



Параметризация теплового излучения с помощью NeuralODE

Студенты: Александр Пономаренко

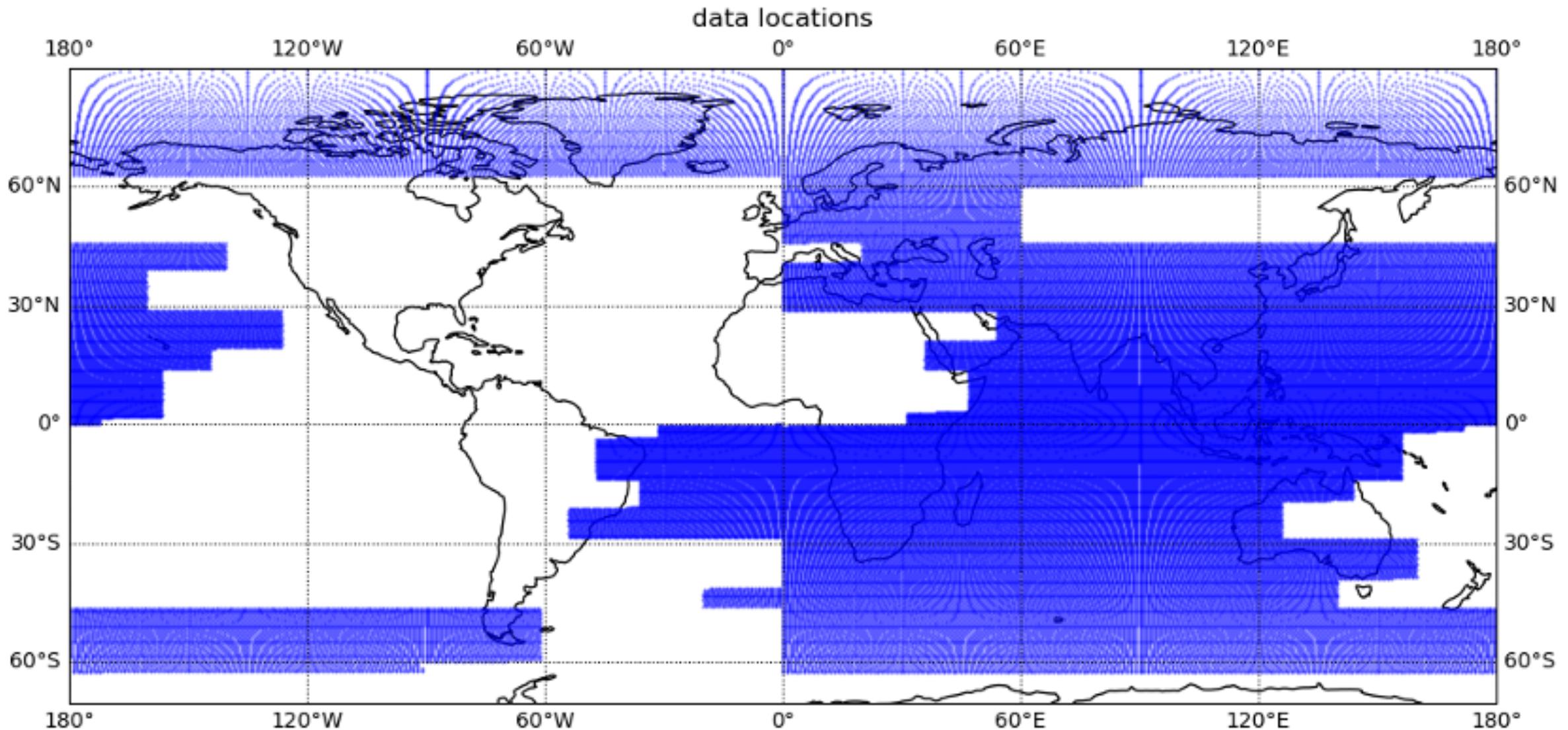
Кураторы: Александр Рябов и Николай Явич



137 слоев

135 680 вертикальных профилей состояний атмосферы

<https://disk.yandex.ru/d/iUvIzjdfvqVWwQ>



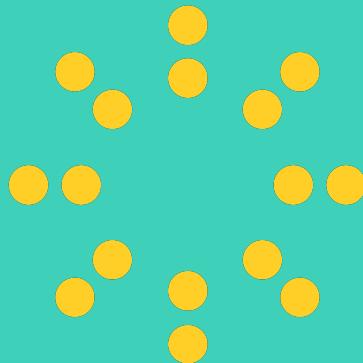
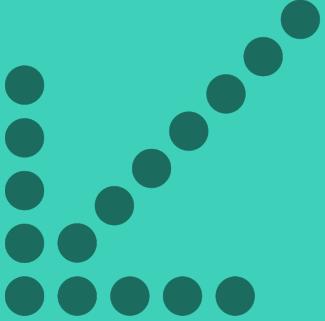
Постановка задачи

