



# Augmented Neural ODEs

Расширенные нейронные ОДУ

Emilien Dupont, Arnaud Doucet,  
Yee Whye Teh

Подготовили студенты:

Пономарева Ольга  
Теняев Александр  
Алимханов Карим  
Бабаев Минходж  
Солодовников Михаил  
Некрасов Артём



В данной презентации мы покажем, что нейронные обыкновенные дифференциальные уравнения (NODE) изучают представления, сохраняющие топологию входного пространства, и доказываем, что это влечет за собой существование функций, которые нейронные ОДУ не могут представлять.

Для устранения этих ограничений мы вводим расширенные нейронные ОДУ (Augmented NODE), которые являются эмпирически более устойчивыми моделями, лучше обобщают и имеют меньшую вычислительную стоимость, чем нейронные ОДУ.



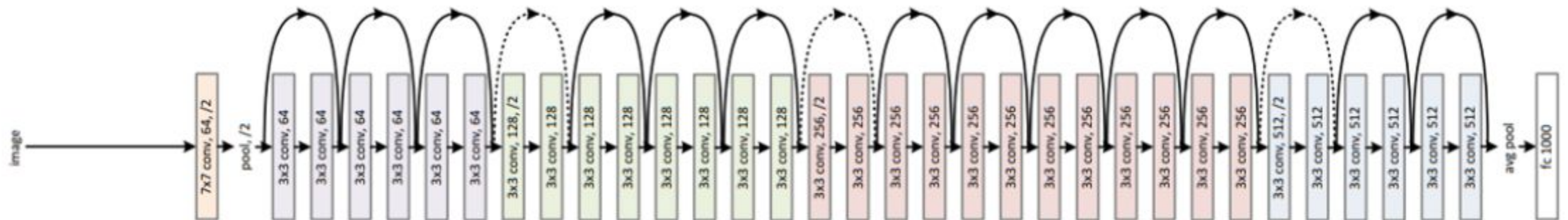
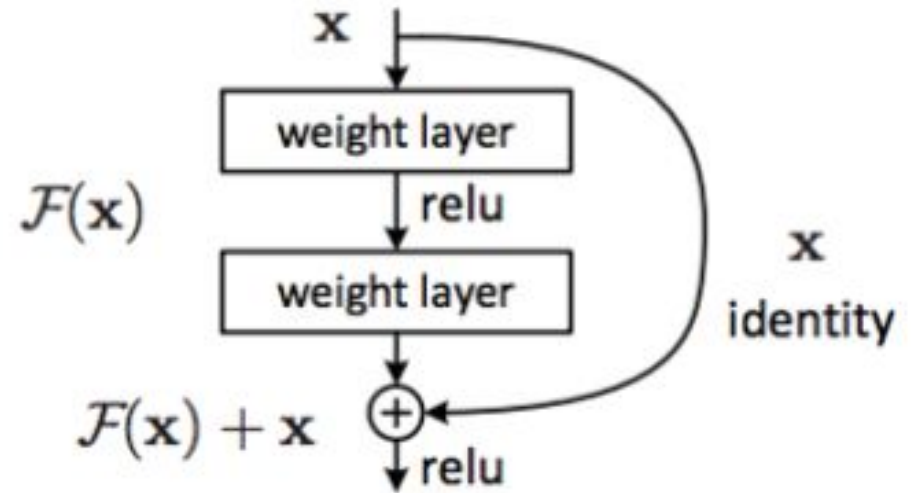
# Немного про ResNets

3

*weight layer* - например, сверточный слой

*relu* - функция активации

*X identity* - skip connection





# Нейронные ОДУ

4

Слой с остаточной связью

$$h_{t+1} = f(h_t, \theta_t) + h_t$$

Возьмем представление  $h$  как функцию от  $t$

$$h(t+1) - h(t) = f(h(t), t, \theta)$$

$$\frac{h(t+1) - h(t)}{(t+1) - t} = f(h(t), t, \theta)$$

Рассмотрим функцию  $h(t)$  как функцию от непрерывного аргумента на некотором отрезке  $[t_0, t_1]$

$$\frac{h(t + \Delta t) - h(t)}{\Delta t} = f(h(t), t, \theta)$$

Устремим число слоёв к бесконечности

$$\frac{\partial h(t)}{\partial t} = f(h(t), t, \theta)$$

Таким образом мы получили ОДУ

$$\frac{\partial h(t)}{\partial t} = f(h(t), t, \theta)$$

Также мы имеем начальное условие

$$h(t_0) = h_{t_0}$$

ОДУ с начальным условием называется задачей Коши.

По теореме о существовании и единственности решения задачи Коши  $h(t)$  представимо в виде

$$h(t) = h(t_0) + \int_{t_0}^t f(h(t), t, \theta) dt$$

Значение  $h(t)$  ищется приблизительно, с заданной точностью, с помощью численного решения дифференциального уравнения

$$h(t) = \text{Solver}(h(t_0), f(\cdot), t_0, t_1, \theta)$$

Мы также определяем поток, связанный с векторным полем  $f(h(t), t)$  ОДУ

$\phi_t : \mathbb{R}^d \rightarrow \mathbb{R}^d$  Поток определяется как скрытое состояние во времени  $t$

$\phi_t(\mathbf{x}) = \mathbf{h}(t)$  при решении ОДУ из начального условия  $h(0) = \mathbf{x}$

Поток измеряет, как состояния ОДУ в заданное время  $t$  зависят от начальных условий  $\mathbf{x}$ . Мы определяем признаки ОДУ как  $\phi(\mathbf{x}) := \phi_T(\mathbf{x})$ , то есть поток в конечное время  $T$ , для которого мы решаем ОДУ.

Архитектура состоит из двух слоев: слоя ОДУ и линейного слоя

Определяем NODE:

$$g : \mathbb{R}^d \rightarrow \mathbb{R} \text{ как } g(\mathbf{x}) = \mathcal{L}(\phi(\mathbf{x}))$$

$$\mathcal{L} : \mathbb{R}^d \rightarrow \mathbb{R} \text{ - линейное отображение}$$

$$\phi : \mathbb{R}^d \rightarrow \mathbb{R}^d \text{ - отображение из данных в признаки}$$

