Метрический анализ пространства параметров глубоких нейросетей

Эрнест Р. Насыров

nasyrov.rr@phystech.edu

Исследуется проблема снижения размерности пространства параметров модели машинного обучения. Решается задача восстановления временного ряда. Для восстановления используются авторегресионные модели: линейные, автоенкодеры, реккурентные сети - с непрерывным и дискретным временем. Проводится метрический анализ пространства параметров модели. Предполагается, что отдельные параметры модели, случайные величины, собираются в векторы, многомерные случайные величины, анализ взаимного расположения которых в пространстве и представляет предмет исследования нашей работы. Этот анализ снижает число параметров модели, оценивает значимости параметров, отбирая их. Для определения положения вектора параметров в пространстве оцениваются его матожидание и матрица ковариации с помощью методов *бутстрэпа* и *вариационного вывода*. Эксперименты проводятся на задачах восстановления синтетических временных рядов, квазипериодических показаний акселерометра, периодических видеоданных. Для восстановления применяются модели SSA, нелинейного PCA, RNN, Neural ODE.

1 Introduction

Высокоразмерные данные избыточны, что представляет сложность для их эффективной обработки и использования. В работе решается задача снижения размерности признакового описания объекта. Ее базовый принцип состоит в том, чтобы отобразить высокоразмерное признаковое пространство в низкоразмерное, сохраняя важную информацию о данных [Jia et al., 2022].

На текущий момент известно много методов снижения размерности данных. В работе [Örnek and Vural, 2019] снижения размерности достигается за счет построения дифференцируемой функции эмбеддинга в низкоразмерное представление, а в [Cunningham and Yu, 2014] обсуждаюся линейные методы. В работе [Isachenko and Strijov, 2022] задача снижения размерности решается для предсказания движения конечностей человека по электрокортикограмме с использованием метода QPFS, учитывающем мультикоррелированность и входных, и целевых признаков.

Наряду с задачей снижения размерности входных данных стоит задача выбора оптимальной структуры модели. В случае оптимизации структуры нейросети, большое внимание уделено изучению признакового пространства модели. В работах [Hassibi et al., 1993] и [Dong et al., 2017] применяется метод OBS (Optimal Brain Surgeon), состоящий в удалении весов сети с сохранением ее качества аппроксимации, причем выбор удаляемых весов производится с помощью вычисления гессиана функции ошибки по весам.

В статье [**?**] приводится метод первого порядка, решающий задачу удаления весов, основанный на нахождении дисперсии градиента функции ошибки по параметру и анализе ковариационной матрицы параметров, а в статье [**?**] нерелевантные веса не удаляют, а прекращают их обучение.

Приведенные выше задачи понижения размерности данных и выбора оптимальной структуры нейросети основаны на исследовании пространства входных данных и пространства признаков соответственно. На взгляд авторов статьи, существенный недостаткок предыдущих работ состоит в том, что в них анализируются *отдельные* параметры (скаляры) моделей и их взаимозависимость. Тем самым не учитывается, что на входные данные действуют *вектора* параметров посредством скалярных произведений, то есть упускается из виду простая *структура* преобразования.

В данной работе мы решаем задачу восстановления временного ряда, в рамках которой занимаемся проблемой снижения пространства параметров модели, основанном на анализе сопряженного пространства ко входному, которое связывает входное пространство и пространство параметров.

Наше исследование в большой степени полагается на простоту устройства глубоких нейросетей, которые являются композицией линейных и простых нелинейных функций (функций активации). Составной блок нейросети описан формой: $y = \sigma(\mathsf{W}x), y \in \mathbb{R}^m, x \in \mathbb{R}^n, \mathsf{W} \in \mathbb{R}^{m\times n}, \sigma : \mathbb{R} \to \mathbb{R}$. Раньше элементы W_{ij} исследовались по-отдельности, как скаляры. Авторы работы предлагают изучать их как векторы-строки $\mathbf{w}_1, \dots, \mathbf{w}_m : \mathsf{W} = \begin{pmatrix} \mathbf{w}_1^\mathsf{T} \\ \dots \\ \mathbf{w}_m^\mathsf{T} \end{pmatrix}$. В нейросети эти строки обычно называются *нейронами.* В SSA $\sigma = Id$, а матрица $\mathsf{W} = \mathsf{W}_k$ это приближение истинной матрицы фазовых траекторий X (матрицы Ганкеля) суммой k элементарных матриц.

Пусть $\mathbf{x} = [x_1, \dots, x_N]^\mathsf{T}, x_i \in \mathbb{R}$ - временной ряд, $1 \leqslant n \leqslant N$ - ширина окна. Точка $\mathbf{x}_t = [x_t, \dots, x_{t+n-1}]^\mathsf{T}$ является точкой фазовой траектории временного ряда в траекторном пространстве $\mathbb{H}_\mathbf{x} \subset \mathbb{R}^n$.