Цель: разработать
суррогатную модель расчёта
восходящего потока
теплового излучения $F^\uparrow(z)$,
с поверхности земли





$$\begin{cases} \frac{d}{dz} F^\uparrow(z) = g(F^\uparrow(z), \theta), \\ F^\uparrow(0) = \sigma T_s^4 \end{cases}$$



высота над уровнем
поверхности земли

неизвестная динамика

параметр динамики

$$\begin{cases} \frac{d}{dz} F^\uparrow(z) = g(F^\uparrow(z), \theta), \\ F^\uparrow(0) = \sigma T_s^4 \end{cases}$$

постоянная

Стефана-Больцмана

температура
поверхности



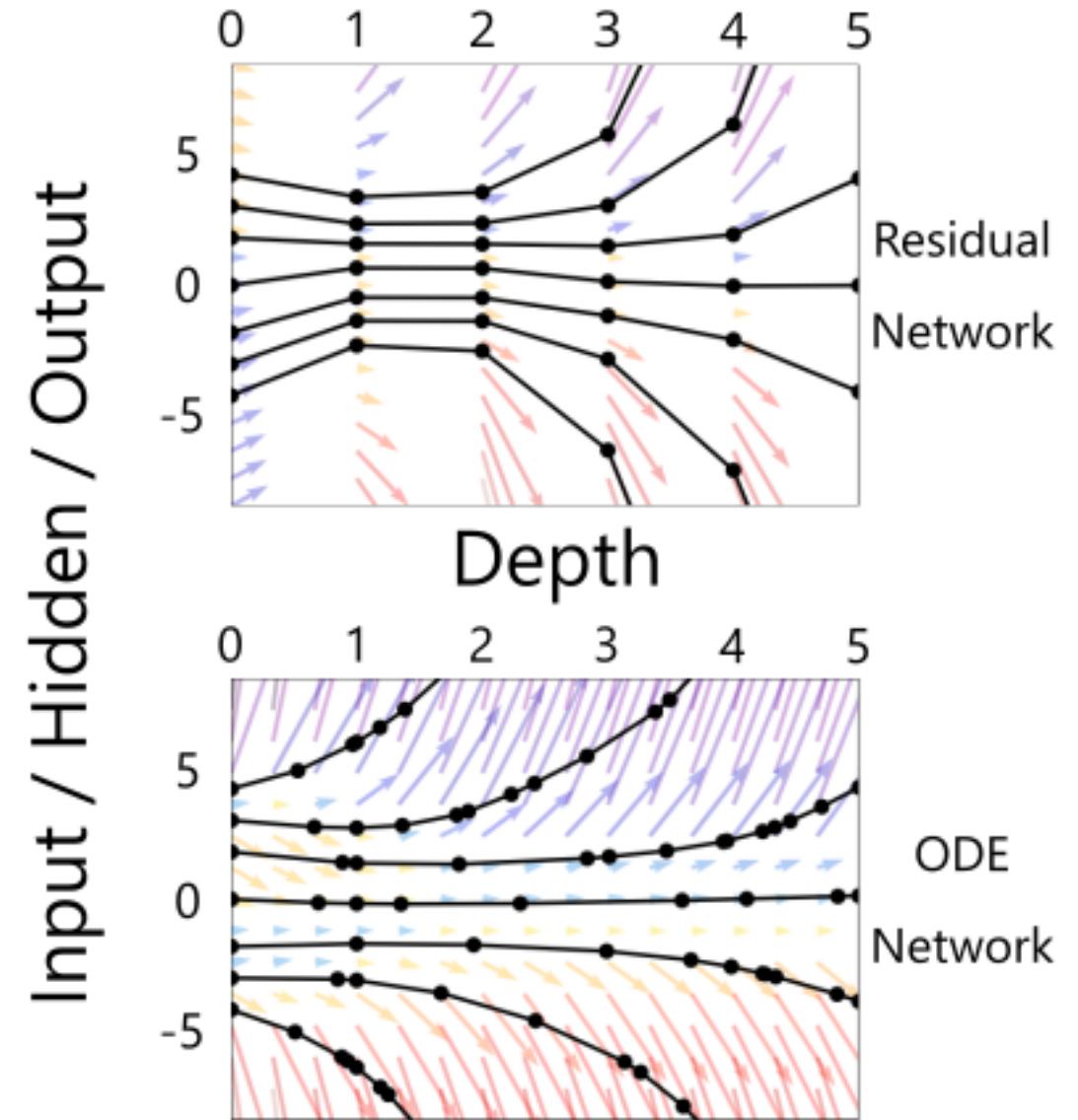
Традиционные подходы расчета нагрева атмосферы чрезвычайно вычислительно затратные, так как требуют многократного решения интегро-дифференциального уравнения переноса.

