Универсальные дифференциальные уравнения

Комбинация лучших сторон разных подходов.

Влад Темкин

Высшая Школа Экономики
Факультет Физики

Стохастические процессы и моделирование

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

$$u' = f(u, p, t)$$

- хорошо интерполируют (очень хорошо)

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

$$u' = f(u, p, t)$$

- хорошо интерполируют (очень хорошо)
- легко интерпретировать

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

$$u' = f(u, p, t)$$

- хорошо интерполируют (очень хорошо)
- легко интерпретировать
- указывают на структуру процесса

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

$$u' = f(u, p, t)$$

Подходы к описанию мира
Физические модели

- хорошо интерполируют (очень хорошо)
- легко интерпретировать
- указывают на структуру процесса
- нужно понимать хоть что-то о мире

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

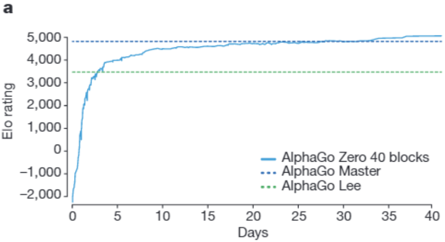
$$u' = f(u, p, t)$$

- хорошо интерполируют (очень хорошо)
- легко интерпретировать
- указывают на структуру процесса
- нужно понимать хоть что-то о мире
- проигрывают другим моделям (а именно машинному обучению), когда речь идёт о предсказаниях просто из набора данных

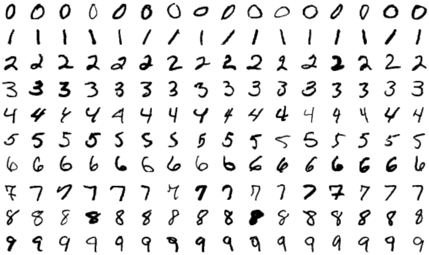
$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{x}} - \frac{\partial \mathcal{L}}{\partial x} = 0$$

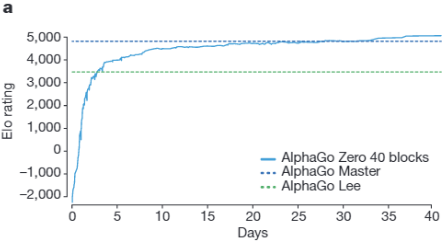
$$\begin{cases} \nabla \cdot \vec{D} = 4\pi\rho \\ \nabla \cdot \vec{B} = 0 \\ \nabla \times \vec{E} = \frac{1}{c}\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{D}}{\partial t} \end{cases}$$

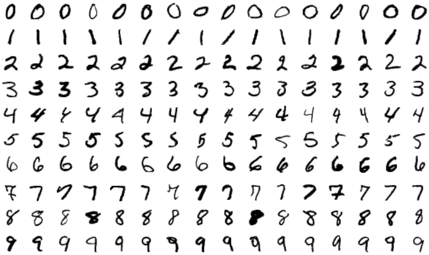
$$u' = f(u, p, t)$$



a
5,000
4,000
3,000
2,000
1,000
0
−1,000
−2,000
Elo rating
0
5
10
15
20
25
30
35
40
Days
AlphaGo Zero 40 blocks
AlphaGo Master
AlphaGo Lee