Предполагается, что каждая точка фазовой траектории распределена нормально вокруг своего матожидания. Тогда и временной ряд является случайным, поэтому результат обучения модели на нем, то есть параметры обученной модели, будут случайными.

В работе исследуется положение случайных векторов параметров модели $\mathbf{w}_i$ в метрическом пространстве. С помощью методов бутстрэпа и вариационного вывода [Hastie et al., 2009] оцениваются их матожидания $\mathbf{e}_i = E(\mathbf{w}_i)$ и ковариационные матрицы $D(\mathbf{w}_i) = \mathbf{A}_i^{-1}$. Мы работаем в гипотезе, что эти векторы $\mathbf{w}_i$ распределены нормально, таким образом пара $(\mathbf{e}_i, \mathbf{A}_i^{-1})$ полностью описывает вероятностное распределение вектора $\mathbf{w}_i$.

В качестве графического анализа пространства изображаются положения этих векторов как смеси гауссианов 1. На рисунке изображены плотности функции распределения трех гауссовских векторов (вертикальная ось) в зависимости от их положения на плоскости. В каждой точке плотность равна сумме плотностей трех распределений, отнормирован-

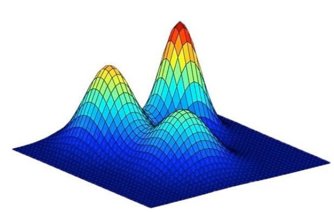

Рис. 1 Смесь гауссианов трех 2-х мерных векторов.

ная таким образом, чтобы площадь под графиком равнялась 1. Центрам 'куполов' соответствуют матожидания векторов, а их форма определяется матрицей ковариации. Таким образом, чем ниже и шире 'купол', тем больше дисперсия и наоборот.

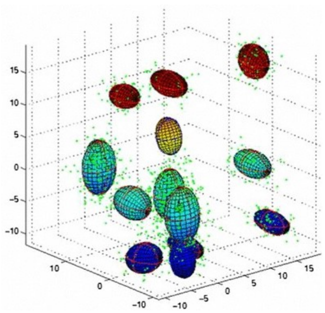

Рис. 2 Доверительные области 3-х мерных векторов.

Также изображается 95% доверительная область каждого вектора 2. На картинке изображены эллипсы, соответствующие доверительным областям для гауссовских векторов в 3-х мерном пространстве. Чем больше ширина эллипса вдоль направления, тем больше дисперсия вектора по этому направлению.

Уменьшение размерности достигается за счет метрического анализа пространства векторов-параметров путем отбора релевантных строк (с малой дисперсией), замены мультикоррелирующих строк на их линейную композицию с помощью обобщения алгоритма QPFS, изучения структуры сообществ строк.

В качестве базовых моделей используются SSA ([Golyandina et al., 2001]), нелинейный PCA, RNN ([Bronstein et al., 2021]), VAE ([Kingma et al., 2019]) и Neural ODE ([Chen et al., 2018]).

Задача восстановления временного ряда решается на синтетических данных зашумленного sin, данных показания акселерометра в датасете MotionSense3 [Malekzadeh et al., 2018], периодичных видеоданных.

2 Problem statement

Пусть имеется множество из m временных рядов $\mathcal{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_m\}, \mathbf{s}_i = [\mathbf{s}_i^1, \ldots, \mathbf{s}_i^T], \mathbf{s}_i^j \in \mathbb{R}$, где n - длина сигналов. Каждый временной ряд - последовательность измерений величины в течение времени.

Definition 1. *Временное представление* $\mathbf{x}_t = [\mathbf{s}_1^t, \ldots, \mathbf{s}_m^t]^{\mathsf{T}} \in \mathbb{R}^m$ *состоит из измерений временных рядов в момент времени* t.

Definition 2. *Представление предыстории длины* h *для момента времени* t *множества временных рядов* $\mathcal{S}$ *- это матрица* $\mathbf{X}_{t,h} = [\mathbf{x}_{t-h+1}, \ldots, \mathbf{x}_t]^{\mathsf{T}} \in \mathbb{R}^{h \times m}$.

Definition 3. *Представление горизонта прогнозирования длины* p *для момента времени* t *множества временных рядов* $\mathcal{S}$ *- это матрица* $\mathbf{Y}_{t,p} = [\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+p}]^{\mathsf{T}}$.

Definition 4. *Прогностическая модель* $\mathbf{f}^{\mathsf{AR}} : \mathbb{R}^{h \times m} \to \mathbb{R}^{p \times m}$ *является авторегрессионной моделью, которая по представлению предыстории* $\mathbf{X}_{t,h}$ *предсказывает представление горизонта планирования* $\mathbf{Y}_{t,p}$.

