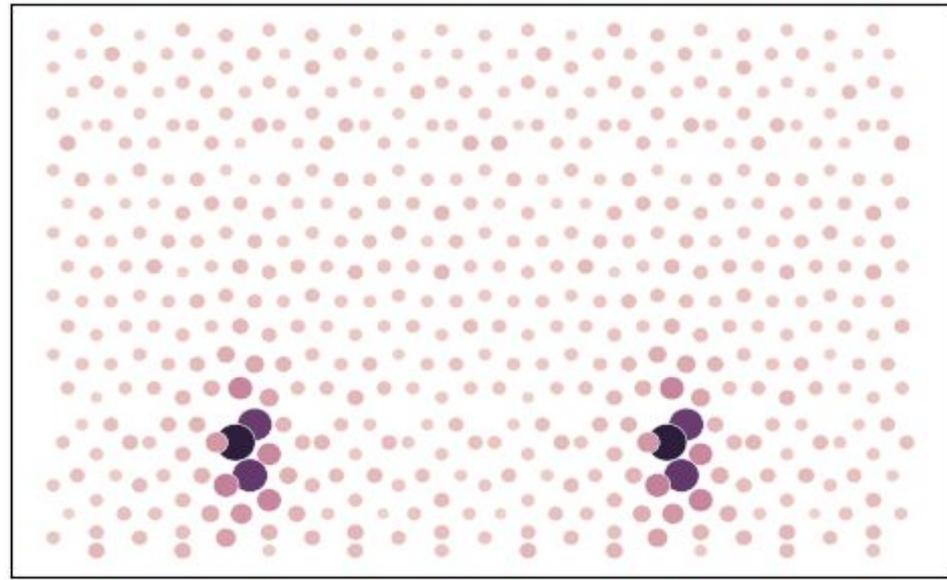
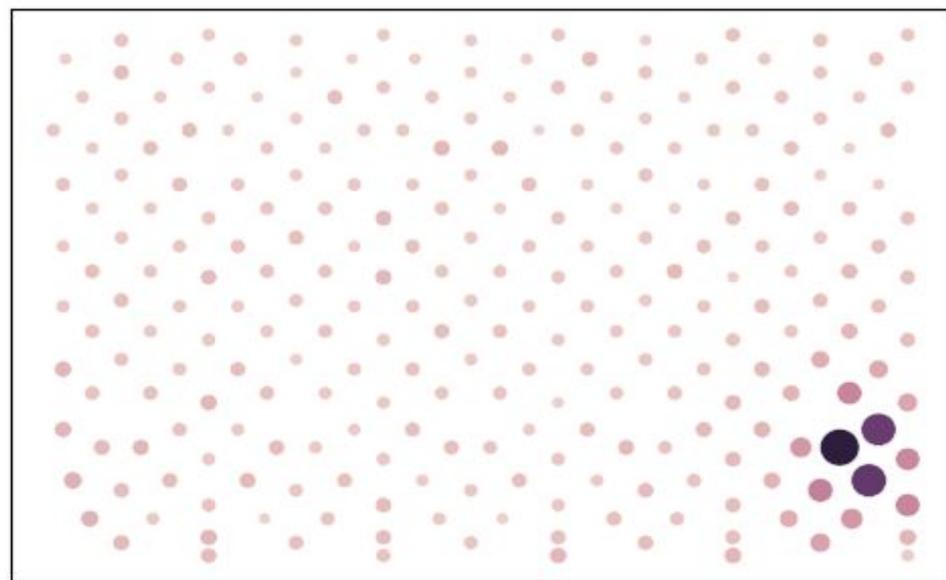