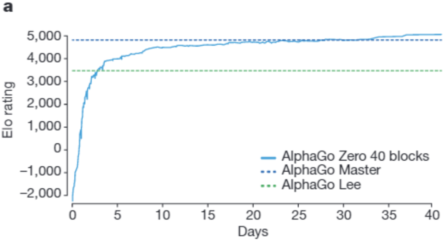


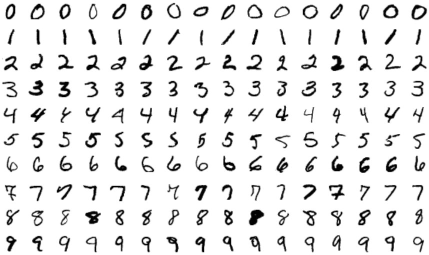


- может решить почти любую ограниченную задачу

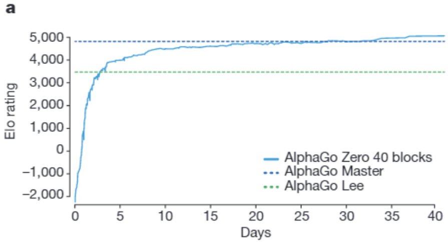
Подходы к описанию мира

Машинное обучение

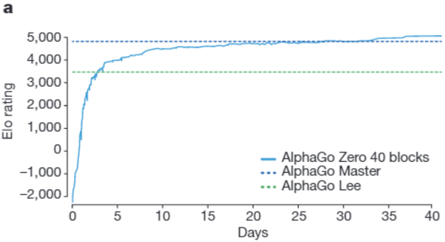


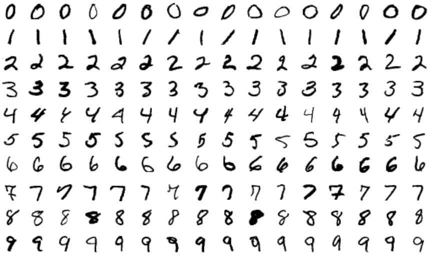


- может решить почти любую ограниченную задачу
- превосходит человека во многих областях

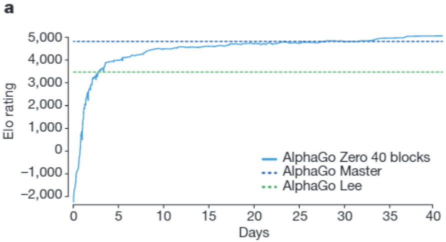


- может решить почти любую ограниченную задачу
- превосходит человека во многих областях
- потенциал кажется безграничным (обучение с подкреплением)

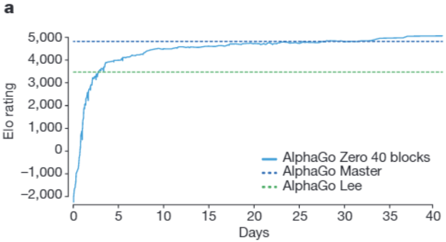


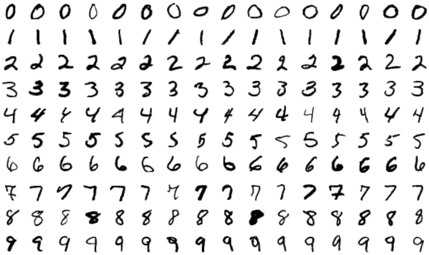


- может решить почти любую ограниченную задачу
- превосходит человека во многих областях
- потенциал кажется безграничным (обучение с подкреплением)
- сложно интерпретировать



- может решить почти любую ограниченную задачу
- превосходит человека во многих областях
- потенциал кажется безграничным (обучение с подкреплением)
- сложно интерпретировать
- не всегда хорошо интерполирует





- может решить почти любую ограниченную задачу
- превосходит человека во многих областях
- потенциал кажется безграничным (обучение с подкреплением)
- сложно интерпретировать
- не всегда хорошо интерполирует
- требуется много данных

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

Универсальная теорема аппроксимации

Нейронная сеть может аппроксимировать любую непрерывную функцию $R^n \to R^m$ с любой заданной точностью.

Комбинирование двух подходов

Различные практики

Универсальная теорема аппроксимации

Нейронная сеть может аппроксимировать любую непрерывную функцию $R^n \to R^m$ с любой заданной точностью.

- Physical Informed Neural Networks (PINNs):

$$u_t + \mathcal{N}[u, \lambda] = 0$$
$$f = u_t + \mathcal{N}[u, \lambda], u = NN$$

Универсальная теорема аппроксимации

Нейронная сеть может аппроксимировать любую непрерывную функцию $R^n \to R^m$ с любой заданной точностью.

