Аддитивная регуляризация и ее метапараметры при выборе структуры сетей глубокого обучения*

М. С. Потанин[1], К. О. Вайсер[2], В. В. Стрижов[3]

Аннотация

Решается задача выбора модели глубокого обучения. Выбор оптимизации функции ошибки. Включение аддитивной регуляризации. Сеть глубокого обучения представима в виде суперпозиции автокодировщика и нейронной сети. Под сложностью модели понимается мера множества допустимых значений параметров. Функция аддитивной регуляризации это линейная комбинация экпертно заданных регуляризаторов. Веса слагаемых есть метапараметры. Исследуются свойства алгоритма оптимизации метапараметров аддитивной регуляризации. Исследуются зависимости точности, сложности и устойчивости модели от метапарметров аддитивной регуляризации.

Ключевые слова: автокодировщик; нейронные сети; структура; аддитивная регуляризация.

Введение

В данной работе рассматривается влияние способа построения функции ошибки на выбор структуры сети глубокого обучения. Функция ошибки —это оценка качества модели.

*Работа выполнена при поддержке РФФИ (проекты 19-07-1155, 19-07-0885) и правительства РФ (соглашение 05.Y09.21.0018). Настоящая статья содержит результаты проекта «Статистические методы машинного обучения», выполняемого В рамках реализации Программы Центра компетенций Национальной технологической инициативы «Центр хранения и анализа больших данных», поддерживаемого Министерством науки и высшего образования Российской Федерации по Договору МГУ им. М. В. Ломоносова с Фондом поддержки проектов Национальной технологической инициативы от 11.12.2018 № 13/1251/2018.

[1]Московский физико-технический институт, mark.potanin@phystech.edu

[2]Московский физико-технический институт, vajser.ko@phystech.edu

[3]Вычислительный центр имени А. А. Дородницына Федерального исследовательского центра «Информатика и управление» Российской академии наук; Московский физико-технический институт, strijov@phystech.edu

Сеть оптимальной структуры как можно более точно аппроксимирует исходную неизвестную зависимость целевой переменной y от векторов признакового пространства $\mathbf{x}$. Моделью называется отображение $f : (\mathbf{x}, \mathbf{w}) \mapsto y$. В общем виде модель глубокого обучения представляет собой суперпозицию обобщенных линейных моделей и выглядит как (6). Такая модель может содержать большое число слоев и нейронов. Под структурной сложностью модели понимается число параметров модели. Предполагается, что чем сложнее модель, тем выше у нее точность аппроксимациии, то есть тем меньше значение функции ошибки. Однако увеличение сложности модели приводит к снижению ее устойчивости, то есть зависимости результата от изменения начальных данных (сюда ссылку про это). Предлагается разработать метод, позволяющий понизить сложность модели при сохранении ее точности. В работе –(прошлая статья)– поиск структуры работал через добавление или удаление элементов из структуры генетическим алгоритмом. Это решение по сути являлось разновидностью полного перебора. В этой работе также предлагается использовать аддитивную регуляризацию для выбора оптимальной структуры.

Задачи, решение которых не существует, не единственно или не устойчиво, принято называть *некорректно поставленными по Адамару* [1].

Определение 1 *Регуляризация —метод решения задач, в котором для выбора оптимального решения задаются дополнительные критерии оптимальности, учитывающие специфические требования решаемой задачи и называемые регуляризаторами.*

Определение 2 *Аддитивная регуляризация —это вид регуляризации, основанный на оптимизации взвешенной суммы регуляризаторов.*

Регуляризация повышает устойчивость прогноза в случае мультикореллированости параметров модели или переменных признакового пространства. Она способствует повышению обобщающей способности модели и снижению риска переобучения [2]. Регуляризация предотвращает ситуацию, когда параметры становятся константными и не изменяются в процессе оптимизации модели.

В качестве объекта исследования выступает способ построения функции ошибки. Аддитивная регуляризация имеет вид (7). Исследуется влияние весов регуляризаторов (1) на сложность и точность модели. Аддитивная регуляризация записывается как

$$\sum_{i=1}^{r} \lambda_i^T \mathcal{R}_i, \tag{1}$$

где $\mathcal{R}_i$ —регуляризатор.

