Метрический анализ пространства параметров глубоких нейросетей.

Насыров Р.Р.

МФТИ

6 сентября 2023 г.

Введение

- Высокоразмерные данные - видео, звук - избыточны.
- Модели тяжело обучаются на избыточных данных, часто переобучаются.
- Нужно бороться с избыточностью и переобучением.

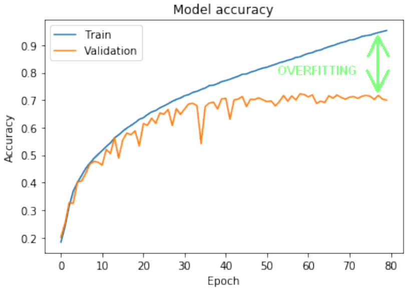

Рис.: Переобучение.

Методы улучшения обучения

- Методы снижения размерности входных данных:
  - PCA
  - Quadratic Programming Feature selection
  - Neural Autoencoders
- Выбор оптимальной структуры модели:
  - Optimal Brain Surgeon
  - Correlational analysis
  - Weights freezing

Цель исследования

Цель исследования: создать метод снижения размерности пространства параметров глубоких нейросетей.
Задача: построить алгоритм изменения нейронов, который

- Оценивает дисперсию нейронов
- Определяет релевантные нейроны, фильтрует выбросы
- Выделяет сообщества нейронов на основе дисперсий
- Находит центры сообществ и заменяет веса нейронов сообщества на вес центра сообщества

Отбор нейронов сети метрическими методами.

Модель двуслойной нейросети:

$$f(x) = \sigma(W_1^T \cdot \sigma(W_2^T x + b_1) + b_2)$$

Модель RNN:

$$h_t = \sigma(W \cdot h_{t-1} + V \cdot x_t),$$

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

Рис.: Схема экспериментов.

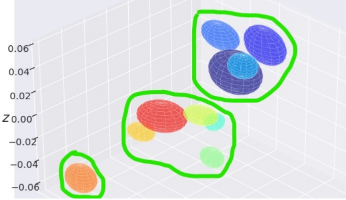

Рис.: Доверительные области нейронов сети 10-3-20.

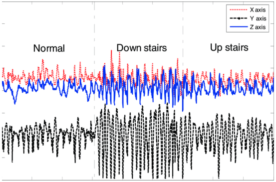

Рис.: Показания акселерометра ходьбы.

Литература

- Weikuan Jia, Meili Sun, Jian Lian, and Sujuan Hou. Feature dimensionality reduction: a review. Complex Intelligent Systems, 8(3):2663–2693, 2022.
- Cem Ornek and Elif Vural. Nonlinear supervised dimensionality reduction via smooth regular embeddings. Pattern Recognition, 87:55–66, 2019.
- RV Isachenko and VV Strijov. Quadratic programming feature selection for multicorrelated282 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.
- Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. Journal of neural engineering, 15(5):056013, 2018.

Постановка задачи

- Решается задача авторегрессионного декодирования.
- Обозначим множество всех одномерных временных рядов через $\mathbb{S}$:

$$\mathbb{S} = \bigcup_{n=1}^{+\infty} \{[s_1, \dots, s_n] \in \mathbb{R}^n\}.$$

- Авторегрессионная модель $\mathrm{f}^{\mathrm{AR}} : \mathbb{S} \to \mathbb{S}$ восстанавливает значения временного ряда.
- Рассматриваются feed-forard нейросети:

$$\mathrm{f}(\mathrm{x}) = \sigma(\mathrm{W}_1^\mathsf{T} \cdot \sigma(\mathrm{W}_2^\mathsf{T} \cdot \dots (\mathrm{W}_n^\mathsf{T} \cdot \mathrm{x} + \mathrm{b}_n) \dots + \mathrm{b}_2) + \mathrm{b}_1)$$

- Гипотеза о порождении данных: $\mathrm{x}_t = [s_t, \dots, s_{t+h-1}] \sim \mathcal{N}(\hat{\mathrm{x}}_t, \mathrm{B})$, где $\hat{\mathrm{x}}_t$ — это матожидание точки фазовой траектории в момент времени t, а B — матрица ковариации.
- Гипотеза о распределении параметров модели:

$$\mathrm{W} = \begin{pmatrix} \mathrm{w}_1^T \\ \dots \\ \mathrm{w}_n^T \end{pmatrix} : \mathrm{w}_i \sim \mathcal{N}(\mu_i, \mathrm{A}_i)$$