Решается задача авторегрессионного декодирования. Она состоит в построении прогностической модели $\mathbf{f}^{\mathsf{AR}}$, дающий *представление горизонта прогнозирования* множества временных рядов по *представлению предыстории* того же множества рядов. В дальнейшем будем считать, что восстанавливаем 1 временной ряд, что есть что $m = 1$.

Обозачим множество всех одномерных временных рядов через $\mathbb{S} : \mathbb{S} = \bigcup\limits_{n=1}^{+\infty} \{[s_1, \ldots, s_n] \in \mathbb{R}^n\}$. Тогда прогностическая модель это функция $f^{\mathsf{AR}} : \mathbb{S} \to \mathbb{R}$. Изначальный временной ряд $\mathbf{s} = [s_1, \ldots, s_T]$ делится на две части $\mathbf{s} = [\mathbf{s}^H | \mathbf{s}^T], \mathbf{s}^H = [s_1, \ldots, s_h], \mathbf{s}^T = [s_{h+1}, \ldots, s_T]$. Задача состоит в том, чтобы предсказать $\mathbf{s}^T$ с максимальной точностью. Предсказание происходит следующим образом:

1. С помощью модели f предсказывается $\hat{s}_{h+1}$.
2. Предсказанный элемент $\hat{s}_{h+1}$ вместе с исходным временным рядом $\mathbf{s}^H$ подаются на вход f для предсказания $\hat{s}_{h+2}$.
3. Шаги $1 - 2$ повторяются, пока не будет предсказан весь $\hat{\mathbf{s}}^T = [\hat{s}_{h+1}, \ldots, \hat{s}_T]$.

Для упрощения нотации обозначим $f(\mathbf{s}^H) = \hat{\mathbf{s}}^T = [\hat{s}_{t+1}, \ldots, \hat{s}_T]$.

Модель $f = f(\mathbf{w}, \mathbf{s}), \mathbf{w} \in \mathbb{W}, \mathbf{s} = [s_1, \ldots, s_t] \in \mathbb{R}^t$ выбирается из некоего параметрического семейства. Параметры модели выбираются таким образом, чтобы минимизировать функцию ошибки $S = S(\mathbf{w}|\mathbf{s}, f)$:

$$\mathbf{w}^* = \underset{\mathbf{w}\in\mathbb{W}}{\arg\min}\, S(\mathbf{w}|\mathbf{s}, f).$$

В работе будет использоваться функция ошибки MSE, то есть

$$S(\mathbf{w}|\mathbf{s}, f) = \sum_{t=h+1}^{T} (\mathbf{s}_t - \hat{\mathbf{s}}_t)^2.$$

.

В качестве моделей для восстановления временного ряда будем использовать SSA, двуслойную нейросеть с ортогональными линейными преобразованиями, RNN и Neural ODE. Остановимся на каждой из них подробнее.

В модели SSA восстановление временного ряда получается за счет разложения матрицы фазовой траектории в сумму одноранговых матриц. По временному ряду $\mathbf{s} = [s_1, \dots, s_T]$ и ширине окна h строится матрица фазовых траекторий

$$\mathbf{X} = \begin{pmatrix} s_1 & \dots & s_k \\ s_2 & \dots & s_{k+1} \\ \dots & \dots & \dots \\ s_h & \dots & s_T \end{pmatrix} = \big[X_1 : \dots : X_k\big], k = T - h + 1, X_i = [s_i, s_{i+1}, \dots, s_{i+h-1}] \in \mathbb{R}^h,$$

которая является матрицей Ганкеля, так как каждая диагональ вида $i + j = const$ содержит одинаковые элементы. Векторы X_i являются точками фазовой траектории сигнала. Далее вычисляются собственные значения и соответствующие собственным подпространствам ортонормированные системы векторов матрицы $\mathbf{S} = XX^T \in \mathbb{R}^{h\times h}$. Пусть $\lambda_1 \geqslant \ldots \geqslant \lambda_L \geqslant 0$ - собственные значения $\mathbf{S}$, $u_1, \dots u_h$ - ортонормированная система собственных векторов, соответствующая собственным значениям, $v_i = \frac{\mathbf{X}^T u_i}{\sqrt{\lambda_i}} (i = 1, \dots, h)$. Тогда $\mathbf{X} = \mathbf{X}_1 + \dots + \mathbf{X}_d, \mathbf{X}_i = \sqrt{\lambda_i} u_i v_i^T$ - SVD-разложение $\mathbf{X}$. Далее происходит группировка матриц $\mathbf{X}_i$, их ганкелизация и восстанавливается матрица сигнала $\hat{\mathbf{X}} = \tilde{\mathbf{X}}_{I_1} + \dots + \tilde{\mathbf{X}}_{I_m}, I_1 \sqcup \dots \sqcup I_m \subset \{1, \dots, h\}, \tilde{\mathbf{X}}_I = hank\big(\sum\limits_{i\in I} \mathbf{X}_i\big)$. Где $hank(X)$ - ганкелизация матрицы X, состоящая в том, что на каждой диагонали вида $i + j = const$ все элементы заменяются на их среднее арифметическое. Параметрами модели SSA являются множества $I_1, \dots, I_m$.