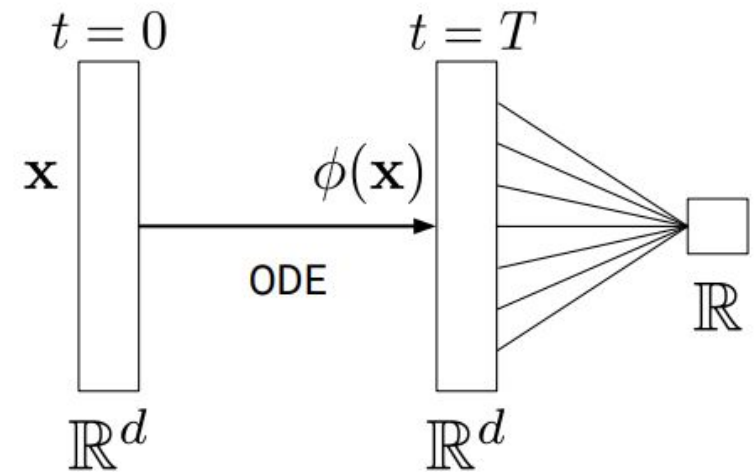
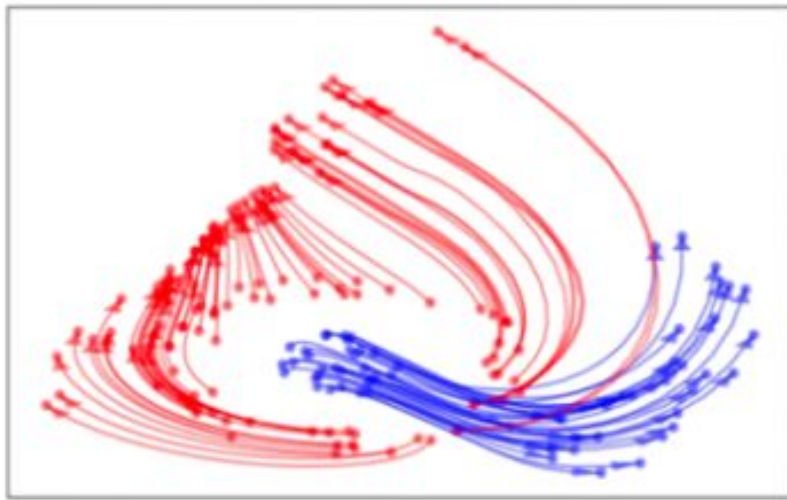


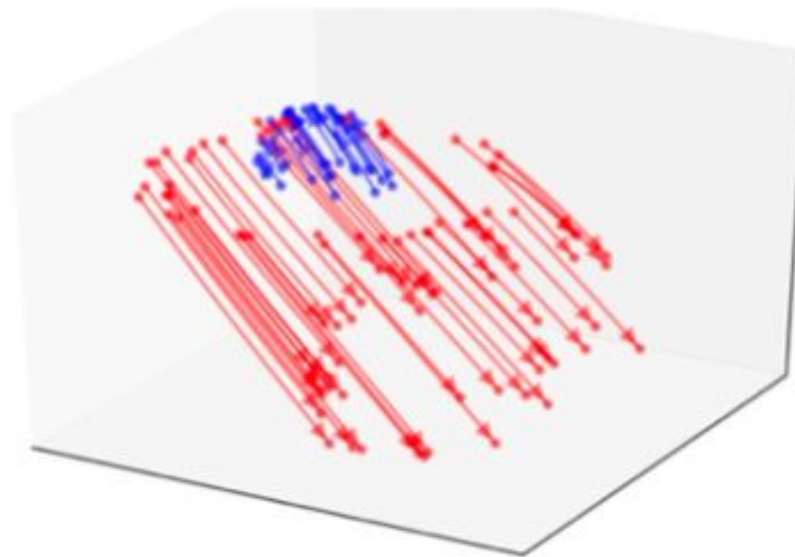
Figure 2: Diagram of Neural ODE architecture.



ANODEs расширяют пространство, в котором решается обыкновенное дифференциальное уравнение, что позволяет модели использовать дополнительные измерения для изучения более сложных функций с использованием более простых потоков



Neural ODE



Augmented Neural ODE

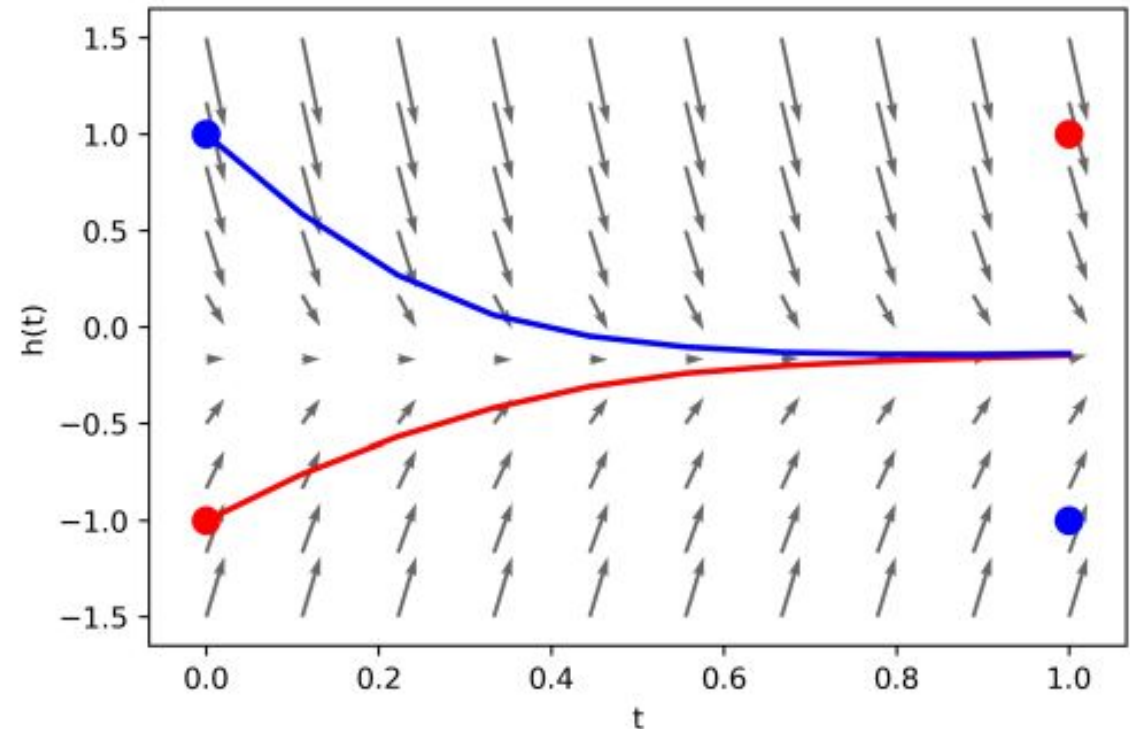
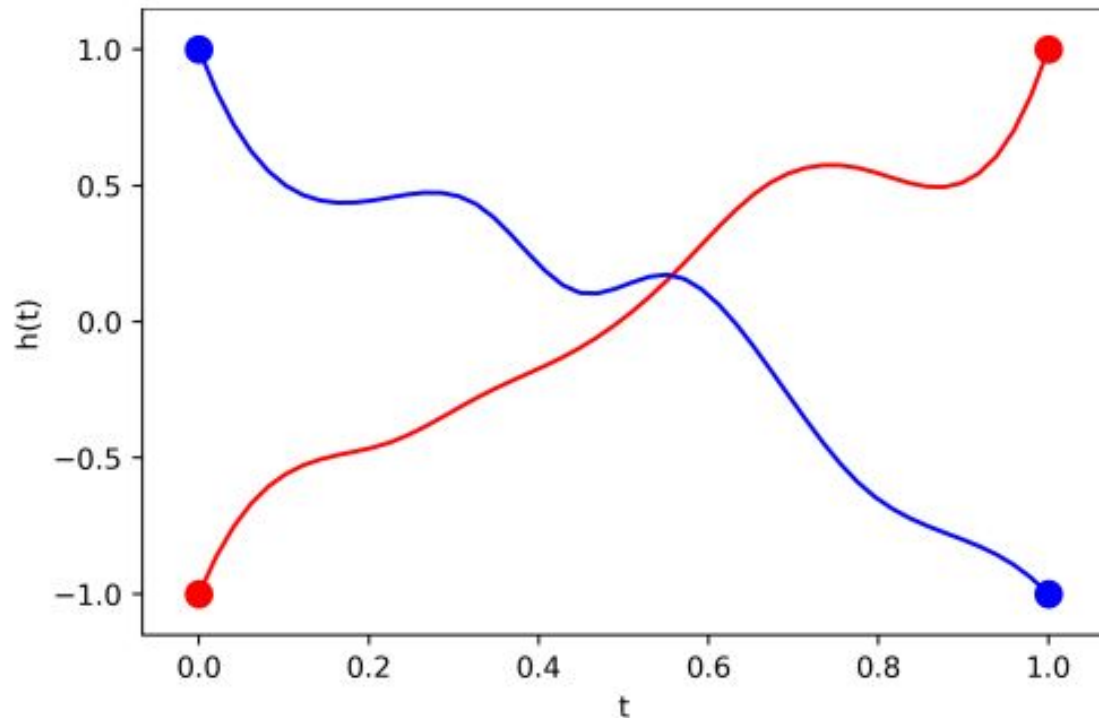