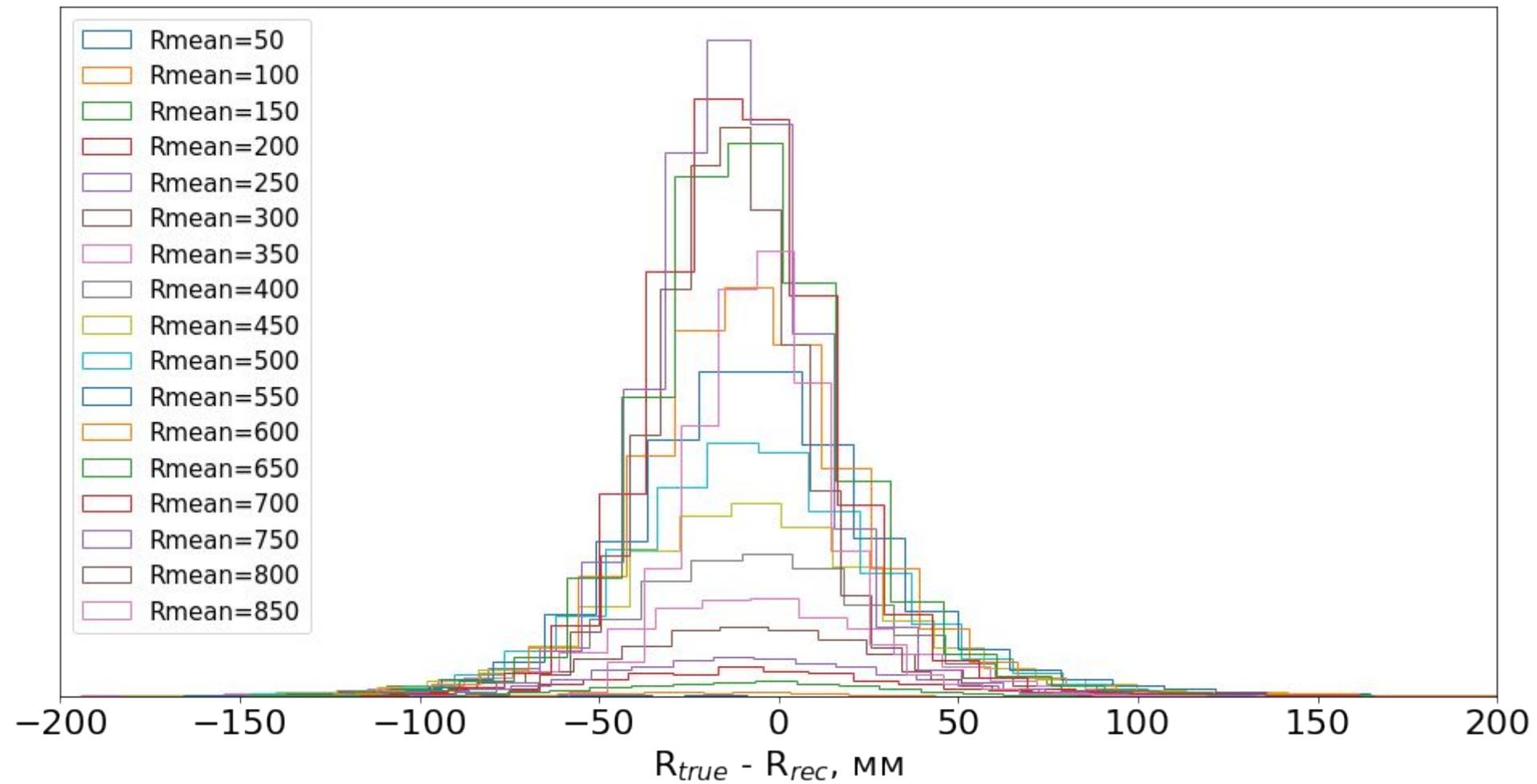




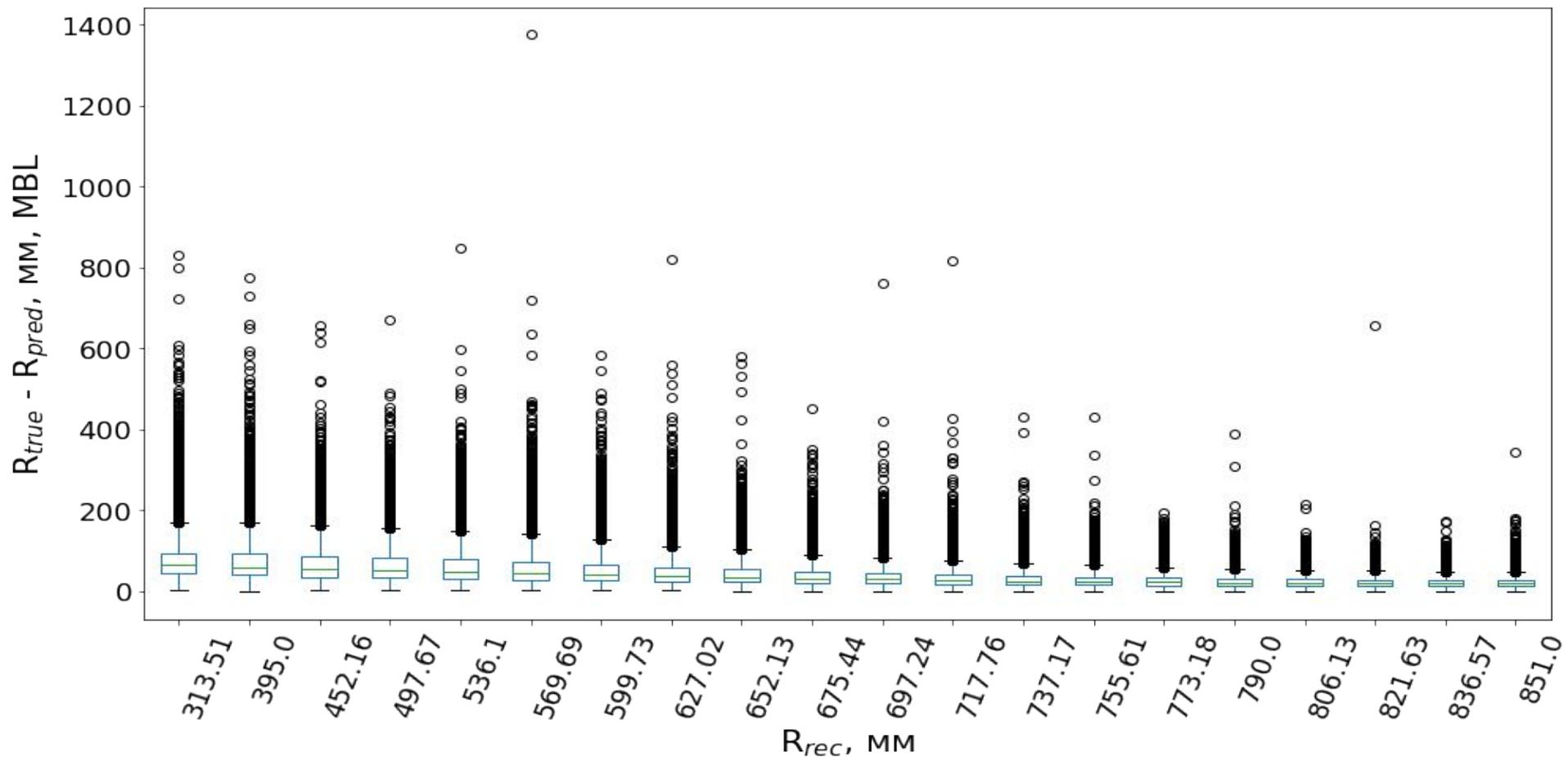




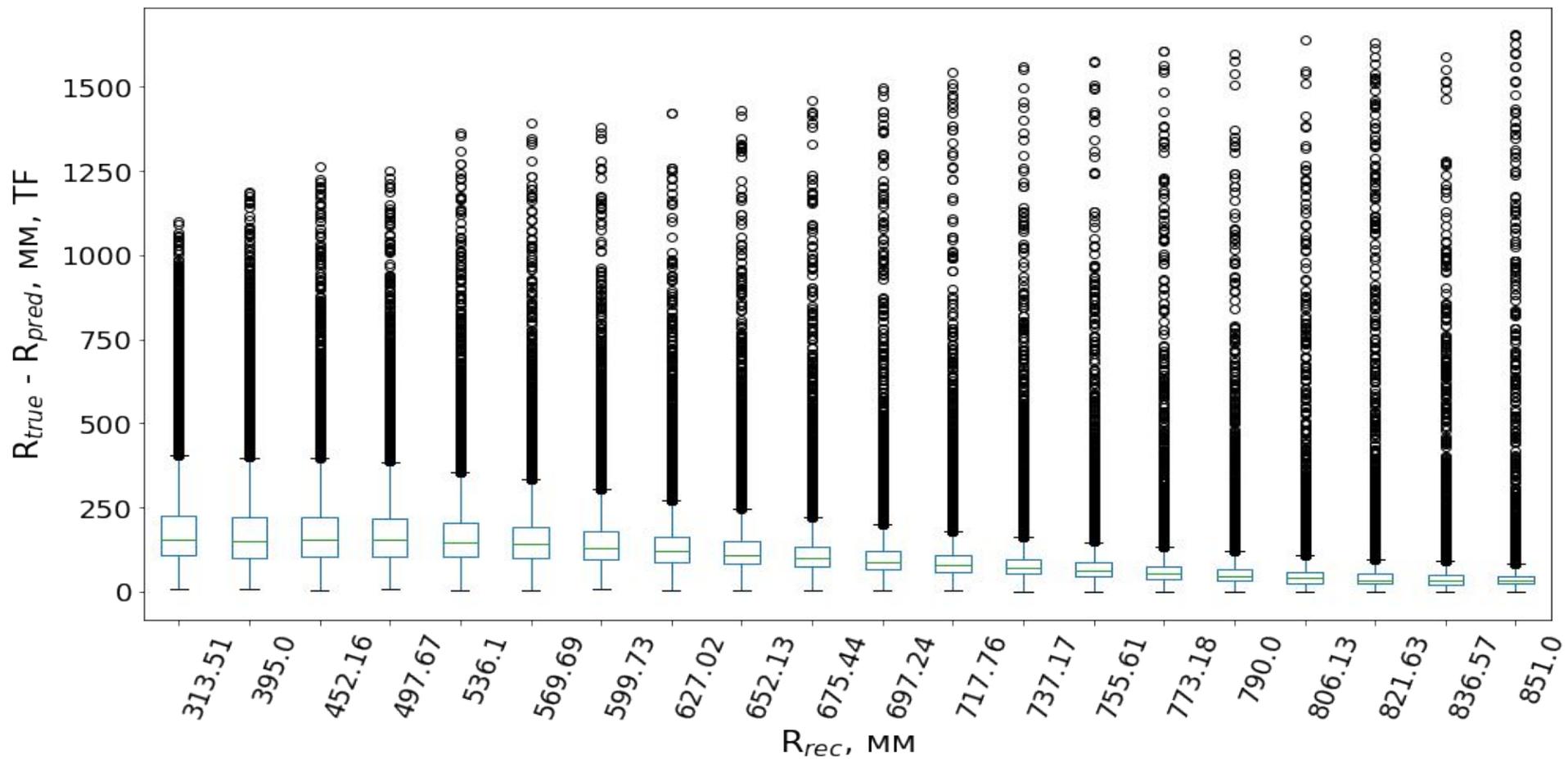




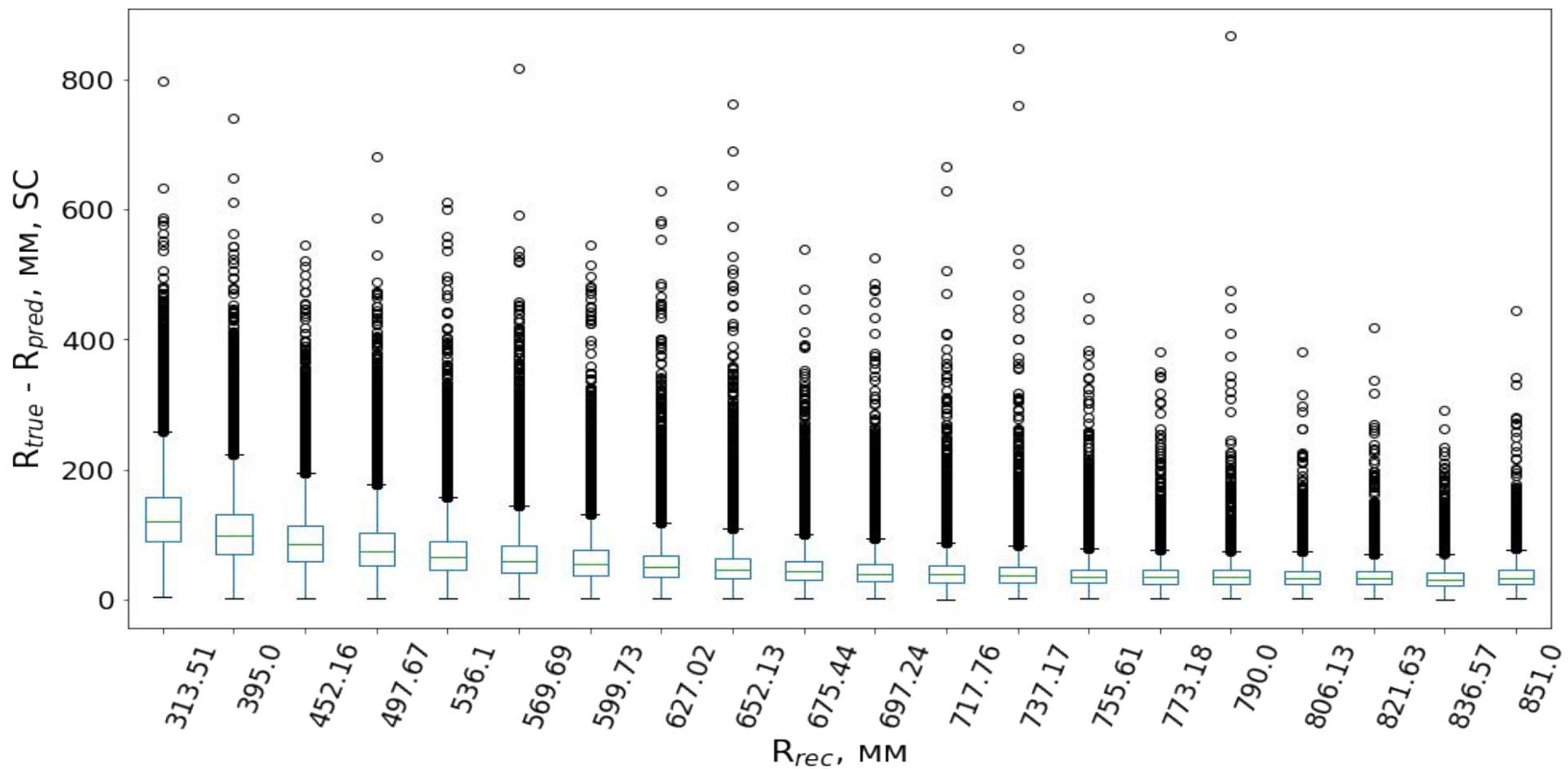
BOXPLOT MBLIKELIHOOD



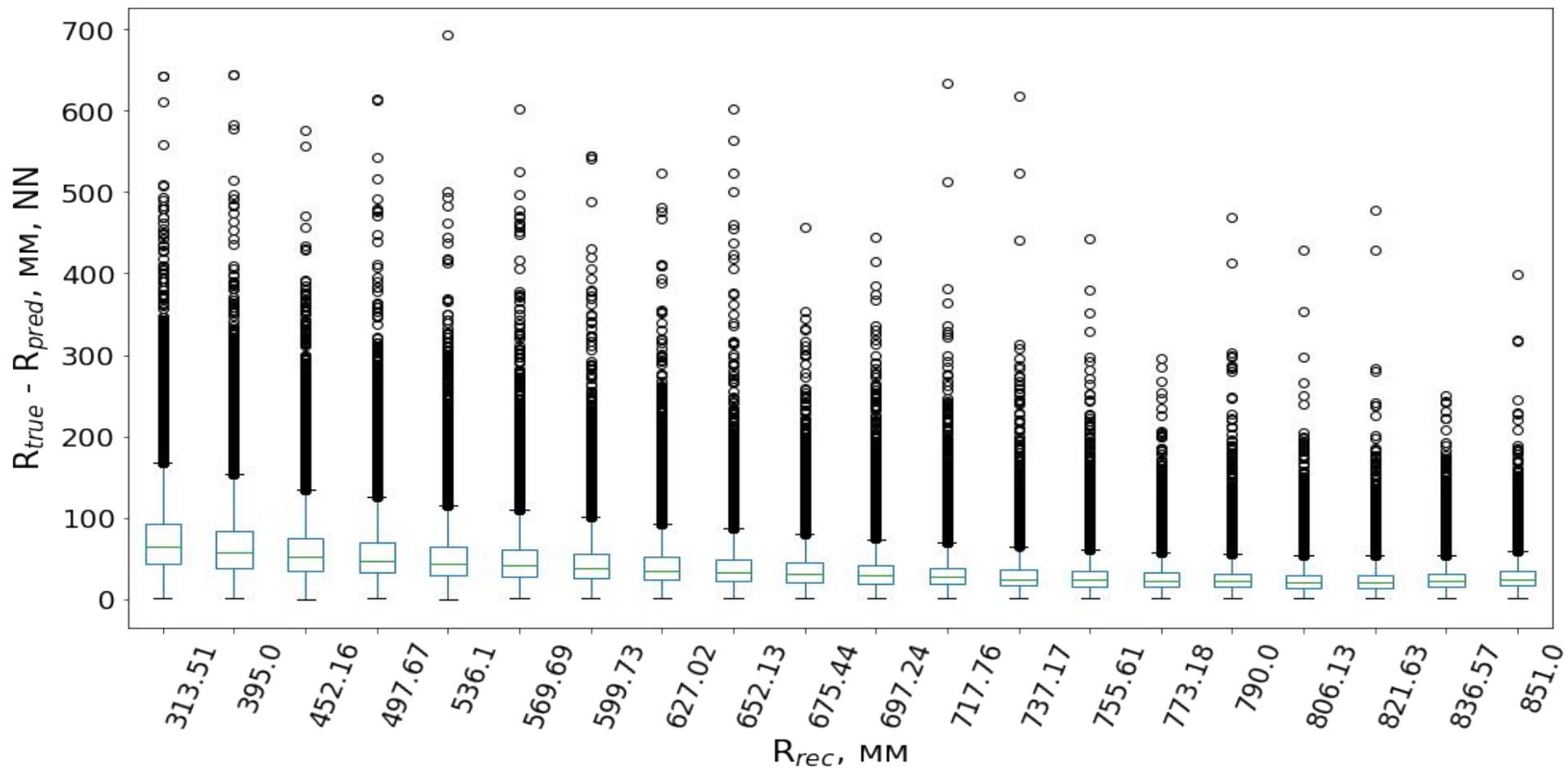
BOXPLOT TIMEFIT2



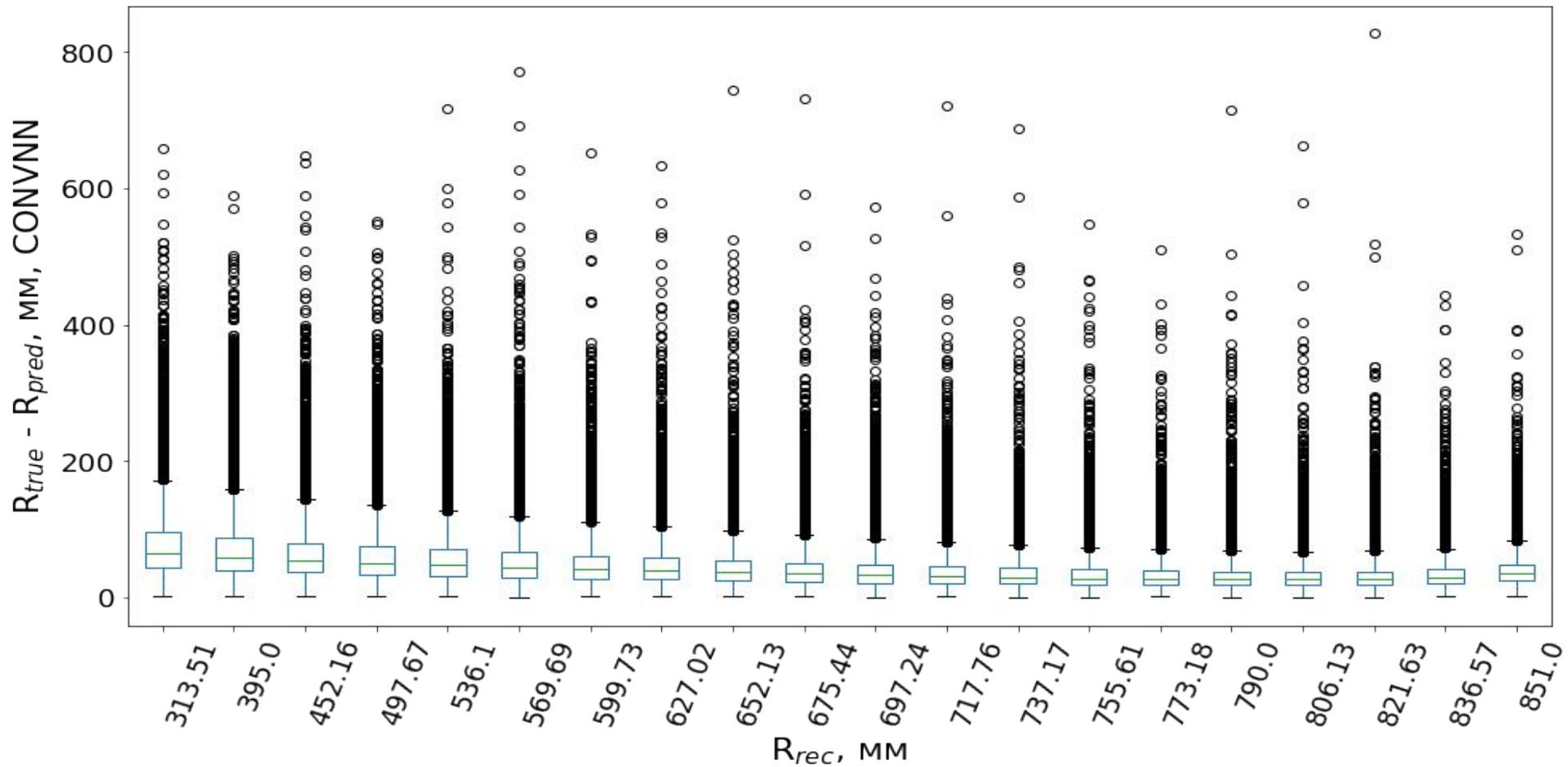
BOXPLOT SHORTCUT

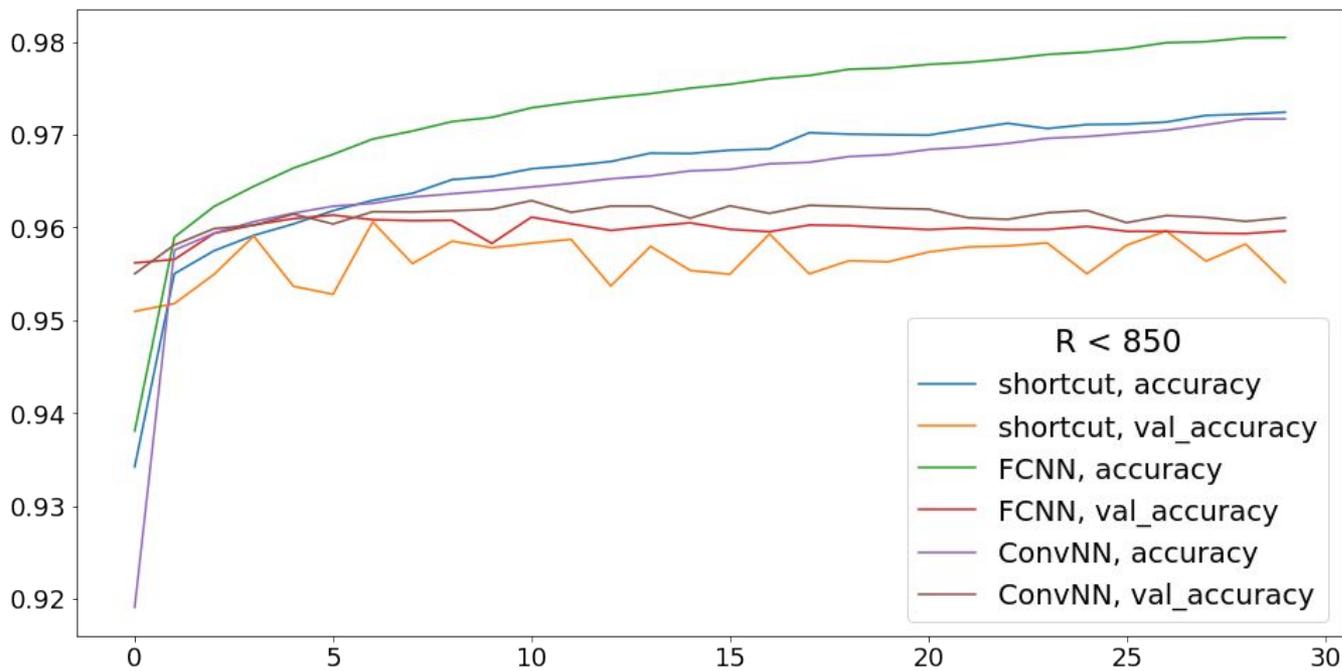


BOXPLOT FCNN