Постановка задачи

- Для каждой матрицы W_i требуется построить матрицу $\widetilde{\mathrm{W}_i}$ меньших размеров, сохраняя точность восстановления ряда на приемлемом уровне.
- Модель $\mathrm{f} = \mathrm{f}(\mathrm{w}, \mathrm{s}), \mathrm{w} \in \mathbb{W}$ выбирается из некоего параметрического семейства нейросетей. Параметры модели выбираются таким образом, чтобы минимизировать функцию ошибки $S = S(\mathrm{w}|\mathrm{s}, f)$:

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

В работе будет использоваться функция ошибки MSE, то есть

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

.

Решение задачи, теоретическая часть

- Задан временной ряд $\mathrm{s} = (s_1, \dots, s_n) \in \mathbb{R}^n$.
- Задана модель учителя в виде суперпозиций линейных и нелинейных преобразований:

$$\mathrm{f} = \sigma \circ \mathrm{U}_T \circ \sigma \circ \mathrm{U}_{T-1} \circ \dots \circ \mathrm{U}_2 \circ \sigma \circ \mathrm{U}_1,$$

где T — число слоев модели учителя (от 5 до 8 у применяемых далее моделей), σ — функция активации, U_t — матрица линейного преобразования. Матрицы U_t, параметры модели учителя, соединяются в вектор параметров u модели учителя f :

$$\mathrm{u} = vec([\mathrm{U}_T, \mathrm{U}_{T-1}, \dots, \mathrm{U}_1]).$$

Решение задачи, теоретическая часть

- Мы работаем в предположении, что параметры учителя распределены нормально. Для модели учителя оценивается апостериорное распределение вектора параметров

$$p(\mathrm{u}) = p(\mathcal{N}(\overline{\mathrm{u}}_{ps}, \mathrm{A}_{ps}^{-1})).$$

- В качестве модели ученика используется байесовская нейронная сеть, задаваемая аналогично модели учителя:

$$\mathrm{g} = \sigma \circ \mathrm{W}_L \circ \sigma \circ \mathrm{W}_{L-1} \circ \cdots \circ \mathrm{W}_2 \circ \sigma \circ \mathrm{W}_1.$$

Мы считаем, что

$$\mathrm{w} \sim \mathrm{N}(\overline{\mathrm{w}}_t, \mathrm{B}^{-1}).$$

Решение задачи, теоретическая часть

Algorithm 1 Алгоритм снижения размерности

1. Зафиксировать структуру модели ученика g и учителя f, то есть параметры T, L и размеры матриц $\mathrm{U}_t, \mathrm{W}_t$.
2. Обучить модель учителя f.
3. Оценить апостериорное распределение параметров учителя $p_{ps}(\mathrm{u})$, то есть параметры $\overline{\mathrm{u}}, \mathrm{A}^{-1}$.
4. Назначить параметрам ученика априорное распределение $p(\mathrm{w}) \sim p_{ps}(\mathrm{u})$.
5. Обучить байесовскую сеть ученика на ответах учителя, получив апостериорное распределение параметров ученика $p(\mathrm{w}|\mathcal{D}) = p(\mathrm{N}(\overline{\mathrm{w}}_t, \mathrm{B}^{-1}))$.
6. Провести метрический анализ пространства параметров ученика и снизить его размерность.

Решение задачи, теоретическая часть

- Обучение байесовской модели ученика производится при помощи вариационного вывода на основе совместного правдоподобия данных:

$$\mathcal{L}(\mathcal{D}) = \log p(\mathcal{D}) = \log \int\limits_{\mathrm{w} \in \mathbb{R}^k} p(\mathcal{D}|\mathrm{w}) \cdot p(\mathrm{w}) d\mathrm{w},$$

  где $p(\mathrm{w})$ - априорное распределение параметров модели ученика, которое получается преобразованием из апостериорного распределения параметров учителя.