# Функции, которые нейронные ОДУ не могут представить

9

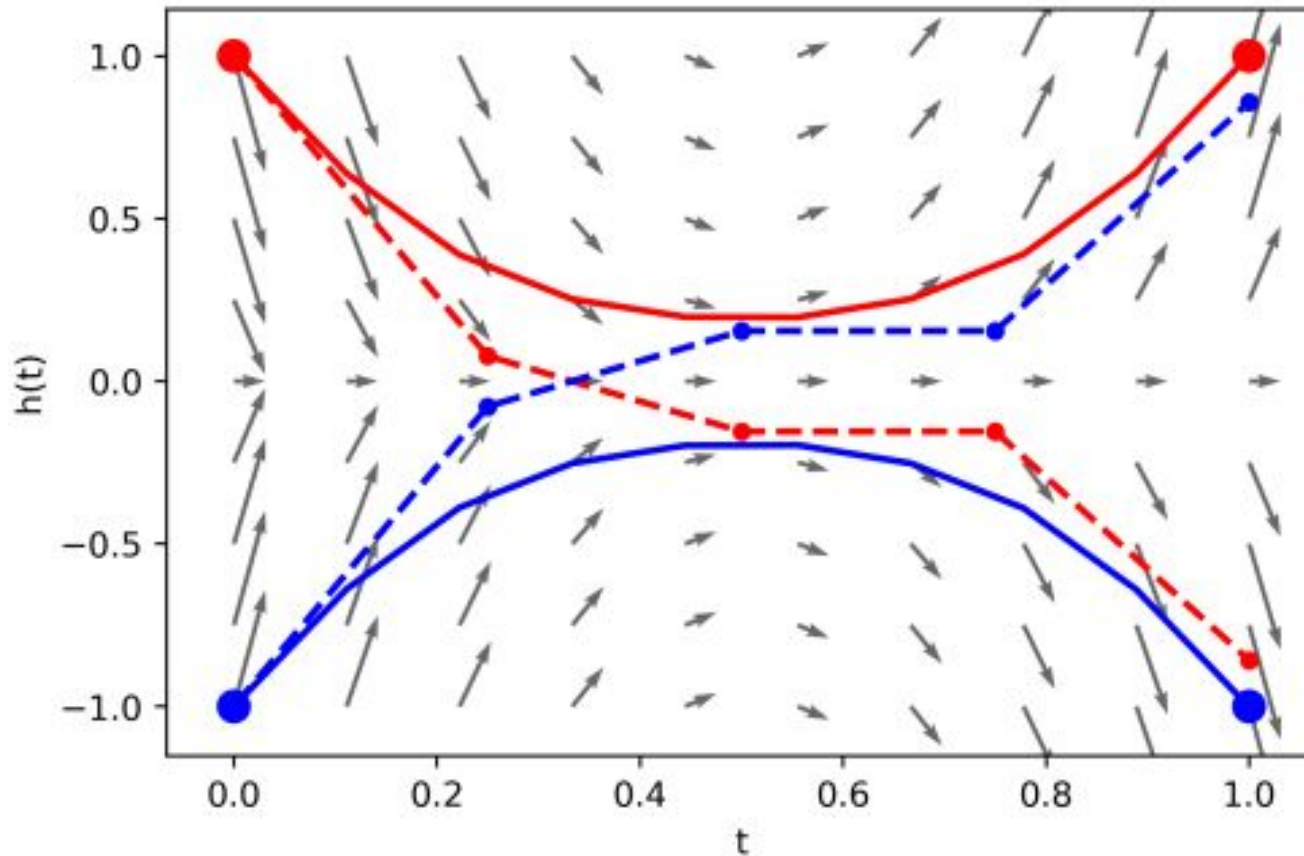
Пусть  $g_{1d} : \mathbb{R} \rightarrow \mathbb{R}$  такая, что  $g_{1d}(-1) = 1$  и  $g_{1d}(1) = -1$





# Функции, которые нейронные ОДУ не могут представить

10



Пунктирные линии - решение ResNet

Сплошные кривые - траектория непрерывного преобразования

Серые стрелки - стрелки векторного поля



# Функции, которые нейронные ОДУ не могут представить

11

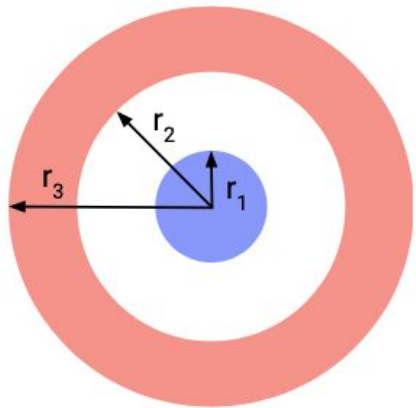
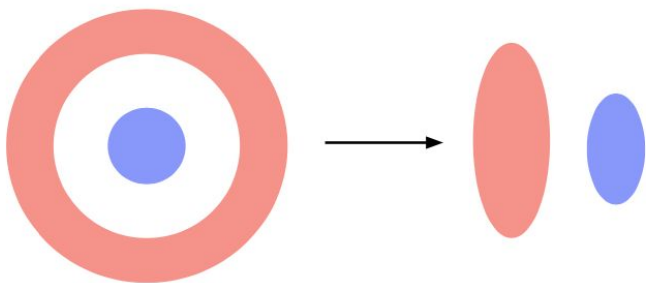


Figure 4: Diagram of  $g(\mathbf{x})$  for  $d = 2$ .