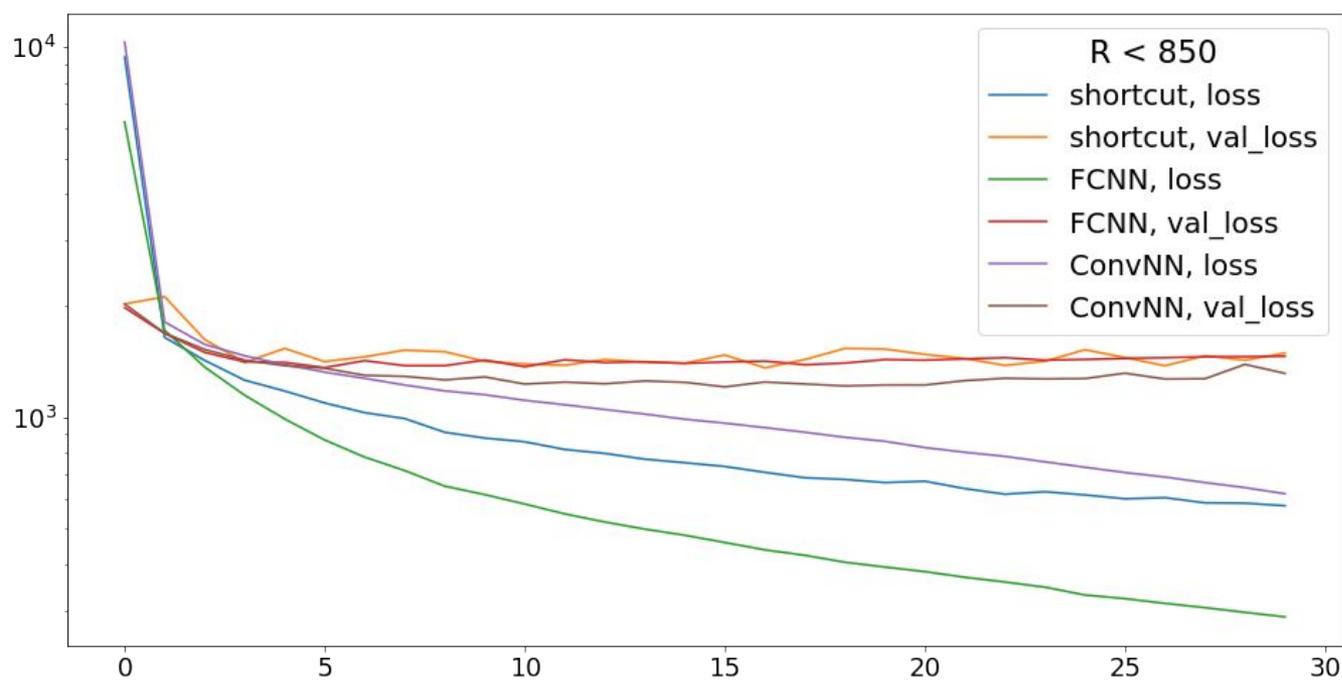


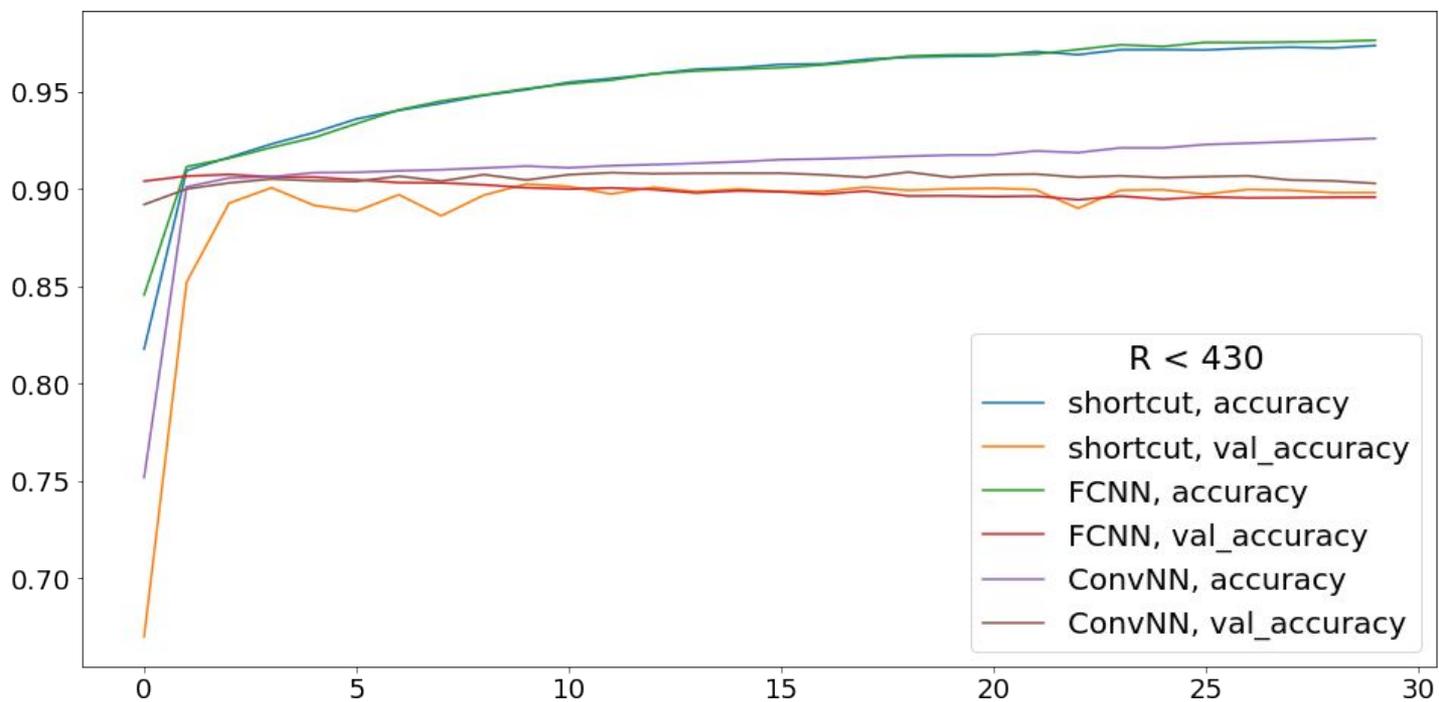
BOXPLOT CONVNN



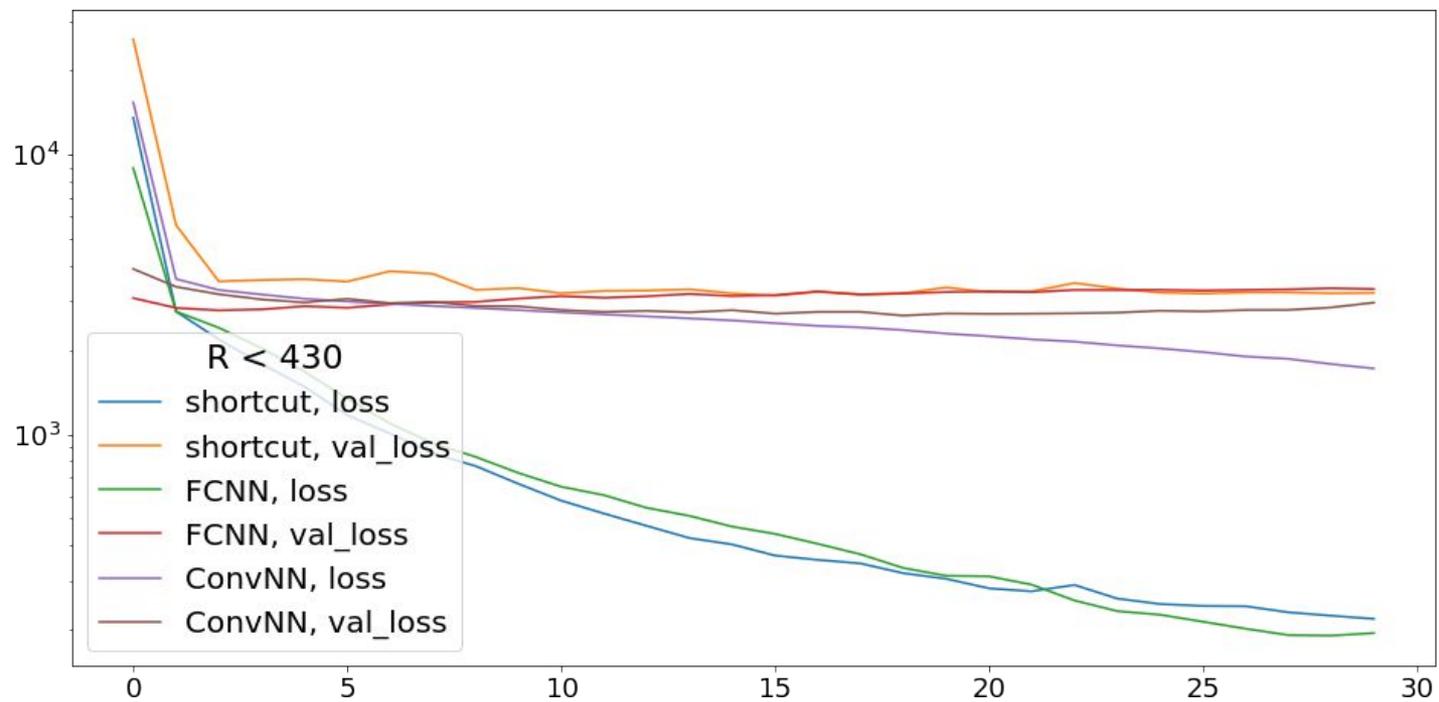


R < 850



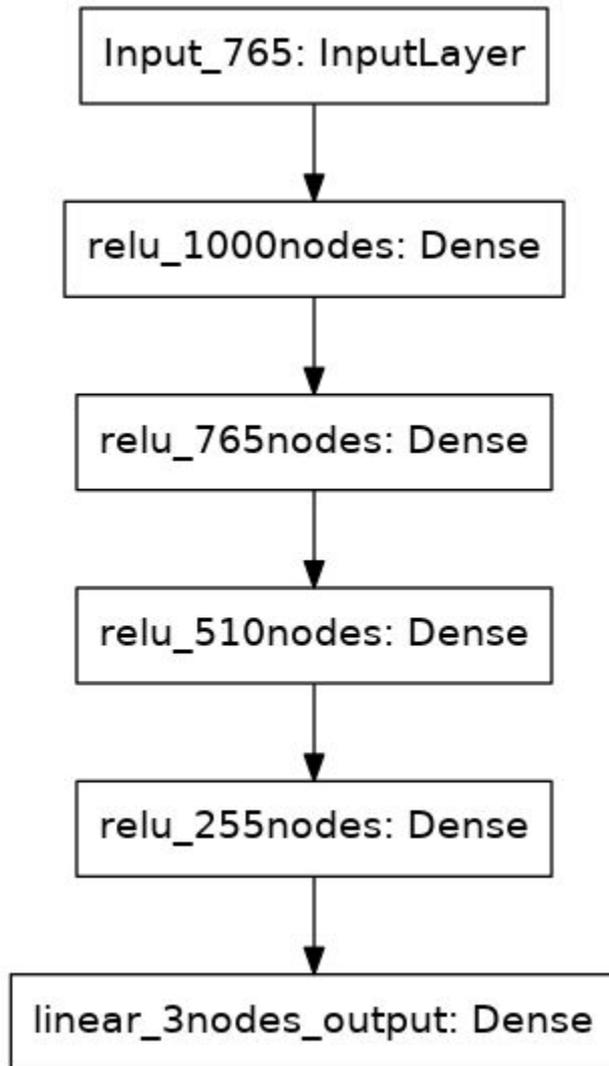


R < 430



40

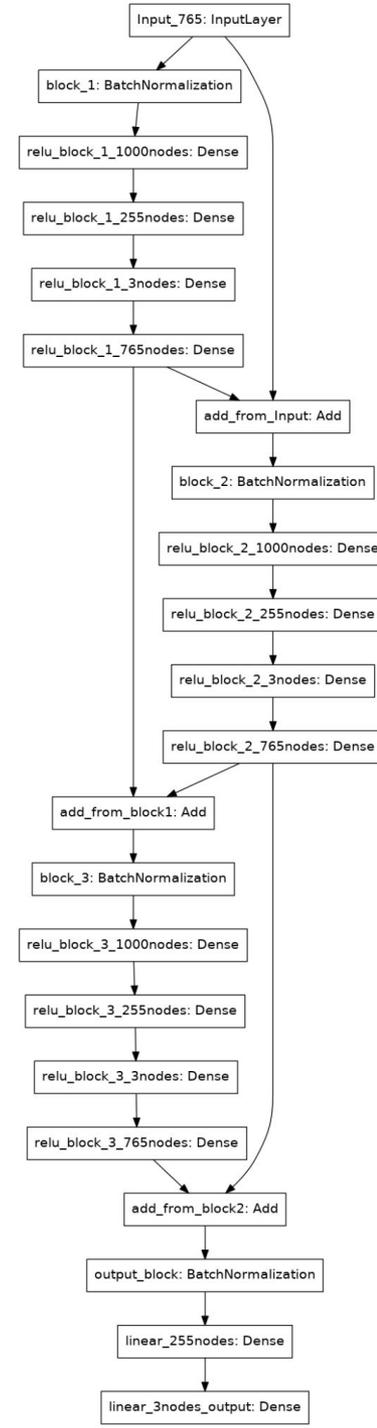
FCNN

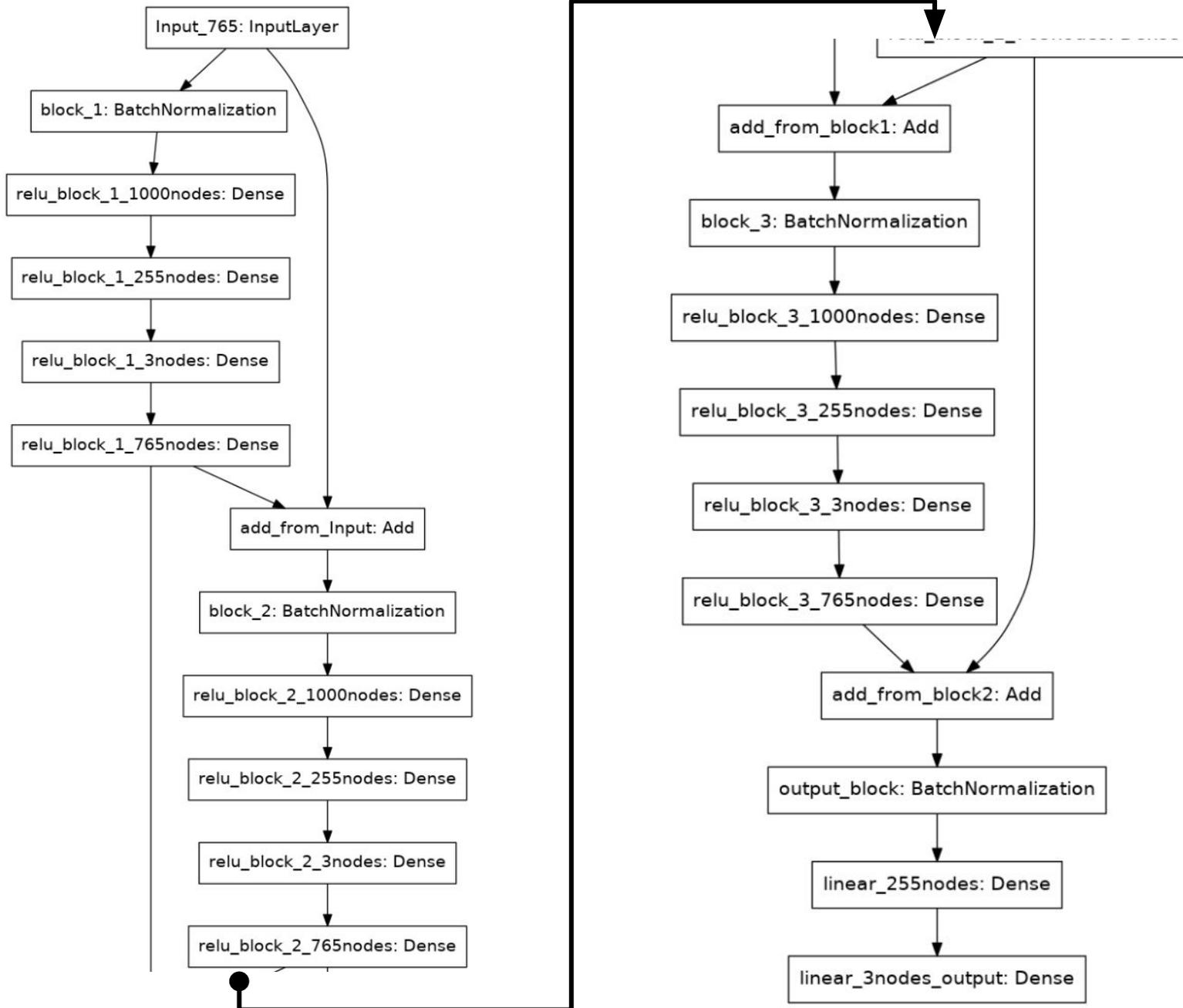


Layer (type)	Output Shape
Input_765 (InputLayer)	[(None, 765)]