- Обучение ученика сводится к решению задачи:

$$\widehat{\mathrm{w}} = \arg \min_{\mu, {}^\circ, \mathrm{w}} \mathrm{D_{KL}}(p(\mathrm{w}|\mathcal{D}) || p(\mathrm{w})) - \log p(\mathrm{y}|\mathrm{X}, \mathrm{w}).$$

  Где второе слагаемое - логарифм правдоподобия выборки, а первое - KL-дивергенция между априорным и апостериорным распределением параметров ученика.

Решение задачи, теоретическая часть

Выделение структуры сообществ проводится с помощью алгоритма «k-means clustering»:

- Дан набор наблюдений $(\mathrm{x}_1, \ldots, \mathrm{x}_n), \mathrm{x}_i \in \mathbb{R}^d$.
- Алгоритм «k-means» делит наблюдения на k классов $\mathcal{S} = (S_1, \ldots, S_k)$, минимизируя внутриклассовую дисперсию:

$$\sum_{i=1}^{k} \sum_{\mathrm{x} \in S_i} ||\mathrm{x} - \mu_i||^2 \to \min_{\mathcal{S}}.$$

- Где внутриклассовое среднее

$$\mu_i = \frac{1}{|S_i|} \sum_{\mathrm{x} \in S_i} \mathrm{x}$$

Вычислительный эксперимент

Цель эксперимента: визуализировать пространство нейронов сети в задаче восстановления временного ряда.

Данные:

- Показания акселерометра, записанные с частотой 50ГЦ во время 6 видов активностей у 20 людей в течение 2 минут.

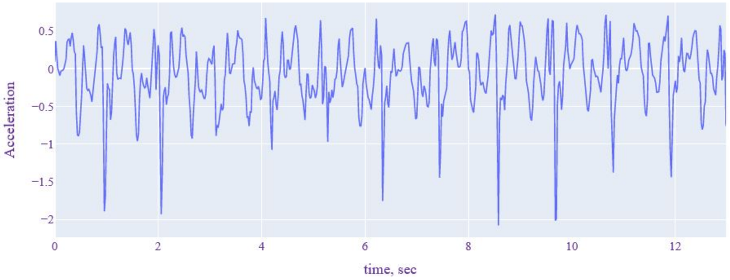

Рис.: Данные акселерометра.

Способы вычисления ковариации параметров нейрона

- Bootstrap: обучаются N одинаковых моделей на случайной подвыборке тренировочных данных. Ковариация нейрона $\mathrm{w} \in \mathbb{R}^n$ считается по формуле:

$$cov(\mathrm{w}_i, \mathrm{w}_j) = \frac{1}{N-1} \sum_{t=1}^{N} (\mathrm{w}_i^t - \overline{\mathrm{w}_i})(\mathrm{w}_j^t - \overline{\mathrm{w}_j})$$

- Bayes NN: обучается байесовская нейросеть, в которой нейроны $\mathrm{w} \in \mathrm{R}^n$ параметризуются с помощью матрицы $A \in \mathbb{R}^{n \times k}, rk(A) = k, AA^T = cov(\mathrm{w})$ и вектора-матожидания $\overline{\mathrm{w}}$.
- Hessian: по обученной модели оцениваем гессиан и градиент ошибки по параметрам модели. По ним вычисляем матрицу:

$$H_n = \mathbb{E}_{\xi \sim \mathcal{D}}[\nabla^2 S(\mathrm{w}^*, \xi)], \;\; G_n = \mathbb{E}_{\xi \sim \mathcal{D}}\left[\nabla \mathrm{S}(\mathrm{w}^*, \xi) \cdot \nabla \mathrm{S}(\mathrm{w}^*, \xi)^{\mathrm{T}}\right]$$

Оценка: $cov(\sqrt{n}\overline{\mathrm{w}}_n) \approx H_n^{-1} G_n H_n^{-1}$.

Визуализация пространства нейронов 3 методами

Визуализация доверительных интервалов первых 16 нейронов нейросети архитектуры 100 — 3 — 100, обученной на датасете MotionSense.