- Physical Informed Neural Networks (PINNs):

$$u_t + \mathcal{N}[u, \lambda] = 0$$
$$f = u_t + \mathcal{N}[u, \lambda], u = NN$$

- Нейронные обыкновенные дифференциальные уравнения (NODE) (и не только обыкновенные):

$$u' = NN(u)$$

Комбинирование двух подходов

Различные практики

Универсальная теорема аппроксимации

Нейронная сеть может аппроксимировать любую непрерывную функцию $R^n \to R^m$ с любой заданной точностью.

- Physical Informed Neural Networks (PINNs):

$$u_t + \mathcal{N}[u, \lambda] = 0$$
$$f = u_t + \mathcal{N}[u, \lambda], u = NN$$

- Нейронные обыкновенные дифференциальные уравнения (NODE) (и не только обыкновенные):

$$u' = NN(u)$$

- Универсальные дифференциальные уравнения

2	4	7	6	5
9	7	1	2	6
8	3	4	5	7
4	3	3	8	4
5	2	1	1	2

5 x 5
Input Image

1	0	-1
1	0	-1
1	0	-1

3 x 3
Kernel

$$\Delta u = u_{xx} + u_{yy} =$$

$$= \frac{u(x+\Delta x, y) - 2u(x,y) + u(x-\Delta x, y)}{\Delta x^2} +$$

$$+ \frac{u(x, y+\Delta y) - 2u(x,y) + u(x, y-\Delta y)}{\Delta y^2}$$

$$\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}$$

$$u_t = \Delta u$$

$$u_t = CNN(u)$$

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

Универсальные дифференциальные уравнения
Что это?

WolframMathWorld: универсальные дифференциальные уравнения – это такие дифференциальные уравнения, чьи решения могут приблизить любую непрерывную функцию с заданной точностью.

Бриггс, 2002:

$$y''''y'^2 - 3y'''y''y' + 2(1 - n^{-2})y''^3 = 0 \text{ для } n > 3$$

WolframMathWorld: универсальные дифференциальные уравнения – это такие дифференциальные уравнения, чьи решения могут приблизить любую непрерывную функцию с заданной точностью.

Бриггс, 2002:

$$y''''y'^2 - 3y'''y''y' + 2(1 - n^{-2})y''^3 = 0 \text{ для } n > 3$$

Есть вариант попроще?

- Пусть у нас есть какая-то информация об интересующей нас системе. Допустим, мы знаем, как эволюционировала бы какая-то её часть, если бы она существовала отдельно. Тогда можем написать дифференциальное уравнение, описывающее эту эволюцию:

$$u' = f(u, p, t),$$

где u – какая-то наблюдаемая величина, t – время, p – прочие параметры системы.

- Пусть у нас есть какая-то информация об интересующей нас системе. Допустим, мы знаем, как эволюционировала бы какая-то её часть, если бы она существовала отдельно. Тогда можем написать дифференциальное уравнение, описывающее эту эволюцию:

$$u' = f(u, p, t),$$

  где u – какая-то наблюдаемая величина, t – время, p – прочие параметры системы.
- А что, если эта часть окажется не изолирована? Наше дифференциальное уравнение, скорее всего, окажется уже неверным...

- Пусть у нас есть какая-то информация об интересующей нас системе. Допустим, мы знаем, как эволюционировала бы какая-то её часть, если бы она существовала отдельно. Тогда можем написать дифференциальное уравнение, описывающее эту эволюцию:

  $$u' = f(u, p, t),$$

  где u – какая-то наблюдаемая величина, t – время, p – прочие параметры системы.
- А что, если эта часть окажется не изолирована? Наше дифференциальное уравнение, скорее всего, окажется уже неверным...
- Что делать?

- Добавим в правую часть дополнительное слагаемое, не уточняя его природу:
$$u' = f(u, p, t) + U_\theta(u),$$
где U_θ – универсальный аппроксиматор, то есть любая непрерывная функция.