Модель двуслойной нейросети задается следующим образом: $f(x) = \sigma(w \cdot \sigma(Wx)), x \in \mathbb{R}^h, W \in \mathbb{R}^{d \times h}, w \in \mathbb{R}^d : w^T w = 1, W^T W = I$. Последние два условия гарантируют, что преобразования будет ортогональным. Здесь нейросеть предсказывает значение сигнала в следующий момент времени на основе его значения в предыдущие t моментов, где t - ширина окна. Нелинейность обеспечивается выбором функции активации $\sigma(x)$.

Модель RNN задается следующим образом:

$h_t = \sigma(W \cdot h_{t-1} + V \cdot \mathbf{x}_t)$

$s_{t+1} = tanh(w_o^\mathsf{T} \cdot h_t)$,

где $h_t \in \mathbb{R}^d$ - скрытое состояние RNN в момент времени t, $W \in \mathbb{R}^{d \times d}, V \in \mathbb{R}^{d \times h}$. $\mathbf{x}_t = [s_{t-h+1}, \dots, s_t] \in \mathbb{R}^h$ - временной ряд, подающийся на вход модели, $w_o \in \mathbb{R}^d, s_{t+1} \in \mathbb{R}$ - прогноз значения сигнала в момент времени $t+1$. Параметрами модели являются матрицы W, V, w_0 а также начальное скрытое состояние h_0, которое мы зафиксируем $h_0 = 0$.

О входном временном ряде выдвинута гипотеза о том, что точки на фазовой траектории распределены по нормальному закону, то есть что $\mathbf{x}_t = [s_t, \dots, s_{t+h-1}] \sim \mathcal{N}(\hat{\mathbf{x}}_t, \mathbf{B})$, где $\hat{\mathbf{x}}_t$ - это матожидание точки фазовой траектории в момент времени t, а $\mathbf{B}$ - матрица ковариации. Мы также предполагаем, что $\mathbf{w} \sim \mathcal{N}(\hat{\mathbf{w}}, \mathbf{A})$.

В работе оцениваются матожидание вектора параметров модели $\hat{\mathbf{w}}$, а также матрица ковариации параметров $\mathbf{A}$ с помощью методов бутстрепа и вариационного вывода.

Для метрического анализа пространства параметров выбираются набор множеств индексов $\mathcal{I}_1 \sqcup \dots \sqcup \mathcal{I}_k \subset \{1, \dots, len(\mathbf{w})\}$, рассматриваются соответствующие им подвектора параметров $\mathbf{w}_{\mathcal{I}_1}, \dots, \mathbf{w}_{\mathcal{I}_k}$ Считается, что каждый $\mathbf{w}_{\mathcal{I}_j}$ соответствует «смысловой единице» модели. В нейросетевых моделях это строки матрицы линейного преобразования пространства параметров W, которые обычно называют *нейронами.*

3 *

Список литературы

Weikuan Jia, Meili Sun, Jian Lian, and Sujuan Hou. Feature dimensionality reduction: a review. *Complex & Intelligent Systems*, 8(3):2663–2693, 2022.

Cem Örnek and Elif Vural. Nonlinear supervised dimensionality reduction via smooth regular embeddings. *Pattern Recognition*, 87:55–66, 2019.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

RV Isachenko and VV Strijov. Quadratic programming feature selection for multicorrelated signal decoding with partial least squares. *Expert Systems with Applications*, 207:117967, 2022.

Babak Hassibi, David G Stork, and Gregory J Wolff. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, pages 293–299. IEEE, 1993.

Xin Dong, Shangyu Chen, and Sinno Pan. Learning to prune deep neural networks via layer-wise optimal brain surgeon. *Advances in Neural Information Processing Systems*, 30, 2017.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Nina Golyandina, Vladimir Nekrutkin, and Anatoly A Zhigljavsky. *Analysis of time series structure: SSA and related techniques*. CRC press, 2001.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, pages 89–95, 2021.

Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. Protecting sensory data against sensitive inferences. In *Proceedings of the 1st Workshop on Privacy by Design in Distributed Systems*, pages 1–6, 2018.