Определение 3 *Метапараметры —веса регуляризаторов, используемые для оптимизации параметров модели.*

Веса регуляризаторов в данной работе изменяются в ходе процедуры оптимизации. Это изменение называется расписанием оптимизацию. Для этого составляется расписание

оптимизации метапараметров λ аддитивной регуляризации. В отличие от регуляризации для линейных моделей, метапарамеры аддитивной регуляризации назначаются для каждого из слоев нейронной сети.

Например, элемент функции ошибки с аддитивной регуляризацией может иметь следующую структуру:

$$\lambda \|L\mathbf{w}\|_2^2,$$

где λ — метапараметр функции ошибки, L — гиперпараметр модели, $\mathbf{w}$ — параметр модели.

Определение 4 *Гиперпараметры —параметры функции аддитивной регуляризации, используемые для оптимизации параметров модели и метапараметров. В этой работе гиперпараметры считаются экспертно заданными, неизменяемыми параметрами. Они являются элементами функции ошибки.*

Варьирование значений метапараметров добавляет и удаляет из рассмотрения различные регуляризаторы. В функцию ошибки добавляются индивидуальные регуляризаторы для каждого слоя нейросети.

Алгоритм оптимизации параметров модели состоит следующих шагов — 1) экспертное задание гиперпараметров модели, 2) оптимизация параметров модели , 3) оптимизация метапараметров модели. Параметры модели оптимизируются алгоритмом обратного распространения ошибки. Оптимизация метапараметров происходит с помощью генетического алгоритма.

Обзор литературы

Регуляризация L_2 работает с помощью установления баланса между смещением и дисперсией. Но ее недостаток в том, что она не может создать разреженную модель, так как сохраняет все множество параметров. Другой вид регуляризации $L1$ был предложен в [3], и он предлагает автоматический выбор параметров. Но он также имеет несколько недостатков, среди которых:

1) Если обозначить за p число независимых переменных, а за n количество объектов в выборке, то в случае $p > n$ lasso регуляризация выбирает максимально n независимых переменных из множества.

2) При наличии групп сильно скоррелированных переменных, lasso регуляризация выбирает только одну переменную из группы, причем не обращая внимания какую именно.

3) В случае $n > p$ и наличии высокой корреляции между переменными было эмпирически показано, что ridge регрессия работает намного лучше lasso.

Таким образом, пукнты 1) и 2) делают lasso неприменимой техникой в некоторых задачах, где требуется отбор признаков. Перечисленные проблемы решаются с помощью другой техники регуляризации elastic net [4], которая позволяет производить автоматический отбор переменных, регулировать их веса, а так же выбирать группы коррелярующих признаков. Метод регуляризации elastic net представляет собой добавление в функцию ошибки двух дополнительных слагаемых

$$S(\lambda_1, \lambda_2, \lambda_3, \mathbf{w}) = \lambda_1|\mathbf{y} - \mathbf{X}\mathbf{w}|^2 + \lambda_2|\mathbf{w}|^2 + \lambda_3|\mathbf{w}|_1$$

Обобщением elastic net является регуляризация Support Features Machine. Функция ошибки задается в виде

$$\min_{\mathbf{w},\mathbf{w_0}} \quad L(\mathbf{w}, \mathbf{w_0})C\sum_{i=1}^{l}(1 - M_i(\mathbf{w}, \mathbf{w_0}))_+ + \sum_{i=1}^{n} R_\mu(w_j), \tag{2}$$

$$R_\mu(w_j) = \begin{cases} 2\mu|w_j|, & |w_j| \le \mu, \\ \mu^2 + w_j^2, & |w_j| \ge \mu. \end{cases}$$

Признаки отбираются с помощью параметра селективности μ. Шумовые признаки ($|w_j| < \mu$) подавляются как и в Lasso, а значимые зависимые признаки группируются также как и в elastic net. На нее похож такой метод как Relevance Features Machine. Задается в виде

$$\min_{\mathbf{w},\mathbf{w_0}} \quad C\sum_{i=1}^{l}(1 - M_i(\mathbf{w}, \mathbf{w_0}))_+ + \sum_{i=1}^{n} \ln(w_j^2 + \frac{1}{\mu}) \tag{3}$$

Отличие (3) от метода (2) в том, что происходит более совершенный отбор признаков, когда они только совместно обеспечивают хорошее решение.