- Добавим в правую часть дополнительное слагаемое, не уточняя его природу:
$$u' = f(u, p, t) + U_\theta(u),$$
где U_θ – универсальный аппроксиматор, то есть любая непрерывная функция.
- Все новые эффекты оказываются учтены в U_θ!

- Добавим в правую часть дополнительное слагаемое, не уточняя его природу:
$$u' = f(u, p, t) + U_\theta(u),$$
где U_θ – универсальный аппроксиматор, то есть любая непрерывная функция.
- Все новые эффекты оказываются учтены в U_θ!
- Но где взять информацию о нём?..

- Добавим в правую часть дополнительное слагаемое, не уточняя его природу:
$$u' = f(u, p, t) + U_\theta(u),$$
где U_θ – универсальный аппроксиматор, то есть любая непрерывная функция.
- Все новые эффекты оказываются учтены в U_θ!
- Но где взять информацию о нём?..
- Обучить нейронную сеть по собранным о системе данным!

Рецепт 1

Универсальное дифференциальное уравнение = известное уравнение + аппроксимированные неизвестные слагаемые

Рецепт 1

Универсальное дифференциальное уравнение = известное уравнение + аппроксимированные неизвестные слагаемые

Рецепт 2

Аппроксимация неизвестных слагаемых = данные о системе + машинное обучение

1. Определить известные части модели. Построить по ним UODE.

1. Определить известные части модели. Построить по ним UODE.
2. Обучить нейронную сеть (или любой другой универсальный аппроксиматор), чтобы найти неизвестные механизмы.

1. Определить известные части модели. Построить по ним UODE.
2. Обучить нейронную сеть (или любой другой универсальный аппроксиматор), чтобы найти неизвестные механизмы.
3. С помощью SINDy свести неизвестные части к механистическим слагаемым.

1. Определить известные части модели. Построить по ним UODE.
2. Обучить нейронную сеть (или любой другой универсальный аппроксиматор), чтобы найти неизвестные механизмы.
3. С помощью SINDy свести неизвестные части к механистическим слагаемым.
4. Проверить то, что полученные законы правдоподобны.

1. Определить известные части модели. Построить по ним UODE.
2. Обучить нейронную сеть (или любой другой универсальный аппроксиматор), чтобы найти неизвестные механизмы.
3. С помощью SINDy свести неизвестные части к механистическим слагаемым.
4. Проверить то, что полученные законы правдоподобны.
5. Экстраполировать полученную модель, исследовать предельные случаи.

1. Определить известные части модели. Построить по ним UODE.
2. Обучить нейронную сеть (или любой другой универсальный аппроксиматор), чтобы найти неизвестные механизмы.
3. С помощью SINDy свести неизвестные части к механистическим слагаемым.
4. Проверить то, что полученные законы правдоподобны.
5. Экстраполировать полученную модель, исследовать предельные случаи.
6. Получить новые данные и проверить на них полученную модель.

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

- Sparse
- Identification of
- Nonlinear
- Dynamics

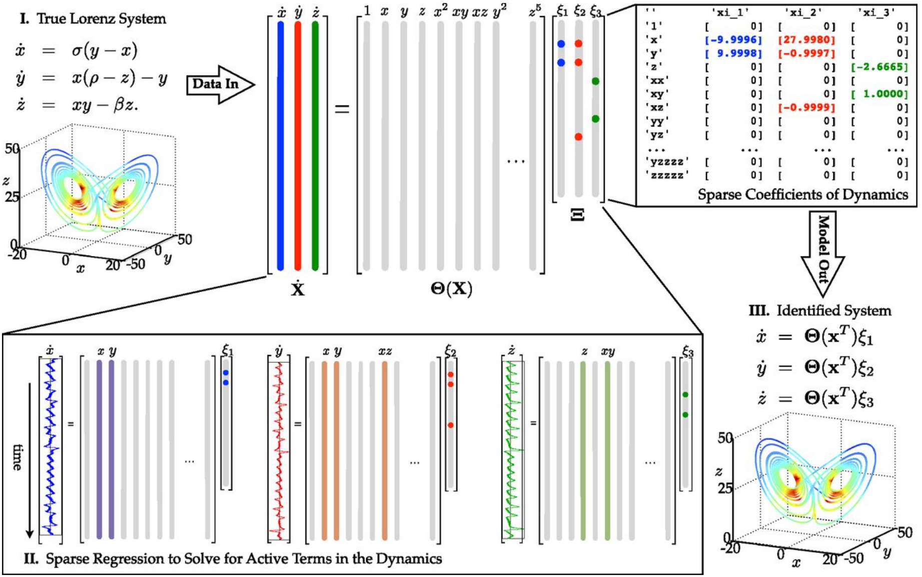
- Sparse
- Identification of
- Nonlinear
- Dynamics