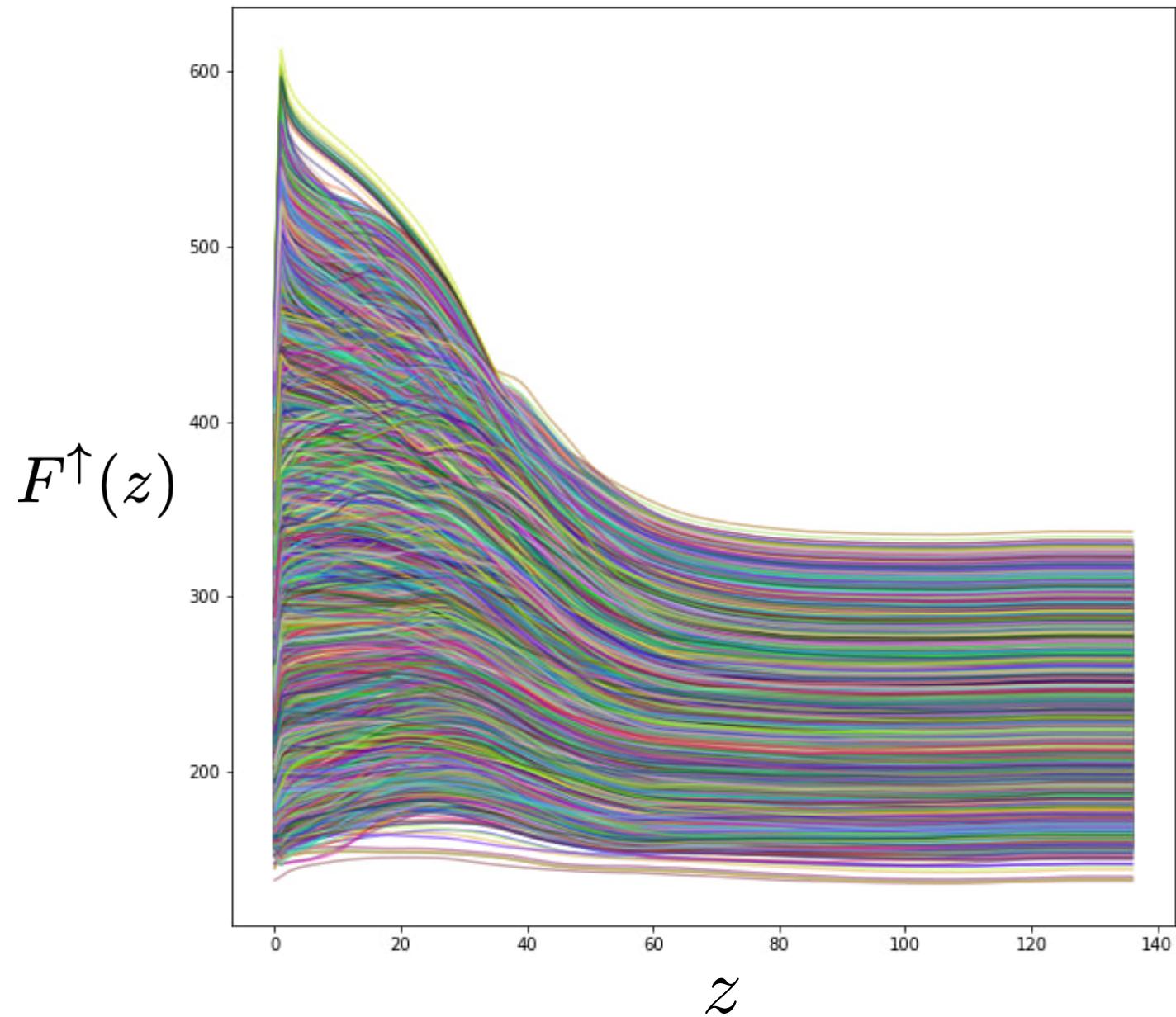


ResNet определяет дискретную последовательность конечных преобразований,

Neural ODE определяет векторное поле, которое непрерывно трансформирует состояние.