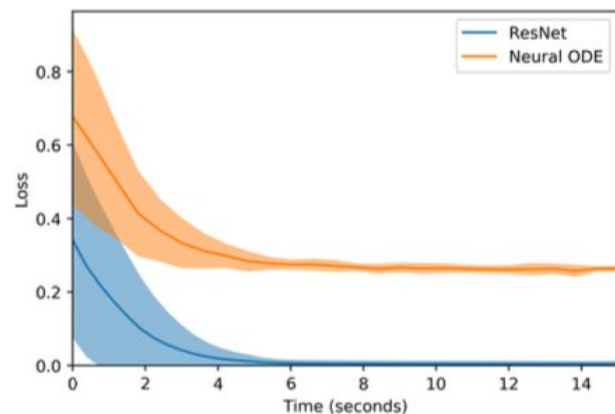
We now introduce classes of functions in arbitrary dimension  $d$  which NODEs cannot represent. Let  $0 < r_1 < r_2 < r_3$  and let  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  be a function such that

$$\begin{cases} g(\mathbf{x}) = -1 & \text{if } \|\mathbf{x}\| \leq r_1 \\ g(\mathbf{x}) = 1 & \text{if } r_2 \leq \|\mathbf{x}\| \leq r_3, \end{cases}$$

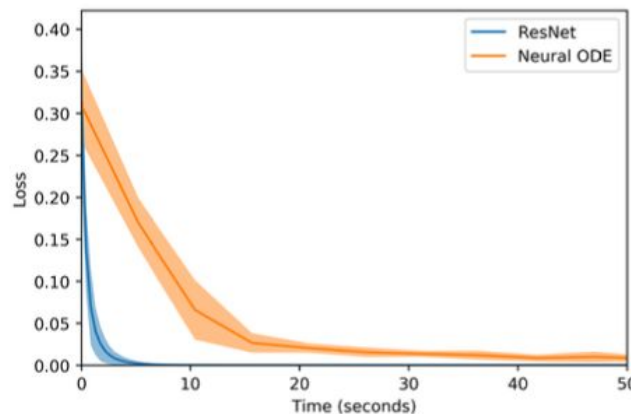
where  $\|\cdot\|$  is the Euclidean norm. An illustration of this function for  $d = 2$  is shown in Fig. 4. The function maps all points inside the blue sphere to  $-1$  and all points in the red annulus to  $1$ .



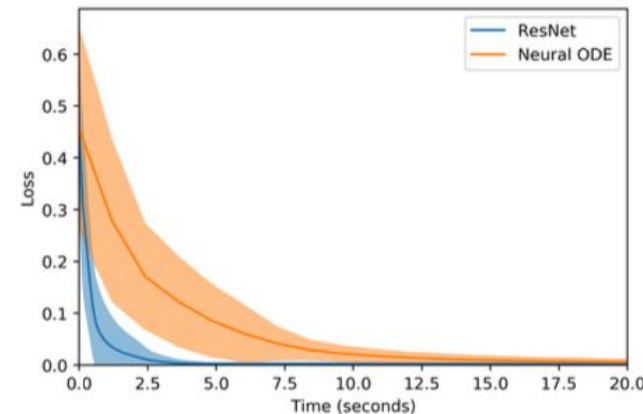
(b)



(a)  $g(\mathbf{x})$  in  $d = 1$



(b)  $g(\mathbf{x})$  in  $d = 2$



(c) Separable function in  $d = 2$

Figure 5: Comparison of training losses of NODEs and ResNets. Compared to ResNets, NODEs struggle to fit  $g(\mathbf{x})$  both in  $d = 1$  and  $d = 2$ . The difference between ResNets and NODEs is less pronounced for the separable function.

# Вычислительные затраты и количество функциональных оценок

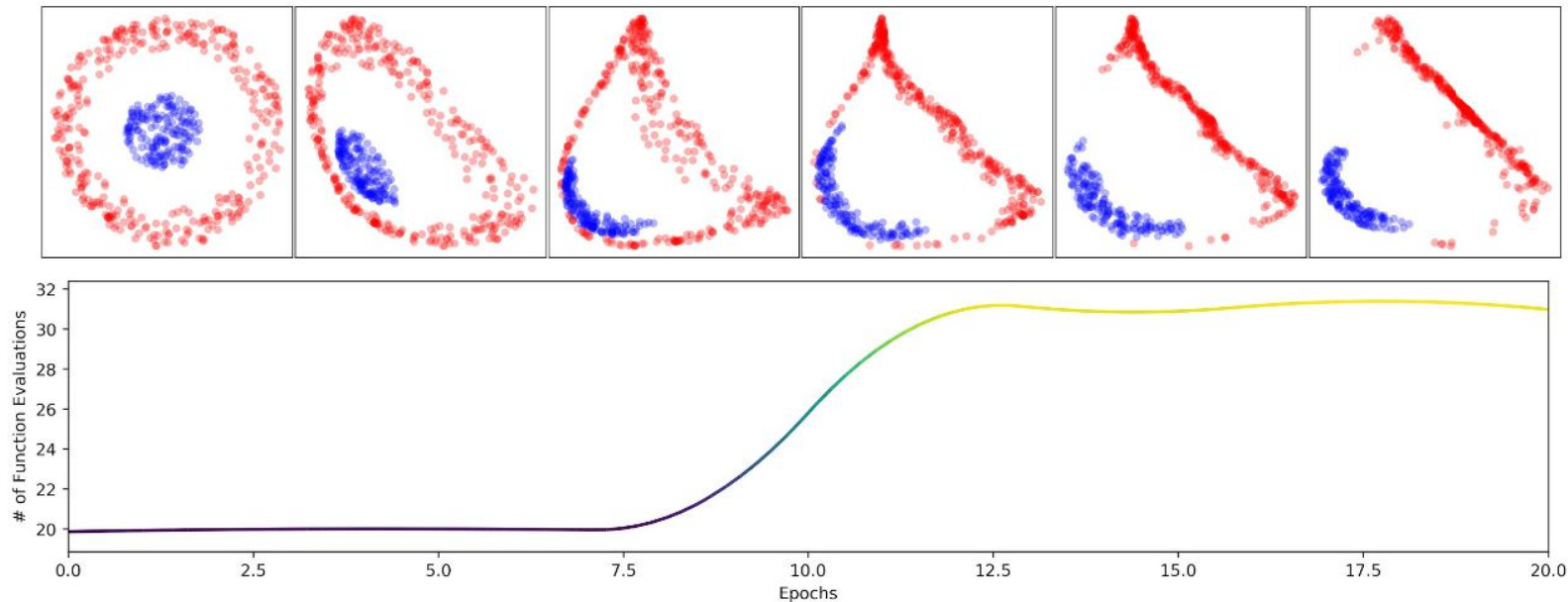


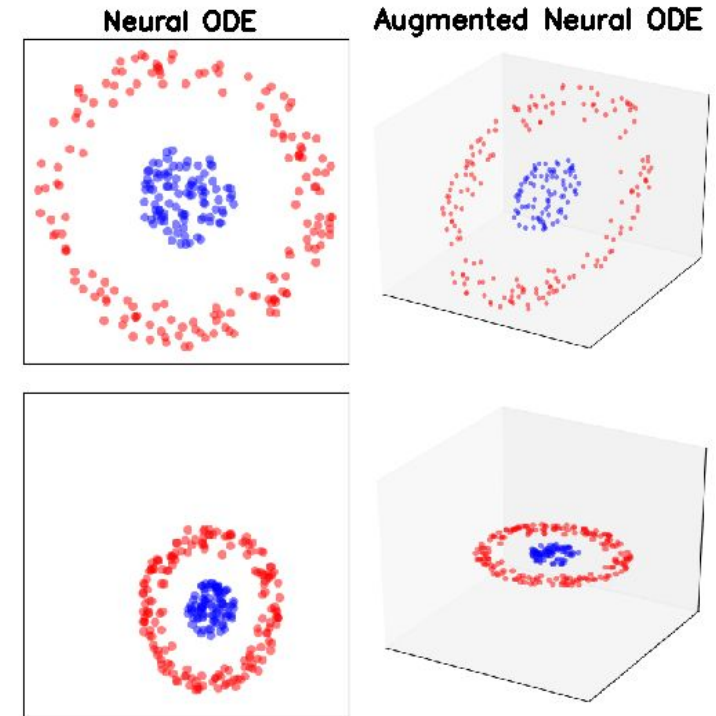
Figure 6: Evolution of the feature space as training progresses and the corresponding number of function evaluations required to solve the ODE. As the ODE needs to break apart the annulus, the number of function evaluations increases.