- Собрать данные: "координаты"и их "производные"
- Составить библиотеку базисных функций
- Решить задачу регрессии
- Выбрать простейшее решение

- Sparse
- Identification of
- Nonlinear
- Dynamics

- Собрать данные: "координаты"и их "производные"
- Составить библиотеку базисных функций
- Решить задачу регрессии
- Выбрать простейшее решение

Everything should be made as simple as possible, but no simpler.



I. True Lorenz System
$\dot{x} = \sigma(y - x)$
$\dot{y} = x(\rho - z) - y$
$\dot{z} = xy - \beta z.$
Data In
$\dot{\mathbf{X}}$
$\Theta(\mathbf{X})$
Ξ
'' 'xi_1' 'xi_2' 'xi_3'
'1' [0] [0] [0]
'x' [-9.9996] [27.9980] [0]
'y' [9.9998] [-0.9997] [0]
'z' [0] [0] [-2.6665]
'xx' [0] [0] [0]
'xy' [0] [0] [1.0000]
'xz' [0] [-0.9999] [0]
'yy' [0] [0] [0]
'yz' [0] [0] [0]
'yzzzz' [0] [0] [0]
'zzzzz' [0] [0] [0]
Sparse Coefficients of Dynamics
Model Out
III. Identified System
$\dot{x} = \Theta(\mathbf{x}^T)\xi_1$
$\dot{y} = \Theta(\mathbf{x}^T)\xi_2$
$\dot{z} = \Theta(\mathbf{x}^T)\xi_3$
time
II. Sparse Regression to Solve for Active Terms in the Dynamics

План доклада

Основные моменты

1. Подходы к описанию мира
   - Физические модели
   - Машинное обучение
2. Комбинирование двух подходов
3. Универсальные дифференциальные уравнения
4. Вспомогательные инструменты
   - алгортим SINDy
5. Примеры применения UDE
   - Модифицированная модель SEIR
   - Модель Лотки-Вольтерра

Примеры применения UDE

- Обыкновенные универсальные дифференциальные уравнения

$$u' = f(u, p, t) + U_\theta(u)$$

- Универсальные нейронные дифференциальные уравнения:

$$u_t = U_\theta(u) + D \cdot CNN(u)$$

- Многомерные универсальные дифференциальные уравнения в частных производных
- Универсальные алгебро-дифференциальные уравнения
- Стохастические универсальные дифференциальные уравнения
- Любые другие, где можно придумать, как заменить часть слагаемых на универсальный аппроксиматор.

$$
\begin{aligned}
S' &= -\beta_0 F \frac{S}{N} - \beta \frac{SI}{N} - \mu S \quad \textit{susceptible} \\
E' &= \beta_0 F \frac{S}{N} + \beta \frac{SI}{N} - (\sigma + \mu) E \textit{exposed} \\
I' &= \sigma E - (\gamma + \mu) I \quad \textit{infected} \\
R' &= \gamma I - \mu R \quad \textit{removed} \\
N' &= -\mu N \quad \textit{population} \\
D' &= d\gamma I - \lambda D \quad \textit{severecases} \\
C' &= \sigma E \quad \textit{cumulativecases}
\end{aligned}
$$

$$\beta = \beta_0 (1 - \alpha)(1 - \frac{D}{N})^{\kappa}$$

Примеры применения UDE

Модифицированная модель SEIR

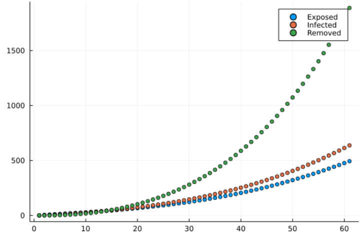


Рис.: Решение исходной системы уравнений.