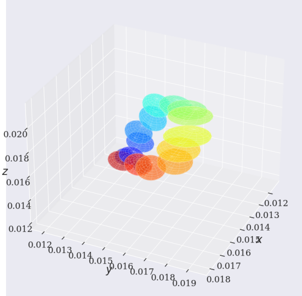

Рис.: Bootstrap

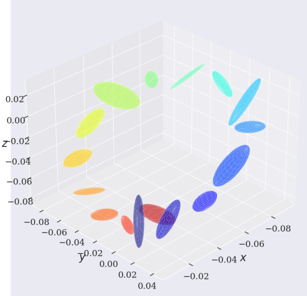

Рис.: Bayes NN

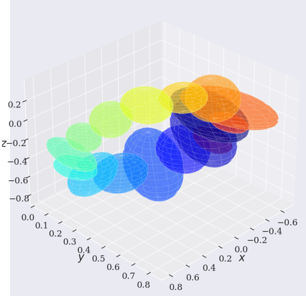

Рис.: Hessian

- нейроны расположены по кругу.
- возможно снижение размерности пространства с 100 до 16.

Снижение размерности пространства параметров с помощью кластеризации методом «k-means».

- На 3 датасетах: MNIST, Iris flower и Motion Sence решались задачи классификации и регрессии с помощью полносвязных нейронных сетей.
- Параметры распределения оцениваются с помощью bootstrap.
- Кластеризация нейронов проводилась с помощью алгоритма $k - means$ двумя способами: в первом расстояние между нейронами считалось, как евклидово расстояние между их средними, а во втором как расстояние Вассерштайна между двумя нормальными распределениями:

$$\rho_1(\mathrm{w}_1, \mathrm{w}_2) = ||\mathrm{m}_1 - \mathrm{m}_2||_2^2$$

$$\rho_2(\mathrm{w}_1, \mathrm{w}_2) = ||\mathrm{m}_1 - \mathrm{m}_2||_2^2 + Tr(\mathrm{A}_1 + \mathrm{A}_2 - 2(\mathrm{A}_1^{0.5}\mathrm{A}_2\mathrm{A}_1^{0.5})^{0.5})$$

Точности модели VS количество параметров.

На графиках показана зависимость точности модели от ее сложности. Красная кривая - кластеризация по 1 метрике, синяя - по 2-й. В скобках подписана размерность нейронов.

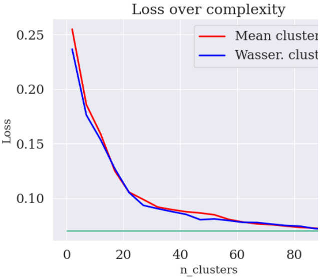

Рис.: MotionSense (10).

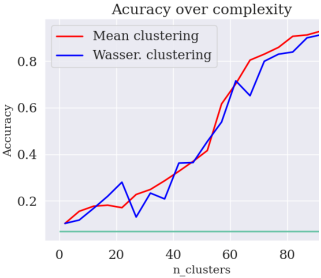

Рис.: MNIST (8).

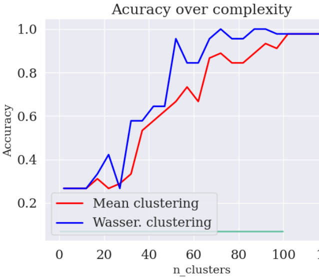

Рис.: IRIS (4).

- Учет коварицаии улучшает качество модели при небольшой размерности нейронов.
- На датасете IRIS удается сократить в 2 раза количество параметров, сохраняя точность более 90%.

Выводы

- При восстановлении временных рядов нейроны имеют регулярную структуру, что допускает значительное сжатие количества параметров.
- Разные методы оценки параметров распределения дают разные численные результаты, но схожие по структуре (по «кругу»).
- Кластеризация по метрике Вассерштайна работает не хуже, а иногда даже значительно лучше, чем по обычной метрике.
- Это подтверждает актуальность задачи оценки ковариации нейронов и рассмотрения нейрона как одного целого.

Следующие шаги

- Сжатие больших нейросетей.
- Разработка методов эффективной оценки матрицы коварации параметров с помощью низкоранговых байесовских сетей, полубайесовских и блочно-диагонально-байесовских.
- Сравнение с известными методами снижения пространства параметров: OBS, OBD и других.