Идея:  $\mathbb{R}^d \longrightarrow \mathbb{R}^{d+p}$

$$\frac{d}{dt} \begin{bmatrix} \mathbf{h}(t) \\ \mathbf{a}(t) \end{bmatrix} = \mathbf{f} \left( \begin{bmatrix} \mathbf{h}(t) \\ \mathbf{a}(t) \end{bmatrix}, t \right), \quad \begin{bmatrix} \mathbf{h}(0) \\ \mathbf{a}(0) \end{bmatrix} = \begin{bmatrix} \mathbf{x} \\ \mathbf{0} \end{bmatrix}$$

$\mathbf{a}(t) \in \mathbb{R}^p$  - дополненная составляющая

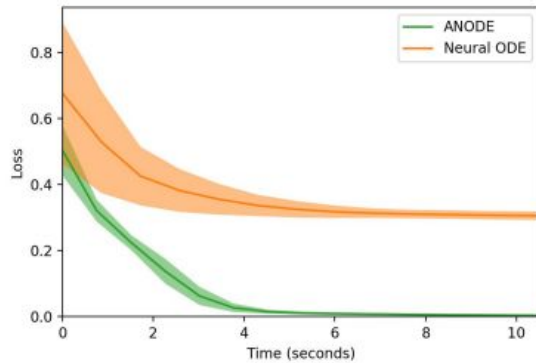




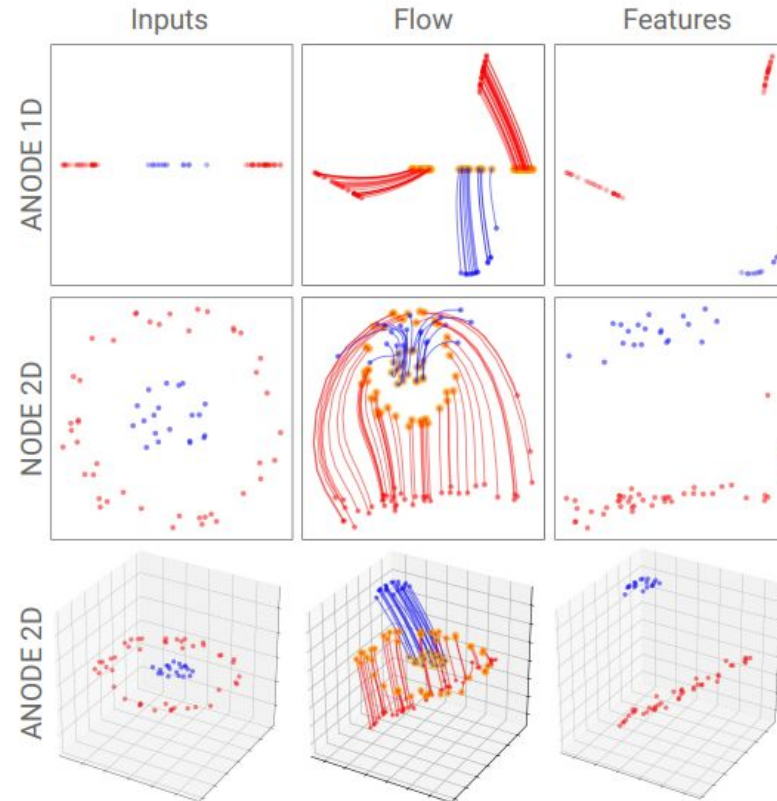
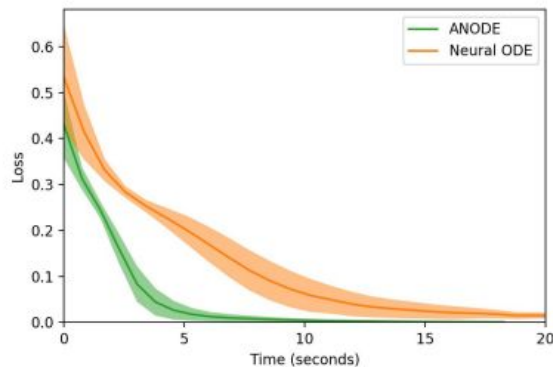
# Augmented Neural ODEs

15

$d = 1$



$d = 2$



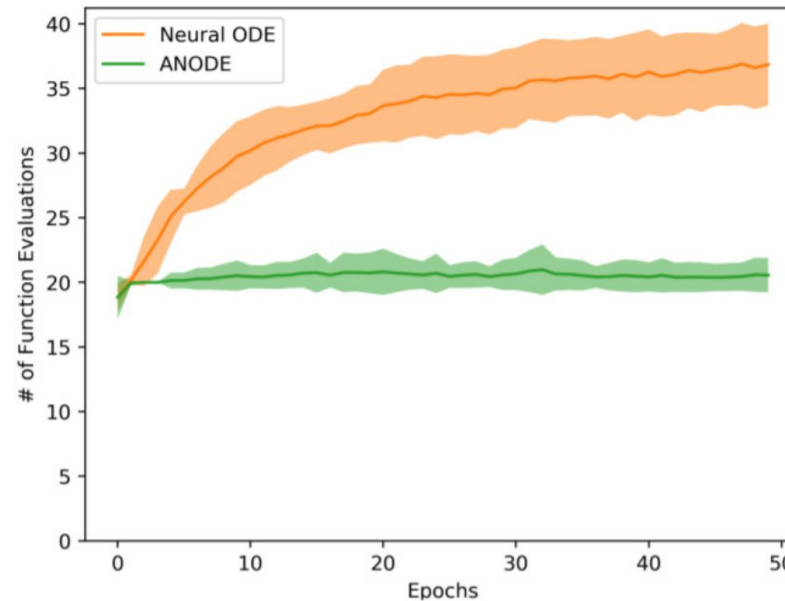
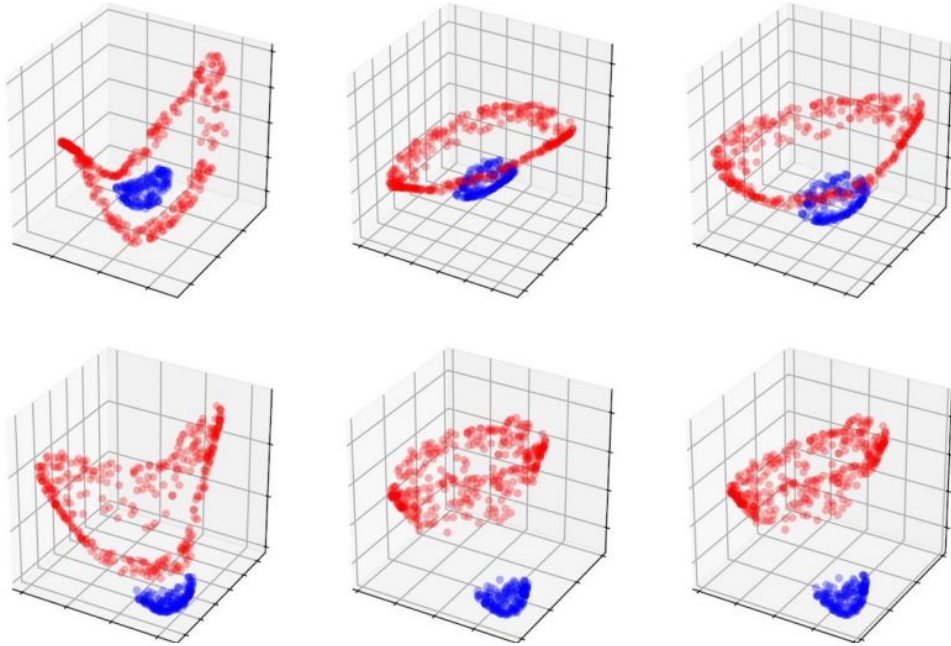
Функция потерь для NODE и ANODE, обученных на  $g(x)$  в  $d = 1$  (вверху) и  $d = 2$  (внизу). ANODE работает быстрее. Справа показаны потоки, выученные НОДами и АНОДами. ANODE обучает простые почти линейные потоки, в то время как NODE обучает сложные (дорогие) потоки