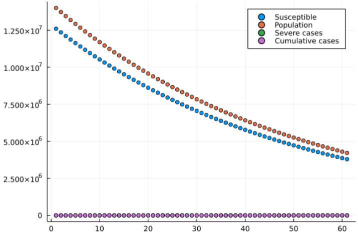
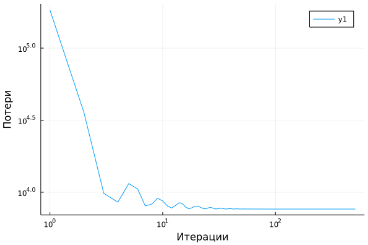


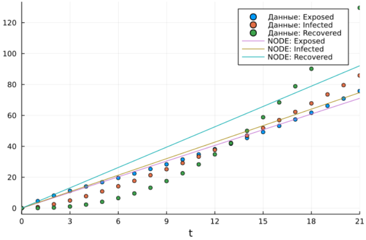
Рис.: То же самое, другие величины.

Определим NODE:

$$\frac{d}{dt}X = \mathcal{NN}(X)$$

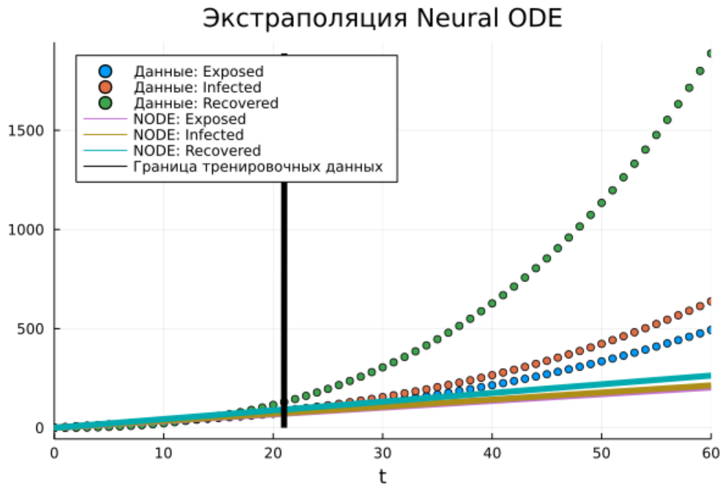
NN=FastChain(...)
loss=sum(abs2,data.-solve(prob))





Примеры применения UDE

Модифицированная модель SEIR: Neural ODE



Примеры применения UDE

Модифицированная модель SEIR

$$
\begin{gathered}
S' = -\beta_0 F \frac{S}{N} - \mathcal{N}\mathcal{N} - \mu S \\
E' = \beta_0 F \frac{S}{N} + \mathcal{N}\mathcal{N} - (\sigma + \mu) E \\
I' = \sigma E - (\gamma + \mu) I \\
R' = \gamma I - \mu R \\
N' = -\mu N \\
D' = d\gamma I - \lambda D \\
C' = \sigma E
\end{gathered}
$$

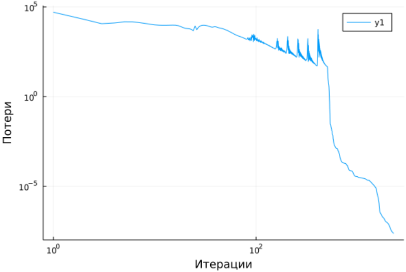


Рис.: Функция потерь при обучении UODE.