1000nodes (Dense)	(None, 1000)
765nodes (Dense)	(None, 765)
510nodes (Dense)	(None, 510)
255nodes (Dense)	(None, 255)
3nodes_output (Dense)	(None, 3)

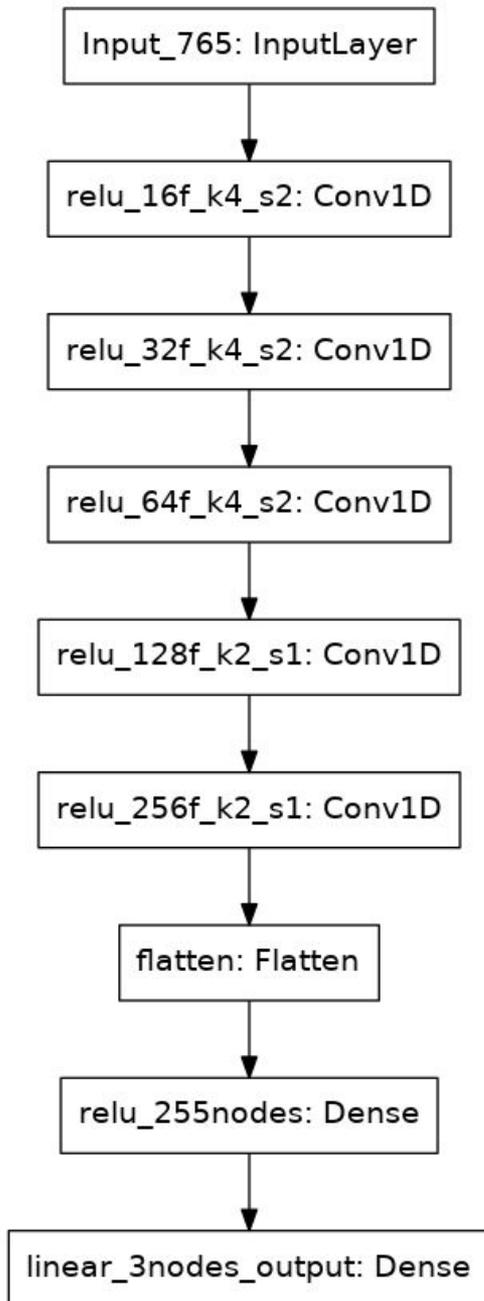
Shortcut

Layer (type)	Output Shape
Input_765 (InputLayer)	[(None, 765)]
block_1 (BatchNormalization)	(None, 765)
block_1_1000nodes (Dense)	(None, 1000)
block_1_255nodes (Dense)	(None, 255)