# Augmented Neural ODEs

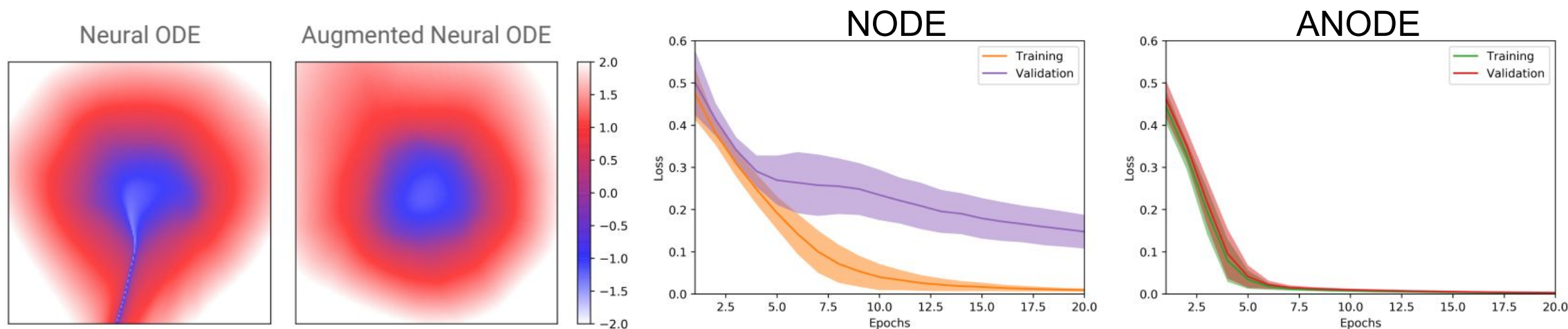
16



По мере обучения ANODE более простым потокам, NFE, требуемый ANODE, практически не увеличивается в процессе обучения, в то время как для NODE он удваивается

Эволюция признаков в процессе обучения для ANODE. Левая верхняя панель показывает пространство признаков для случайно инициализированного ANODE, а внизу справа - признаки после обучения. На правой панели показана эволюция NFE в процессе обучения

Для того чтобы проверить, действительно ли ANODE лучше обобщают → визуализируем, какому значению соответствует каждая точка входного пространства, отображаемая NODE и ANODE (оба оптимизированы примерно до нулевых потерь при обучении).



(Слева) Графики того, как NODE и ANODE отображают точки во входном пространстве на различные выходы (для нулевых потерь при обучении). ANODE обобщает лучше. (Средняя) Потери при обучении и проверке для NODE, (Правая) для ANODE



# Эксперименты с изображениями

18

*input x: image*

$$\mathbb{R}^{c \times h \times w} \rightarrow \mathbb{R}^{(c+p) \times h \times w}$$

$c$  — количество каналов;

$h$  — высота изображения;

$w$  — ширина изображения;

$p$  — дополнительное количество измерений.

★ *MNIST* — это набор данных, состоящий из 70 000 изображений рукописных цифр от 0 до 9. Изображения имеют размер 28×28 пикселей.