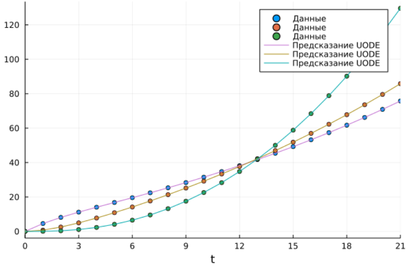


Рис.: Обучающая выборка.

Примеры применения UDE

Модифицированная модель SEIR

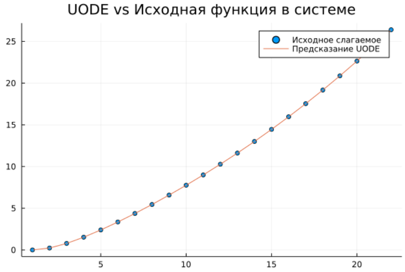


Рис.: Функция потерь при обучении UODE.

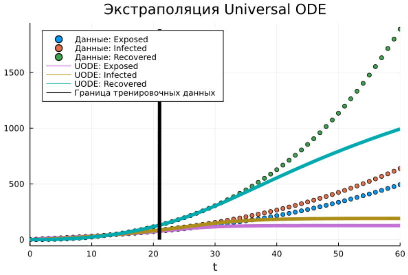


Рис.: Экстраполяция обученного UODE.

Примеры применения UDE

Модель Лотки-Вольтерра

- Модель Лотки-Вольтерра:

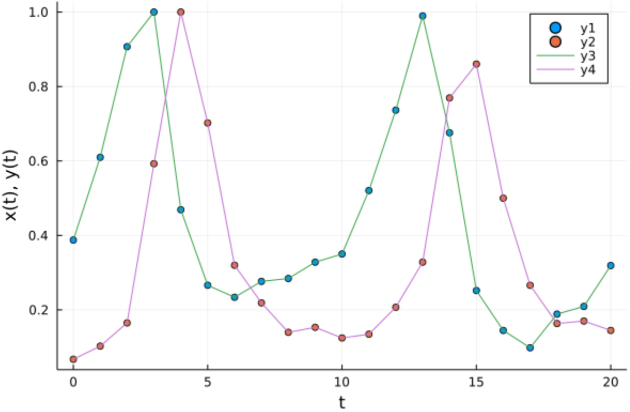
$$\dot{x} = \alpha x - \beta xy$$
$$\dot{y} = -\gamma y + \delta xy$$

- Забыли о вторых слагаемых в правых частях:

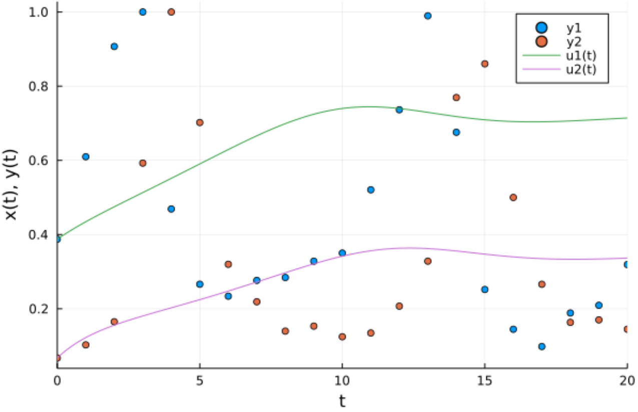
$$\dot{x} = \alpha x + U_x(x, y)$$
$$\dot{y} = -\gamma y + U_y(x, y)$$

- Зайцы экспоненциально плодятся в отсутствии хищника
- Волки экспоненциально умирают из-за отсутвия добычи
- Механизм взаимодействия не знаем =(

К счастью, у нас в руках оказались данные (Hudson-Bay), которые могут описываться этой моделью.