В общем случае, если функция L — выпуклая, то можно воспользоваться регуляризацией Moreau-Yosida. Записывается $\lambda > 0$

$$M_{\lambda L}(x) = \inf_u (L(u) + \frac{1}{2\lambda}||x - u||_2^2)$$

У этой функции ряд замечательных свойств —1) $M_{\lambda f}(x)$ — выпуклая функция в силу инфимальной конволюции; 2) множество точек минимума для L и $M_{\lambda L}(x)$ совпадают; 3) $M_{\lambda L}(x)$ — гладкая функция в силу сильной выпуклости ее первой сопряженной функции и совпадении со второй сопряженной функцие $M_{\lambda L}(x) = M_{\lambda L}^{**}(x)$.

В статье [5] представлены несколько стратегий стабилизации оптимизации структуры очень глубоких сетей. Они основаны на системах обыкновенных дифференциальных уравнений (ODE). Так же в этой статье вводят регуляризацию, чувствительную к плавному изменению параметров модели между соседними слоями. Кроме того, как показано в численных экспериментах, регуляризация повышает устойчивость модели.

Для практического использования аддитивной регуляризации важно, чтобы помимо высокой точности аппроксимации целевой функции модель являлась бы интерпретируемой. Например, исследования показали, что в области медицины предпочитаются модели основанные на деревьях решений, из-за того, что можно проследить,на основании каких правил строился прогноз. Тем не менее, возможность узнать причины предсказаний модели не являтся основным критерием применимости модели. Если эти причины расходятся со здравым смыслом и экспертными правилами данной области, то такая модель не является приемлемой. Эти два критерия называются интерпретируемостью и точностью аппроксимации модели.

В работе [6] авторы рассматривают новый способ регуляризации EYE (expert yielded estimates), который включает в себя экспертные знания об отношениях между признаками и зависимой переменной. Авторы расматривают задачу минимизации эмпирического риска

$$\hat{\mathbf{W}} = \arg\min \mathrm{S}(\mathbf{W}, \mathbf{X}, \mathbf{y}) + n\lambda\mathcal{J}(\mathbf{W}, \mathbf{\Gamma})$$

в которой минимизируется сумма функции ошибки и регуляризации $\mathcal{J}$. Имеется множества признаков $\mathcal{D}$, из которых для множества $\mathcal{K} \subseteq \mathcal{D}$ имеется дополнительная информация о том, что эти признаки являются важными в рассматриваемой экспертной области. Следовательно для $\hat{\mathbf{W}}_{\mathcal{D}\setminus\mathcal{K}}$ требуется разреженность, а для $\hat{\mathbf{W}}_{\mathcal{K}}$ —нет. Базовый подход, используемый авторами, заключается в использовании $L1$ и $L2$ регуляризаций, тогда регуляризационный член имеет вид

$$\mathcal{J} = (1-\beta)||\mathbf{\Gamma} \odot \mathbf{W}||_2^2 + \beta||(1-\mathbf{\Gamma}) \odot \mathbf{W}||_1,$$

где параметр β контролирует баланс между признаками из $\mathcal{K}$ и $\mathcal{D} \setminus \mathcal{K}$. Предлагаемое авторами решение имеет следующий вид

$$\mathcal{J} = ||(1-\mathbf{\Gamma}) \odot \mathbf{W}||_1 + \sqrt{||(1-\mathbf{\Gamma}) \odot \mathbf{W}||_1^2 + ||\mathbf{\Gamma} \odot \mathbf{W}||_2^2}. \tag{4}$$

Его использование в задачах стратификации риска пациентов позволило получить модели, в которых основные используемые признаки сильно пересекались с факторами, которые считаются значимыми в медицинской среде. При э том удалось сохранить высокое качество предсказания.

Постановка задачи

Мотивация Для решения задачи выбора модели предлагается построить модифицированную функцию ошибки. Требуется исследовать способ построить вид функции ошибки аддитивной регуляризации и ее влияние на точность и сложность модели глубокого обучения.