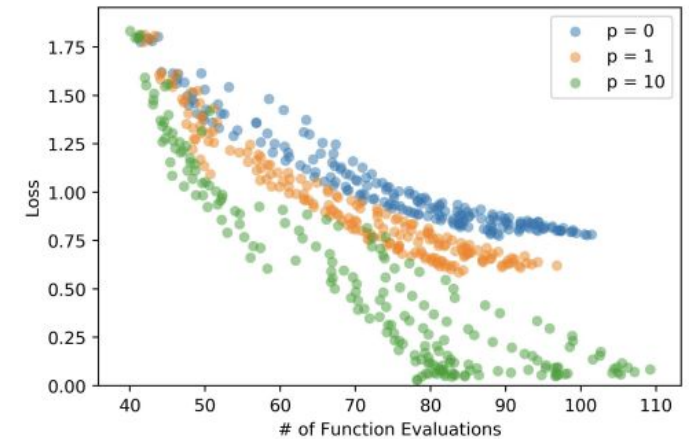
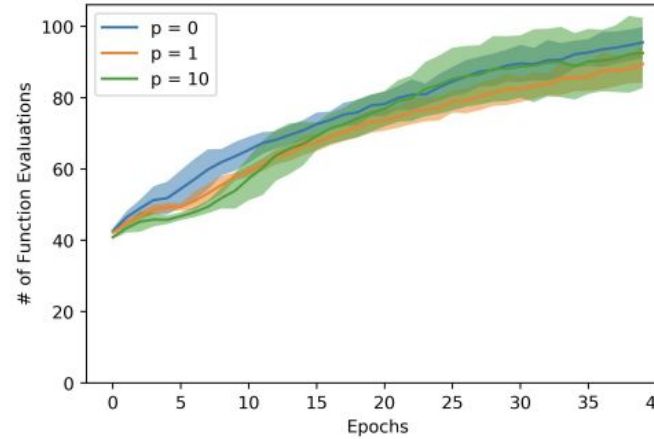
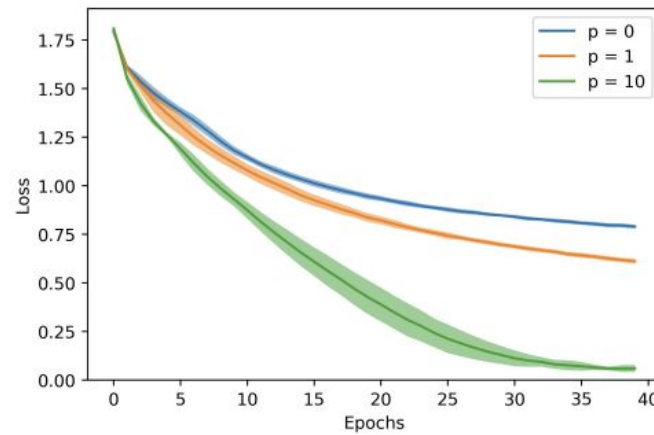
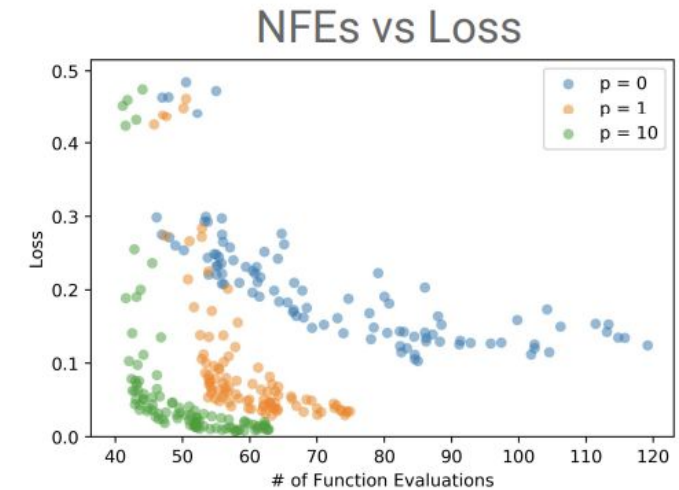
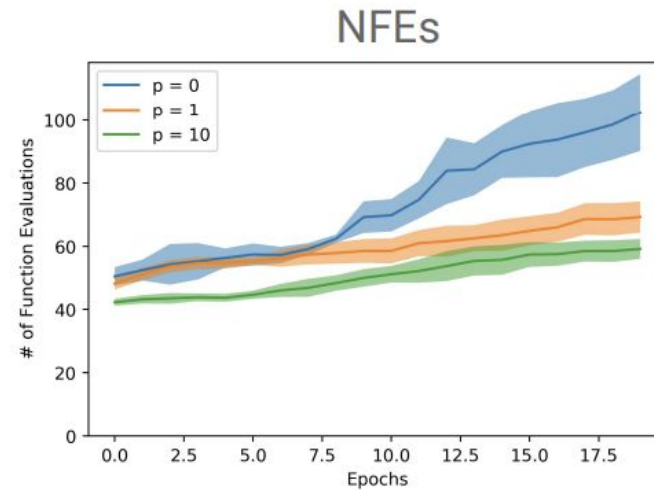
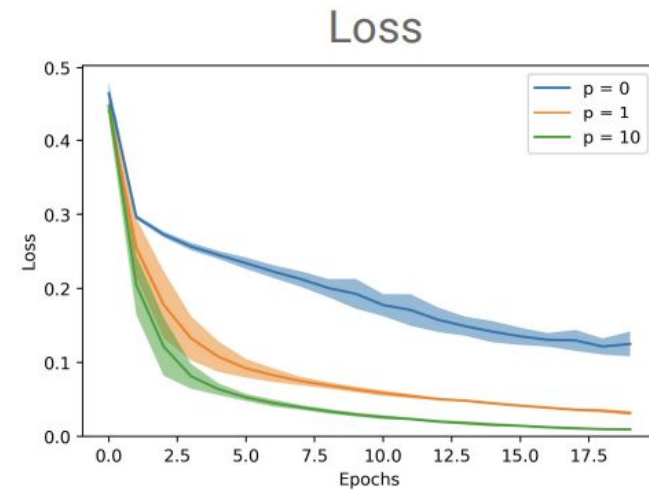
★ *CIFAR10* — это набор данных, состоящий из 60 000 изображений цветных объектов из 10 категорий: самолет, автомобиль, птица, кошка, собака, лошадь, корова, лягушка, корабль, грузовик. Изображения имеют размер 32×32 пикселей.

★ *SVHN* — это набор данных, состоящий из 73 257 изображений домов с номерами домов из Нью — Джерси. Изображения имеют размер 32×32 пикселей.



# Эксперименты с изображениями

19





# Эксперименты с изображениями

## Точность.

20

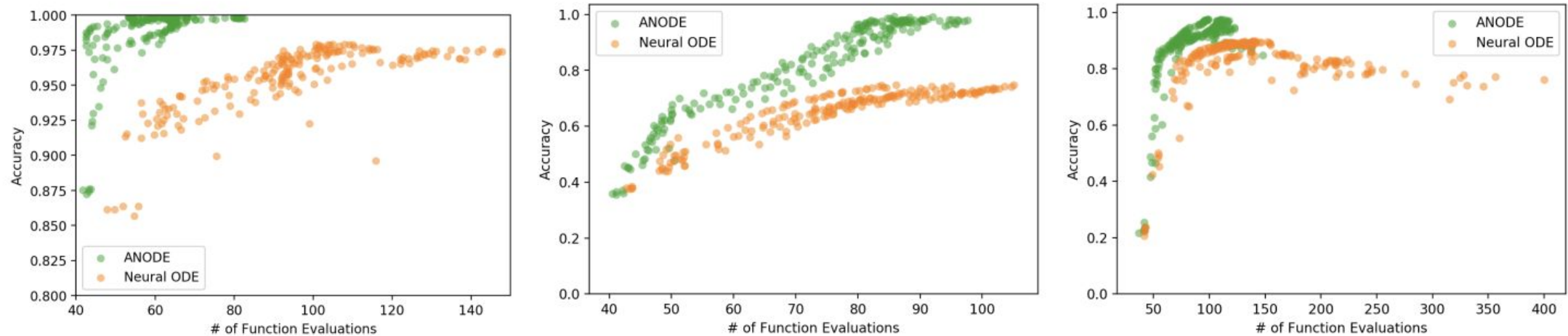
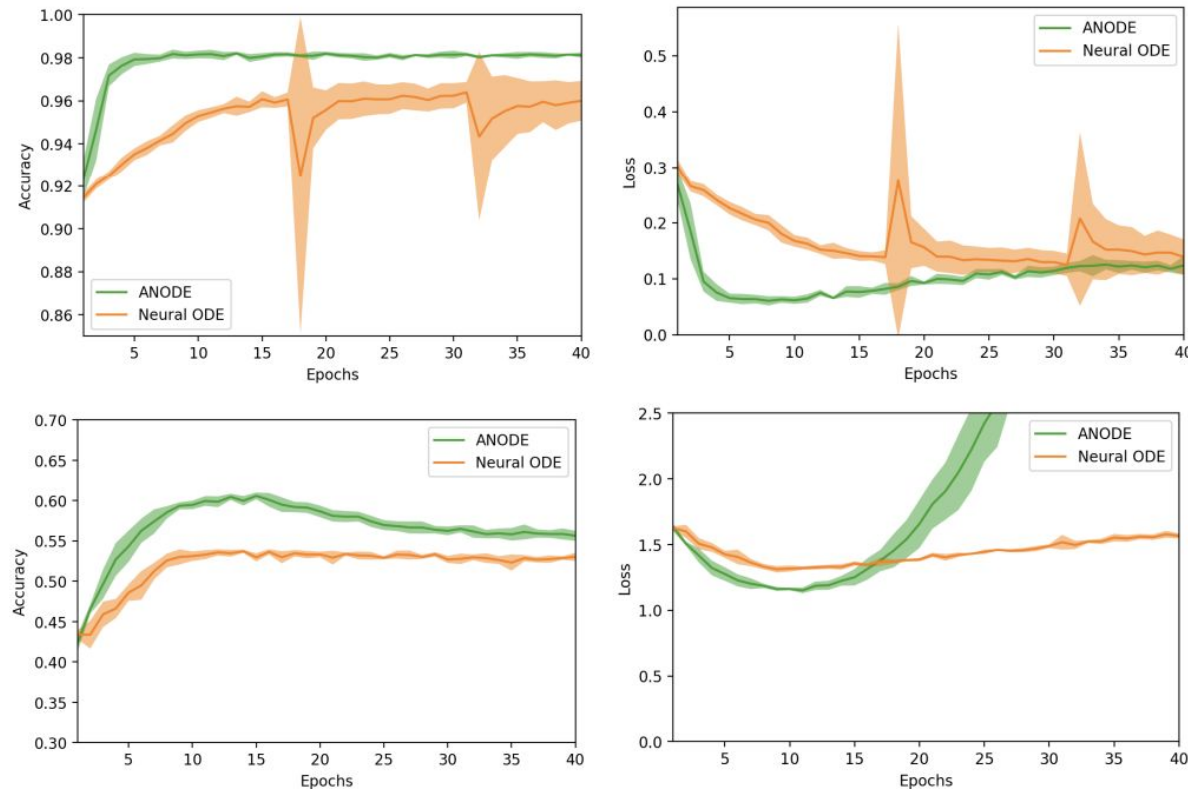


