Neural SDE для нахождения моментов разладки временных рядов

Папай Иван Дмитриевич

Московский физико-технический институт

Курс: Автоматизация научных исследований
(практика, В. В. Стрижов)/Группа 128б

Эксперт: к.ф. м.н.,с. ВЦ РАН В. В. Стрижов

Консультант: студент магистратуры 5 курс Э. А. Владимиров

2024

Цель исследования

Цель

Главной целью исследования является подбор оптимального способа поиска точек разладки во временных рядах. Под точками разладки понимаются позиции временного ряда, по достижению которых временной ряд резко меняет свое поведение в своем стохастическом смысле.

Задача

По имеющемуся временному ряду уметь вычислять его точки разладки. При этом ошибка первого рода у модели должна быть минимальной.

Суть вкратце

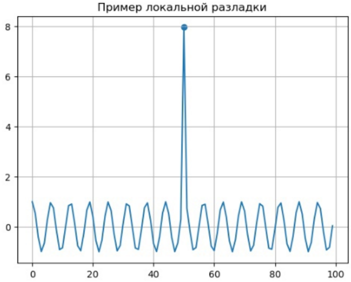

Рис.: локальная разладка

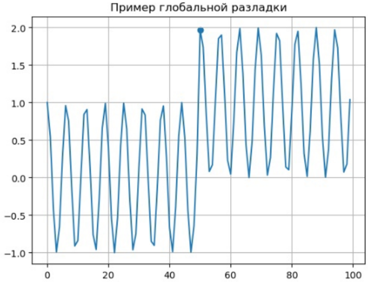

Рис.: глобальная разладка

Литература

[1] “Neural Ordinary Differential Equations Ricky” T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, David Duvenaud
[2] “Neural SDE: Stabilizing Neural ODE Networks with Stochastic Noise” Xuanqing Liu, Tesi Xiao, Si Si, Qin Cao, Sanjiv Kumar, Cho-Jui Hsieh
[3] “Riemannian Neural SDE: Learning Stochastic Representations on Manifolds” Sung Woo Park , Hyomin Kim , Kyungjae Lee , Junseok Kwon
[4] “Riemannian Diffusion Models” Chin-Wei Huang, Milad Aghajohari, Avishek Joey Bose, Prakash Panangaden, Aaron Courville
[5] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In Advances in Neural Information Processing Systems, pages 6572–6583, 2018.

Модель ResNet

Изначально Neural ODE был разработан как альтернатива методу Residual Networks(ResNet), состоящих из последовательности скрытых слоёв, значения на каждом из которых подчинялись следующей формуле:

$$h_{k+1} = h_k + f(h_k, w_k) \qquad (1)$$

— где h_k - вход k-го слоя для $k \in [1, K]$, K-число слоёв и f(h_k, w_k) - функция, параметризованная по w_k - динамический параметр, задающийся перед началом обучения модели.

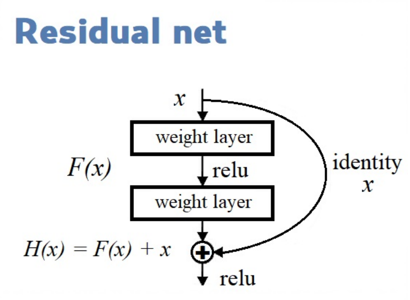

Рис.: схематичное описание работы

Модель Neural ODE

$$h_t = h_s + \int_s^t f(h_l, l; w)dl, \tag{2}$$

— вычисление такого дифференциального уравнения является задачей для Neural ODE. В данной работе в качестве числа слоев берется число элементов во временном ряду.

Algorithm 1 Neural ODE-solver

Require: динамические параметры w, начальное/конечное время t_0, t_1, конечное значение $z(t_1)$, градиент функции потерь в конечной точке $\frac{\delta L}{\delta z(t_1)}$

$s_0 = [z(t_1), \frac{\delta L}{\delta z(t_1)}, 0_{[w]}]$ ▷ Начальное состояние

$[z(t_0), \frac{\delta L}{\delta z(t_1)}, \frac{\delta L}{w}] = ODESolve(s_0, [f(z(t), t, w), -a(t)^T \frac{\delta f}{\delta z}, -a(t)^T \frac{\delta f}{\delta w}], t_1, t_0, w)$

$return \frac{\delta L}{\delta z(t_0)}, \frac{\delta L}{w}$ ▷ Возвращаем градиенты

Рис.: псевдокод Neural ODE

Здесь ODESolve - это черный ящик, возвращающий решение ODE — в качестве него предлагается использовать метод Рунге-Кутта, а L — MSE (mean squared error — среднее квадратов отклонения элементов выборки от их оценок).

Модель Neural CDE

В случае анализа временных рядовт было предложено [7] рассматривать не классический интеграл Римана-Стилтьеса, а стохастический интеграл Стратоновича:

$$h_t = h_s + \int_s^t f(h_l, l; w) dB_l, \tag{3}$$

— где $B_t = [B_t^1 ... B_t^K]$ - Винеровский процесс той же размерности, что и X_t.
Таким образом получается контролируемое дифференциальное уравнение, подходящее для данных типа Time-Series, особенно когда временные метки ряда не являются регулярными в плане тождественности их расстояний между собой.