Выбор модели Задана выборка, конечное множество пар

$$(\mathbf{x}, y) \in D, \quad \mathbf{x} \in \mathbb{R}^n, \quad y \in \mathbb{R}, \tag{5}$$

где $\mathbf{x}$ — вектор независимых переменных, y — зависимая переменная. Моделью называется отображение $f : (\mathbf{x}, \mathbf{w}) \mapsto y$. Требуется построить аппроксимирующую модель $f(\mathbf{x})$ вида:

$$f = \sigma_k \circ \underset{1\times 1}{\mathbf{w}_k^{\mathsf{T}}}\boldsymbol{\sigma}_{k-1} \circ \mathbf{W}_{k-1}\boldsymbol{\sigma}_{k-2} \circ \cdots \circ \underset{n_2\times 1}{\mathbf{W}_2}\boldsymbol{\sigma}_1 \circ \underset{n_1\times n}{\mathbf{W}_1}\underset{n\times 1}{\mathbf{x}}. \tag{6}$$

Эта модель рассматривается как суперпозиция линейной модели, глубокой нейросети и автоэкнодера.

Структура сети. В работе ==[прошлая статья]== рассматривается задача оптимизации структуры сети генетическим алгоритмом. Предлагается использовать тот же подход и в текущей статье с добавлением аддитивной регуляризации.

Решается задача выбора оптимальной структуры модели

$$f = \sigma_k \circ \mathbf{\Gamma}_k \otimes \underset{1\times 1}{\mathbf{w}_k^{\mathsf{T}}}\boldsymbol{\sigma}_{k-1} \circ \mathbf{\Gamma}_{k-1} \otimes \mathbf{W}_{k-1}\boldsymbol{\sigma}_{k-2} \circ \cdots \circ \mathbf{\Gamma}_2 \otimes \underset{n_2\times 1}{\mathbf{W}_2}\boldsymbol{\sigma}_1 \circ \mathbf{\Gamma}_1 \otimes \underset{n_1\times n}{\mathbf{W}_1}\underset{n\times 1}{\mathbf{x}},$$

где $\mathbf{\Gamma}$ — матрица, задающая структуру модели; $\otimes$ — адамарово произведение, определяющееся как поэлементное умножение. Если элемент $\gamma \in \{0, 1\}$ матрицы $\mathbf{\Gamma}$ равен нулю, то соответствующий элемент матрицы параметров $\mathbf{W}$ обнуляется, и не участвует в работе модели. Множество индексов соответствующих ненулевым элементам матрицы $\mathbf{\Gamma}$ обозначается $\mathcal{A}$. Требуется найти такое подмножество индексов $\mathcal{A}^*$, которое доставляет минимум функции:

$$\mathcal{A}^* = \arg\min_{\mathcal{A}\subseteq\mathcal{I}} L(f_{\mathcal{A}}|\mathbf{w}^*, \mathfrak{D}_{\mathcal{C}}),$$

на разбиении выборки $\mathfrak{D}$, определенным множеством индексов $\mathcal{C}$. Здесь $\mathcal{I} = \mathcal{C} \sqcup \mathcal{L}$ — все индексы всех матриц $\mathbf{\Gamma}$. То есть требуется снизить число признаков и повысить устойчивость модели.

При этом параметры $\mathbf{w}^*$ модели доставляют минимум ошибки:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} L(\mathbf{w}|\mathfrak{D}_{\mathcal{L}}, f_{\mathcal{A}}),$$

на разбиении выборки, определенной множеством $\mathcal{L}$. Алгоритм поиска оптимальной структуры сети предполагает минимизацию функции ошибки.

Функция ошибки и критерии качества модели. Ключевой идеей данной работы является построение новой функции ошибки с использованием метапараметров аддитивной регуляризации. Предлагается использовать композитную функцию ошибки. Она состоит из нескольких слагаемых. Первое слагаемое соответствует точности восстановления зависимой переменной. Второе слагаемое это точность реконструкции независимой переменной автокодировщиком. Остальные k слагаемых отвечают за аддитивную регуляризацию. Создается каталог слагаемых функции ошибки для построения аддитивной регуляризации.