Рисунок 11: Точность vs NFEs для MNIST (слева), CIFAR10 (середина) и SVHN (справа).



# Эксперименты с изображениями

## Обобщение для изображений.



	NODE	ANODE
MNIST	96.4% $\pm$ 0.5	<b>98.2% <math>\pm</math> 0.1</b>
CIFAR10	53.7% $\pm$ 0.2	<b>60.6% <math>\pm</math> 0.4</b>
SVHN	81.0% $\pm$ 0.6	<b>83.5% <math>\pm</math> 0.5</b>

Таблица 1: Тестовая точность и ее стандартное отклонение по 5 запускам на различных наборах данных изображений.

Рисунок 12: Тестовая точность и потери



# Эксперименты с изображениями

## Устойчивость.

22

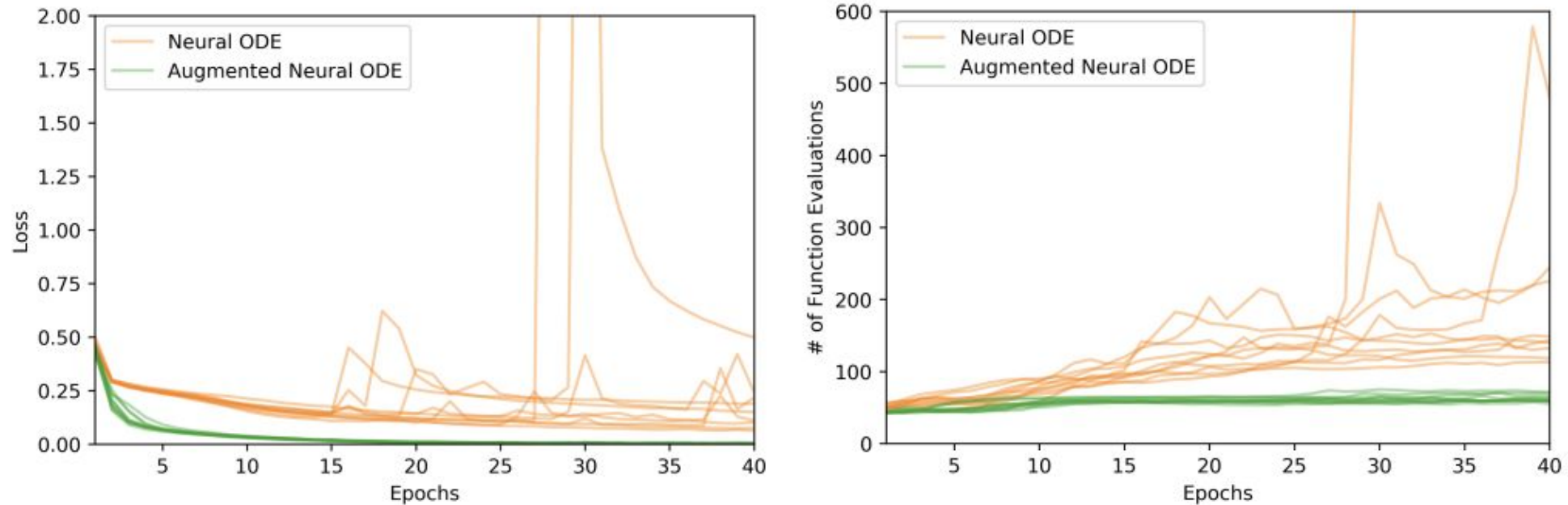


Рисунок 13: Нестабильности в потере (слева) и NFEs (справа) при подгонке NODEs к MNIST.





## Взаимосвязь с другими моделями

23

- ANODE преодолевают ограничения представления NODE, сохраняя все его привлекательные свойства. Улучшение NODE может привести к потере его таких свойств.
- Важным отличием при использовании аугментации для NODE является снижение вычислительных затрат, чего нет в ResNets.
- Использование аугментации для дискретных нормализующих потоков, таких как RealNVP, MAF и Glow может улучшить производительность.

ANODE



NODE

ResNet

Нормализующие  
потоки



- ANODEs быстрее, чем NODEs, но все же медленнее, чем ResNets.
- Дополнение изменяет размерность входного пространства, что в зависимости от области применения может быть нежелательным.
- Дополненную размерность можно рассматривать, как дополнительный гиперпараметр для настройки. Хотя модель устойчива для широкого диапазона дополненных размерностей, но для чрезмерно больших дополненных размерностей (например, добавление 100 каналов к MNIST) модель обычно работает хуже с более высокими потерями и NFEs.

**Преодоление ограничений NODEs.** Чтобы позволить траекториям пересекаться друг с другом, дополняется пространство, на котором решается ОДУ. Однако могут быть другие способы добиться этого, например, обучение дополнению (как в ResNets) или добавление шума (подобно Wang et al. (2018)).

**Дополнение для нормализующих потоков.** NFEs обычно становится неприемлемо большим при обучении моделей непрерывного нормализующего потока (CNF) (Grathwohl et al., 2018). Добавление дополнения к CNFs, вероятно, могло бы смягчить этот эффект

**Улучшенное понимание дополнения.** Было бы полезно предоставить более теоретический анализ того, как и почему дополнение улучшает обучение NODEs и как это может повлиять на выбор архитектур и оптимизаторов для NODEs.

В этой статье авторы:

- Выявили и проанализировали некоторые ограничения Нейронных ОДУ.
- Доказали, что существуют классы функций, которые NODEs не могут представлять
- Доказали, что NODEs обучаются только на признаках, которые гомеоморфны входному пространству. Экспериментально показали, что это приводит к более медленному обучению и сложным потокам, которые вычислительно дороги.
- Чтобы смягчить эти проблемы, был предложен Augmented Neural ODEs (ANODEs), которые обучаются потоку от входа к признакам в дополненном пространстве.
- Провели эксперименты, которые показывают, что ANODEs могут моделировать более сложные функции, используя более простые потоки, одновременно снижая потери, снижая вычислительные затраты и улучшая устойчивость и обобщение.