Черные точки представляют оценки



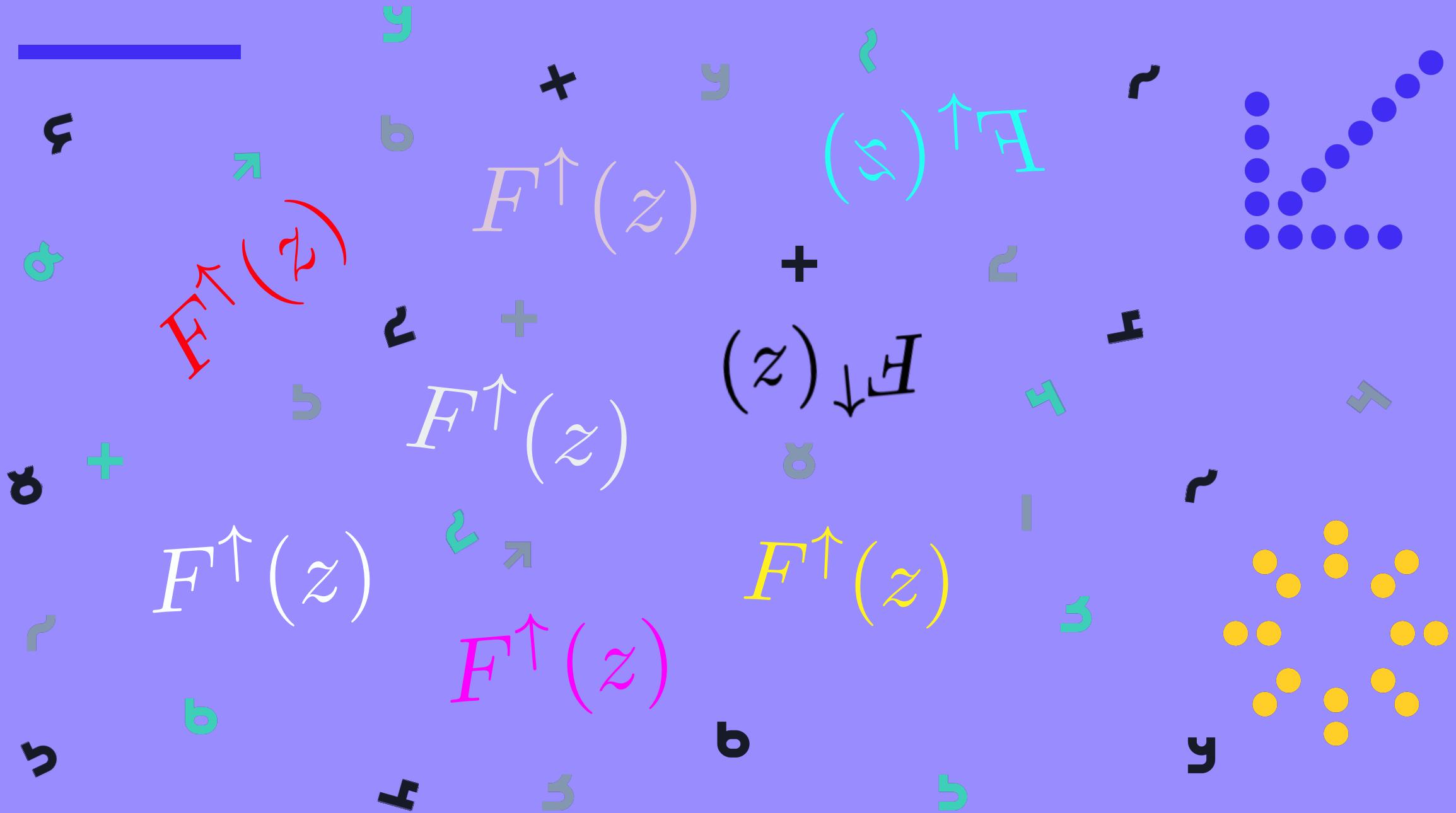


Variable Name	Meaning	Dimensions	Minimum Value	Maximum Value	Units
aerosol_mmr	12 types of aerosol MMR	(135680, 12, 137)	2.87E-21	2.52E-07	1
cfc11_vmr	CFC-11 VMR	(135680, 137)	0	8.57E-10	1
cfc12_vmr	CFC-12 VMR	(135680, 137)	1.04E-37	4.97E-10	1
ch4_vmr	methane VMR	(135680, 137)	2.43E-07	2.03E-06	1
cloud_fraction	cloud fraction	(135680, 137)	0	1	1
co2_vmr	carbon dioxide VMR	(135680, 137)	0.000400577	0.000422012	1
n2o_vmr	Nitrous oxide VMR	(135680, 137)	9.42E-08	3.32E-07	1
o2_vmr	nitrogen VMR	(135680, 137)	0.20943998	0.20943998	1
o3_mmr	ozone MMR	(135680, 137)	1.37E-15	1.67E-05	1
overlap_param	cloud overlap parameter	(135680, 136)	0.021936536	0.99294233	1
pressure_hl	Pressure	(135680, 138)	0	104678.69	Pa
q	specific humidity	(135680, 137)	1.00E-12	0.021662947	1
q_ice	ice water mixing ratio	(135680, 137)	0	0.009069286	1
q_liquid	liquid water mixing ratio	(135680, 137)	0	0.001543405	1
re_ice	Ice cloud effective radius	(135680, 137)	1.31E-05	0.000100676	m
re_liquid	Liquid cloud effective radius	(135680, 137)	4.00E-06	3.00E-05	m
temperature_hl	Temperature	(135680, 138)	178.16977	318.41193	K
lat	Latitude	(135680)	-62.692627	89.73092	degree
lon	Longitude	(135680)	0	359.65384	degree
lw_emissivity	Longwave emissivity	(135680, 2)	0.9237313	0.9973768	1
skin_temperature	Skin temperature	(135680)	216.04092	325.48035	K