Подробное рассмотроение типов используемых регуляризаторов представлено в Supplementary

Роль в аддитивной регуляризации	Тип регуляризатора
Ошибка выхода нейронной сети	$\|\|\mathbf{y} - f(\mathbf{W})\|\|_2^2$
Ошибка восстановления на каждом слое	$\|\|\mathbf{x} - \mathbf{r}(\mathbf{x})\|\|_2^2$
L_1 и L_2 регуляризация	$\|\|\mathbf{w} - \mathbf{w}_0\|\|_1$, $\|\|\mathbf{w} - \mathbf{w}_0\|\|_2^2$
Штраф за отличие матрицы одного слоя от тождественного преобразования	$\|\|\mathbf{W} - \mathbf{I}\|\|$
Штраф за отличие матрицы одного слоя от метода главных компонент	$\|\|\mathbf{W}\mathbf{W}^T - \mathbf{I}\|\|$
Тихоновская регуляризация	$\|\mathbf{TW}\|$

Таблица 1: Каталог регуляризаторов аддитивной функции ошибки

Задача (10) является задачей минимизации функции L, включающая слагаемое (8) и (9) для оптимизации параметров модели (6)

$$L = \lambda_x E_{\mathbf{x}} + \lambda_y E_{\mathbf{y}} + \lambda_1 \mathcal{R}_1 + \cdots + \lambda_k \mathcal{R}_k = \lambda_x E_x + \lambda_y E_y + \sum_{i=1}^{k} \lambda_i \mathcal{R}_i(\mathbf{W}), \tag{7}$$

где $\mathcal{R}_i = \mathcal{R}(\mathbf{W}) = (\mathfrak{r}_1(\mathbf{W}), \cdots, \mathfrak{r}_r(\mathbf{W}))^{\intercal}$ —вектор, состоящий из значений регуляризаторов i-ого слоя.Метапараметры аддитивной регуляризации представляют себя матрицу размером $k \times r$, где k —число слоев, а r —число регуляризаторов в каждом слое:

$$\begin{bmatrix} \lambda_{1,1} & \lambda_{1,2} & \dots & \lambda_{1,r} \\ \dots & \dots & \dots & \dots \\ \lambda_{k,1} & \lambda_{k,2} & \dots & \lambda_{k,r} \end{bmatrix}.$$

В свою очередь $\boldsymbol{\lambda}_i$ представляет собой вектор

$$\lambda_{\mathbf{k}} = [\lambda_1, \lambda_2, \dots, \lambda_r].$$

Каждый элемент этого вектора соответствует регуляризатору соответствующего слоя. Подобный подход позволяет варьировать структуру функции ошибки. Например,если занулить ошибки E_y и аддитивную регуляризацию, оставив только ошибку E_x в функции (7), то слой будет вести как автокодировщик. Или наоборот, при слабой регуляризации параметра E_y получится нейросетевой слой.

При оптимизации структуры Г модели глубокого обучения используется три вида критериев качества: точность, устойчивость и сложность.

Точность. Когда в качестве используемой модели выступает нейросеть или линейная регрессия, то функция ошибки имеет вид:

$$E_y = \sum_{((\mathbf{x},y)\in D)\in\mathcal{I}} \big(y - f(\mathbf{x})\big)^2. \tag{8}$$

Эта функция ошибки включает в себя полученные предсказания модели и значения зависимых переменных. В задачах регрессии точность аппроксимации имеет вид:

$$\text{MAE} = \frac{\sum\limits_{(\mathbf{x},y)\in D} |y - f(\mathbf{x})|}{|D|}.$$

При включении в модель (6) метода главных компонент или автокодировщика, метки объектов не используются. Функция ошибки штрафует невязки восстановленного объекта:

$$E_{\mathbf{x}} = \sum_{i\in\mathcal{I}} \|\mathbf{x}_i - \mathbf{r}(\mathbf{x}_i)\|_2^2\,, \tag{9}$$

где $\mathbf{r}(\mathbf{x})$ это линейная реконструкция объекта $\mathbf{x}$. Параметры автокодировщика

$$\mathbf{W}_{\text{AE}} = \{\mathbf{W}^{'}, \mathbf{W}, \mathbf{b}^{'}, \mathbf{b}\}$$

оптимизированы таким образом (9), чтобы приблизить реконструкцию $\mathbf{r}(\mathbf{x})$ к исходному вектору $\mathbf{x}$.