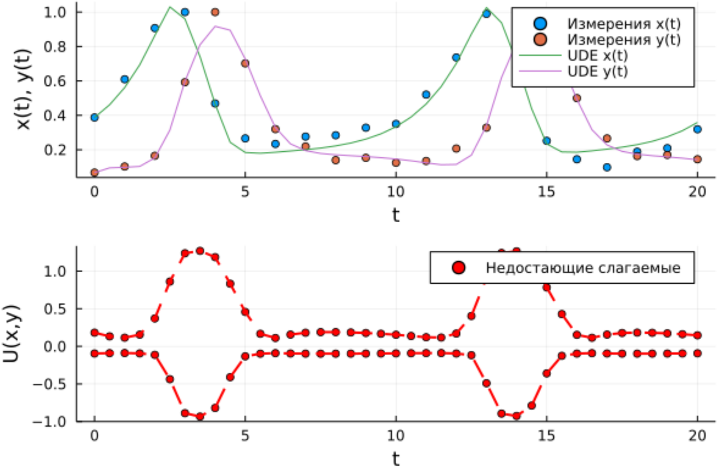
Использование SINDy просто на исходных данных результата не даёт.



Примеры применения UDE

Модель Лотки-Вольтерра

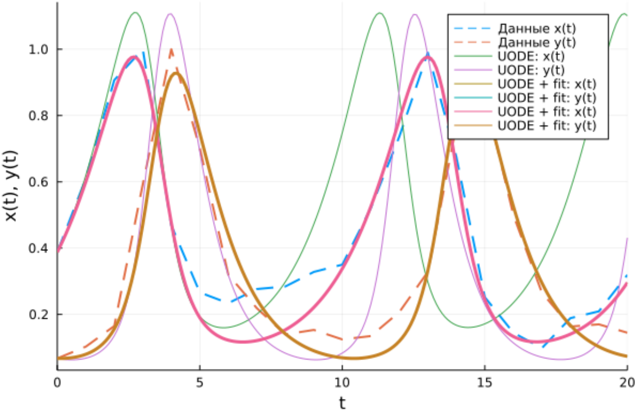
Обучаем наш UODE и смотрим на результат.



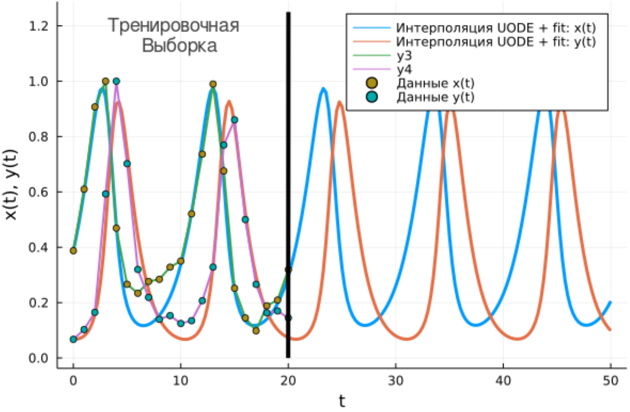
- На полученном результате можем снова попробовать применить SINDy и получаем:
  Differential(t)(u[1]) = p_1*u[1]*u[2]
  Differential(t)(u[2]) = p_2*u[1]*u[2]
  (-1.56, 2.4)
- Здесь была использована библиотека полиномов до 3 степени включительно, а также $\sin u, \cos u$.
- UODE + SINDy восстановили модель Лотки-Вольтерра!
- При этом коэффициенты в полученной модели всё ещё можно улучшить, чтобы добиться большей точности

Примеры применения UDE
Модель Лотки-Вольтерра

Обучаем параметры $(\alpha, \beta, \gamma, \delta)$ полученной модели под исходные данные.
Получили $\alpha \approx 0.557, \beta \approx 0.826, \gamma \approx -1.7, \delta \approx 2.04$.



Имея хорошую модель, можем экстраполировать её на большие времена!



Список литературы

- Christopher Rackauckas and Yingbo Ma and Julius Martensen and Collin Warner and Kirill Zubov and Rohit Supekar and Dominic Skinner and Ali Ramadhan and Alan Edelman. Universal Differential Equations for Scientific Machine Learning.
- Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems.
- Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations
- Silver, D., Schrittwieser, J., Simonyan, K. et al. Mastering the game of Go without human knowledge. Nature 550, 354–359 (2017). https://doi.org/10.1038/nature24270