flux_up_lw	Upwelling longwave fluxes,	(135680, 138)	74.891685	627.7309	Wt/m^2

ДОСТУПНЫЕ ПРИЗНАКИ

Хронология проекта

- 17 Выбрал проект, на который никто больше не записался
- 18 Получил описание и данные
- 20 Посмотрел статью про NeuralODE. Понятного мало.
- 21 В баре с сёбутыльниками однокашниками разобрали статью. Якобы что-то прояснилось.
- 22 Не без боли установил TorchDyn. Загрузил данные.
- 24 Тревожно. В примерах понятного мало.
- 25 Собрал минимально работающий пример только с одним регрессом. Как добавить ещё – не понятно.
- 26 Пообщался с Петей Чижовым, осознал, что всё делаю не правильно. Всё переделал. Поставил вычисления на ночь.
- 27 Результаты получились плохими. Всё плохо.
Ночью в 0:40 куратор натолкнул на мысль, что всё же до этого шёл по правильному пути.
1:40 Понял как правильно добавлять в систему произвольное число регрессоров. Кодирую.
3:00 Запустил вычисления на ночь
- 28 Всё заработало



θ

```
nn.Sequential(  
    nn.Linear(9, 64),  
    nn.Tanh(),  
    nn.Linear(64, 9)  
)
```

всего 1.2k
параметров

$F^\uparrow(0)$

температура
земли (константа)

уровень облачности
(константа)

содержание озона
(константа)

Ice cloud effective radius
(константа)

Liquid cloud effective radius
(константа)

ice water mixing ratio
(константа)

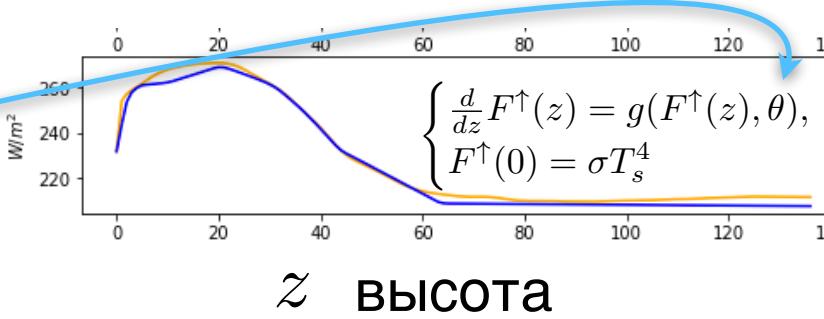
liquid water mixing ratio
(константа)

температура
воздуха (по слоям)