Модель Neural SDE

Обобщением предыдущих двух методов является модель Neural SDE. Для учёта шума и максимально возможной апроксимации проноза в нашем дифференциальное уравнение следует учесть и недетерменированную компоненту, и диффузию. Получится следующее выражение, являющееся полным стохастическим дифференциалом:

$$dX_t^w = h(t, X_t^w; w)dt + \sigma(X_t^w; w)dB_t \tag{4}$$

— где $B_t = [B_t^1 ... B_t^K]$ - Винеровский процесс той же размерности, что и X_t, а $\sigma(X_t^w; w)$ - его матрица ковариаций в t-й момент времени

Обобщим это выражение для модели ResNet:

$$dh_t = f(h_t, t, w), \tag{5}$$

— таким образом выражение (1) для (k+1)-го слоя изменится:

$$h_{k+1} = h_k + f(h_k, w_k) + \sigma(X_k^w; w)B_k, \tag{6}$$

— соответственно алгоритм остаётся тем же, что и для ODE.

Вычислительный эксперимент(пример выборки)

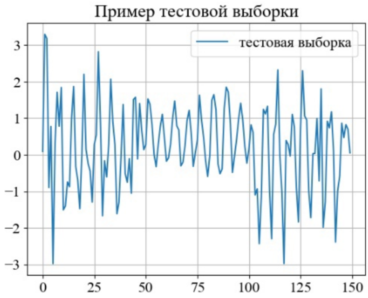

Рис.: выборка

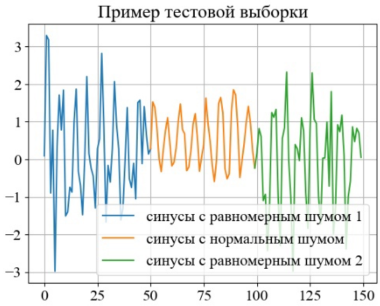

Рис.: происхождение разных отрезков ряда

Вычислительный эксперимент(подбор параметров)

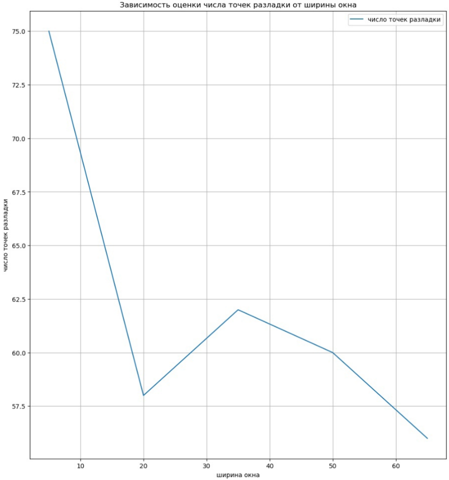

Рис.: подбор ширины окна

Вычислительный эксперимент(процесс обучения)

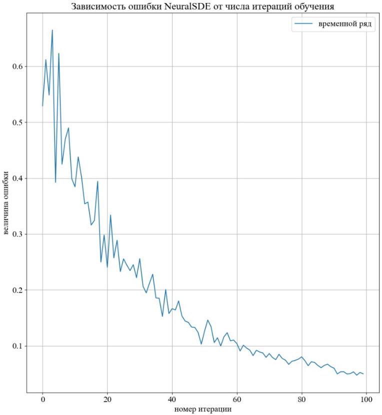

Рис.: процесс обучения

Строится матрица Ганкеля, описывающая фазовые траектории процесса, размера (n+1) на w вида:

$$\begin{bmatrix} f_1 & f_2 & \cdots & f_w \\ f_2 & f_3 & \cdots & f_{w+1} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n+1} & f_{n+2} & \cdots & f_{n+w} \end{bmatrix} \quad (7)$$

— где n - размер выборки, w - ширина окна.

Результаты поиска точек разладки

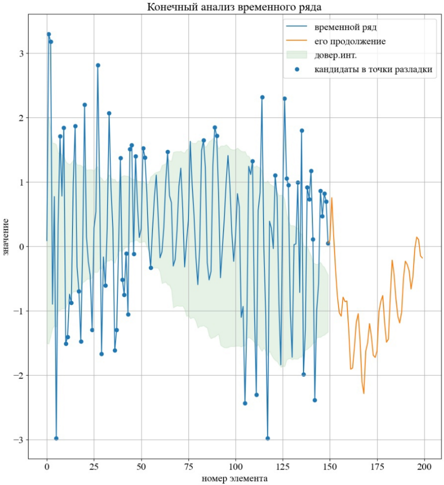

Рис.: воссоздание их фаз.траекторий с помощью Neural SDE

Сравнение с другими моделями

$$\begin{bmatrix} Model & Speed(sec) & Error \\ NeuralODE & 34.7 & 0.054 \\ NeuralCDE & 32.5 & 0.0545 \\ NeuralSDE & 47.2 & 0.044 \\ SSA & 6.48 & 0.12 \\ CuSum & 3.56 & 0.08 \\ CuSumSqr & 20.03 & 0.061 \end{bmatrix} \tag{8}$$

Заключение

- Результатом проведенной работы является обоснование корректности нового способа поиска точек разладки на практике. Плюсы и минусы метода ясны, и зная их, исследователь способен использовать его как новый и полезный инструмент.
- Дальнейшим направлением исследования будет оптимизация работы алгоритма и поиск способов уменьшить время, необходимое для его исполнения.