block_1_3nodes (Dense)	(None, 3)
block_1_765nodes (Dense)	(None, 765)
add_from_Input (Add)	(None, 765)
block_2 (BatchNormalization)	(None, 765)
block_2_1000nodes (Dense)	(None, 1000)
block_2_255nodes (Dense)	(None, 255)
block_2_3nodes (Dense)	(None, 3)
block_2_765nodes (Dense)	(None, 765)
add_from_block1 (Add)	(None, 765)
block_3 (BatchNormalization)	(None, 765)
block_3_1000nodes (Dense)	(None, 1000)
block_3_255nodes (Dense)	(None, 255)
block_3_3nodes (Dense)	(None, 3)
block_3_765nodes (Dense)	(None, 765)
add_from_block2 (Add)	(None, 765)
output_block (BatchNormalization)	(None, 765)
255nodes (Dense)	(None, 255)
3nodes_output (Dense)	(None, 3)





ConvNN



Layer (type)	Output Shape
input_11 (InputLayer)	[(None, 765, 1)]
conv1d (Conv1D)	(None, 383, 16)
conv1d_1 (Conv1D)	(None, 192, 32)
conv1d_2 (Conv1D)	(None, 96, 64)
conv1d_3 (Conv1D)	(None, 96, 128)
conv1d_4 (Conv1D)	(None, 96, 256)
flatten (Flatten)	(None, 24576)
dense_55 (Dense)	(None, 255)
dense_56 (Dense)	(None, 3)

2D ConvNN

Layer (type)	Output Shape
16f_5x5k_relu_same (Conv2D)	(None, 128, 128, 16)
2x2p_2x2s_1 (MaxPooling2D)	(None, 64, 64, 16)
64f_2x2k_relu_same (Conv2D)	(None, 64, 64, 64)
2x2p_2x2s_2 (MaxPooling2D)	(None, 32, 32, 64)
128f_2x2k_relu_same (Conv2D)	(None, 32, 32, 128)
2x2p_2x2s_3 (MaxPooling2D)	(None, 16, 16, 128)
flatten_3 (Flatten)	(None, 32768)
relu_4096nodes (Dense)	(None, 4096)
relu_512nodes (Dense)	(None, 512)
relu_128nodes (Dense)	(None, 128)
relu_32nodes (Dense)	(None, 32)
output (Dense)	(None, 1)