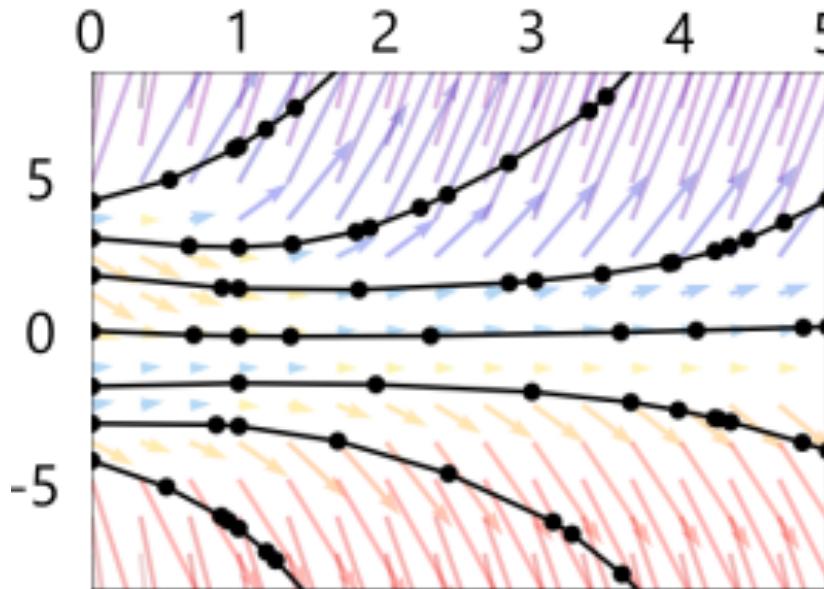
$x \in \mathbb{R}^9$

state

$F^\uparrow(z)$



$F^\uparrow(137)$



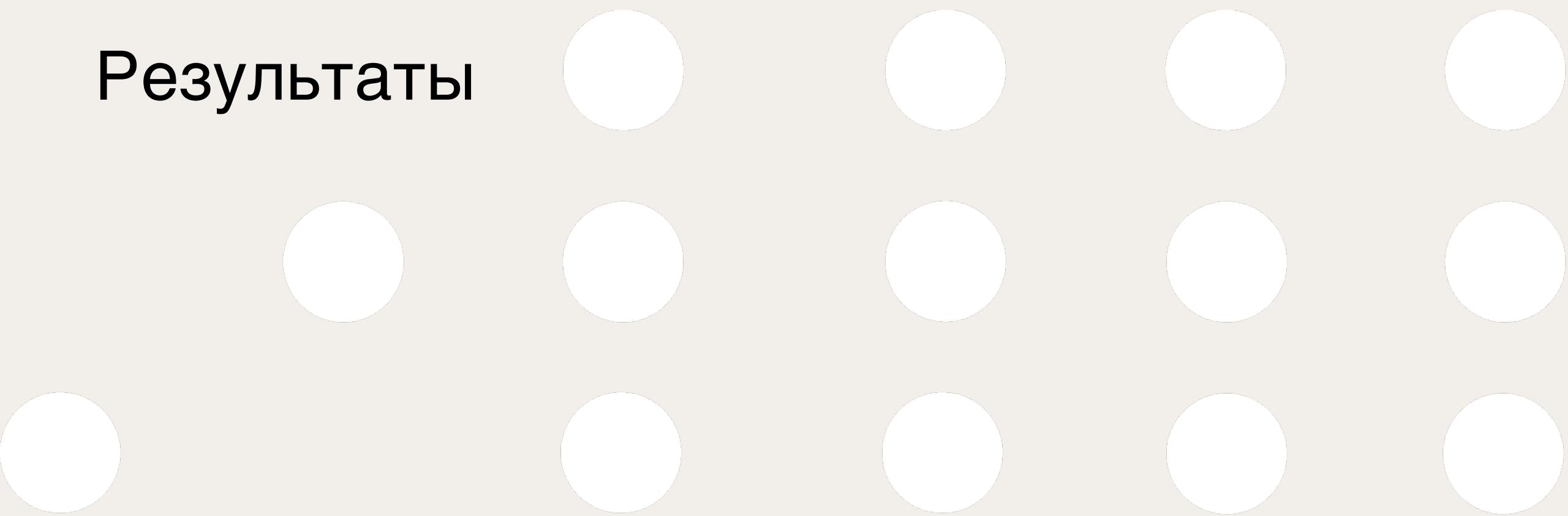
Neural ODE

$$l_\theta := \int_0^{137} g(F^\uparrow(z)) dz$$

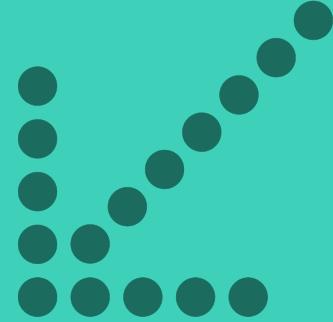
делал презентацию
и нашёл ошибку.
правильно брать
значения по слоям

02

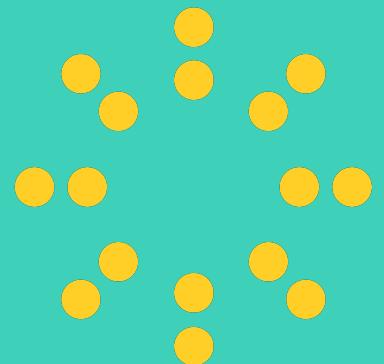
Результаты

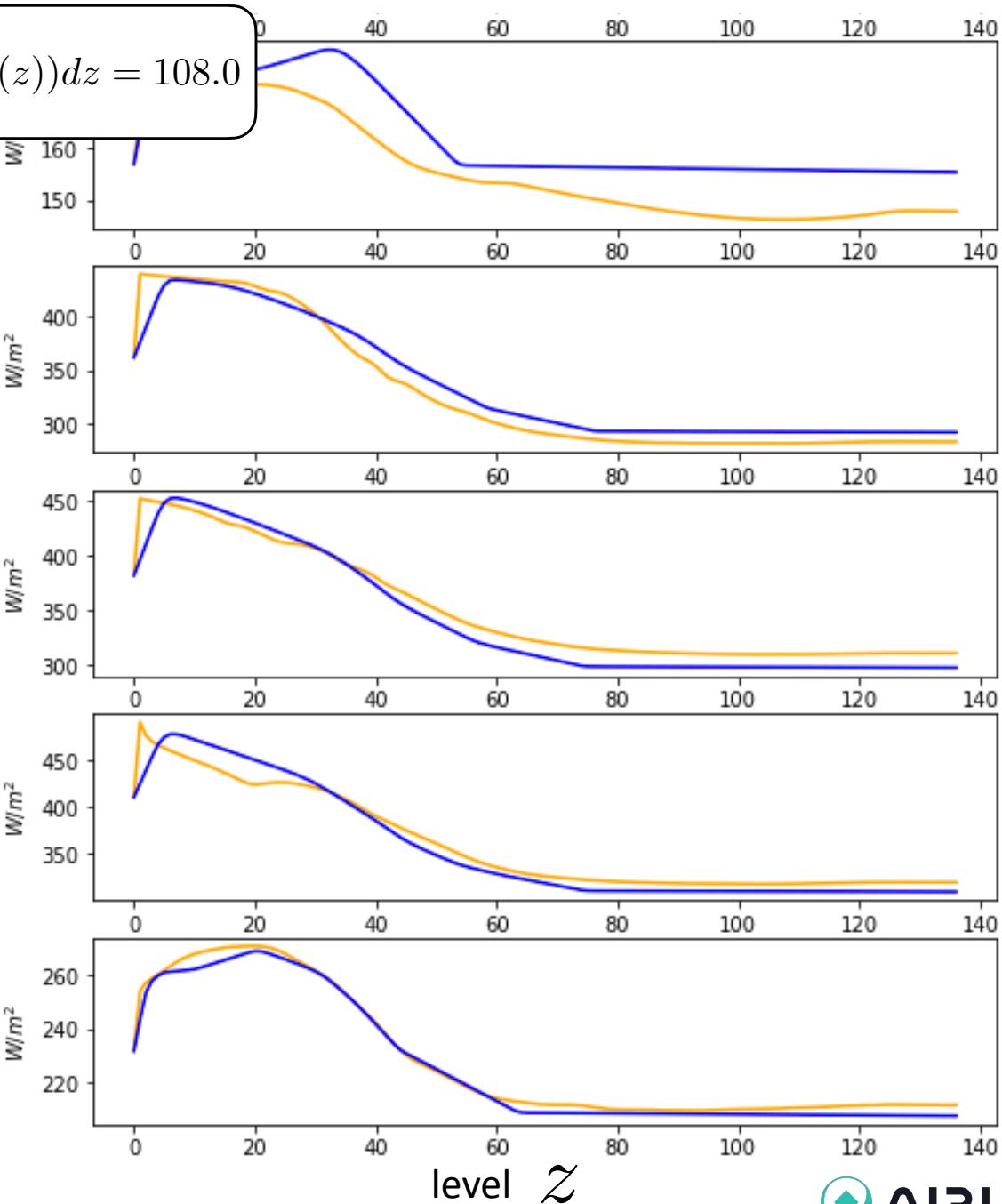
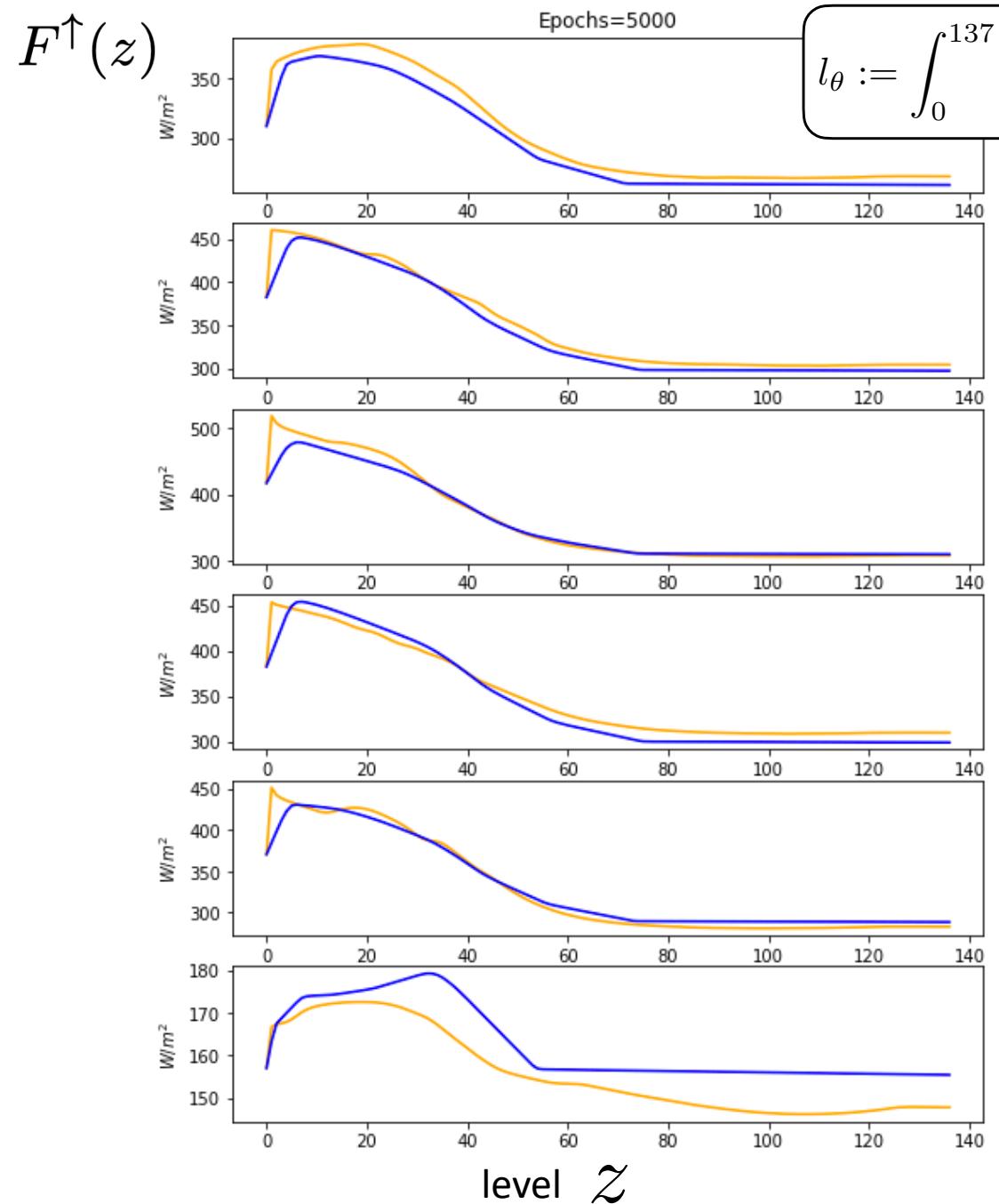


Использовалось только
5833 без облачных
регионов
max cloud_fraction < 0.001



Тренировка: 5716
Тест: 117





The diagram illustrates the decomposition of a vector field F into a sum of divergence-free and curl-free components. The total vector field F is shown as a red arrow pointing upwards. It is decomposed into two terms:

$$F = \nabla \phi + \mathbf{v}$$

The first term, $\nabla \phi$, is shown as a blue arrow pointing upwards, representing a scalar potential. The second term, \mathbf{v} , is shown as a green arrow pointing upwards, representing a velocity field.

Each term is further decomposed into divergence-free and curl-free components:

$$\nabla \phi = \nabla \psi + \mathbf{w}$$

$$\mathbf{v} = \mathbf{u} + \mathbf{v}$$

The divergence-free component $\nabla \psi$ is shown as a red arrow pointing upwards, and the curl-free component \mathbf{w} is shown as a blue arrow pointing upwards. The curl-free component \mathbf{u} is shown as a green arrow pointing upwards, and the divergence-free component \mathbf{v} is shown as a yellow arrow pointing upwards.

On the right side of the diagram, there are three sets of colored dots representing the components of the vector field. The top set consists of blue dots forming a diagonal line. The middle set consists of yellow dots forming a diagonal line. The bottom set consists of orange dots forming a diagonal line.



@ALEXANDERPONOMA
REJKO

До новых встреч!

<https://github.com/aponom84/Thermal-Radiation-NeuralODE>

<https://disk.yandex.ru/d/iUvIzjdfvqVWwQ>



[Telegram](#)



[VK](#)



[YouTube](#)



[LinkedIn](#)