Сложность. Введем отношение порядка $\succ$ на множестве значений сложности. Это отношение задается множеством параметров модели:

1) один параметр: $w \in \mathbb{R}^1 \succ w \in \lambda_1[0,1] + \lambda_0 \succ w \in c + \lambda_0$,

2) вектор(нейрон): $\mathbf{w} \in \mathbb{R}^n \succ \|\mathbf{w}\|^2 = 1 \succ \mathbf{w} = \text{const}$,

3) матрица(слой): $\mathbf{W} \in \mathbb{R}^{c\times n} \succ \mathbf{W}^\mathsf{T}\mathbf{W} = \mathbf{I} \succ \mathbf{W} = \text{const}$.

Множество, которому принадлежит сложность модели – порядковое. Исходя из введенного понятия сложности модели упорядочены по возрастанию сложности:

1) линейная регрессия, $\sigma^{'} = \text{id}, \sigma = \text{id}, \mathbf{W} = \mathbf{I}_n$,

2) линейная регрессия и метод главных компонент, $\sigma^{'} = \text{id}, \mathbf{W}^\mathsf{T}\mathbf{W} = \mathbf{I}_n$,

3) линейная модель и автокодировщик, $\mathbf{W}^\mathsf{T}\mathbf{W} \neq \mathbf{I}_n$,

4) линейная модель и стэк автокодировщиков, представимый в виде суперпозиции (6),

5) двухслойная нейронная сеть,

6) глубокая нейронная сеть.

Устойчивость — это минимум дисперсии функции ошибки (5):

$$\mathsf{D}(S) \to \min\,.$$

Формулировка задачи Таким образом, задача сводится к следующему виду:

$$L(\mathbf{w}, \lambda) = E_{\mathbf{y}} + E_{\mathbf{x}} + \sum_{i=1}^{r} \lambda_i^T \mathcal{R}(\mathbf{W}_i)$$

$$\begin{aligned} \mathbf{w} &= \arg\min L(f|\lambda, \mathbf{\Gamma}) \\ \lambda &= \arg\min L(f|\mathbf{w}, \mathbf{\Gamma}) \\ \mathbf{\Gamma} &= \arg\min L(f|\mathbf{w}, \lambda) \end{aligned} \tag{10}$$

Расписание оптимизации

Требуется создать расписание оптимизации метапараметров регуляризации $\boldsymbol{\lambda}$. Требуется назначать метапараметр в зависимости от номера итерации. Для выбора $\boldsymbol{\lambda}$ предлагается использовать три подхода: 1) экспертное задание, 2) эвристики, 3) алгоритмы оптимизации

У метапараметров регуляризации два назначения. Первое —это параметр регуляризации перед ошибкой каждого слоя нейросети (11). С его помощью задается величина вклада соответствующего регуляризатора в функцию ошибки. Второе назначение — изменение функции слоя в структуре всей нейросети. Например, если метапараметр λ_x будет положительным, а все остальные нулевыми, то слой будет вести себя как автокодировщик.

Экспертное задание расписания оптимизации

1. Шаг 1: Пусть s первых слоев обучаются как автокодировщик, а последние $k - s$ слоев обучаются как нейронная сеть, то есть для s слоев $\lambda_x = 1$, для последних s слоев $\lambda_y = 1$, а прочие метапараметры равны 0. На второй итерации веса автокодировщика замораживаются и больше не оптимизируются.

Шаг 2 :В L добавляются все регуляризаторы $s + 1$-ого слоя, кроме λ_x с параметрами 1. Сеть дообучается. Веса $s + 1$ слоя замораживаются, процедура идет дальше: к $s + 2$-ому слою добавляются все регуляризаторы кроме λ_x с параметрами 1, сеть дообучается и так далее.

2. Шаг 1 аналогичен предыдущему.

Шаг 2: Начиная с $s + 1$ добавляется регуляризатор λ_1 ко всем слоям, сеть дообучается, веса на $s + 1$ слое замораживаются. И так далее, с каждой новой итерацией добавляется новый регуляризатор ко всем слоям, начиная с последнего замороженного.

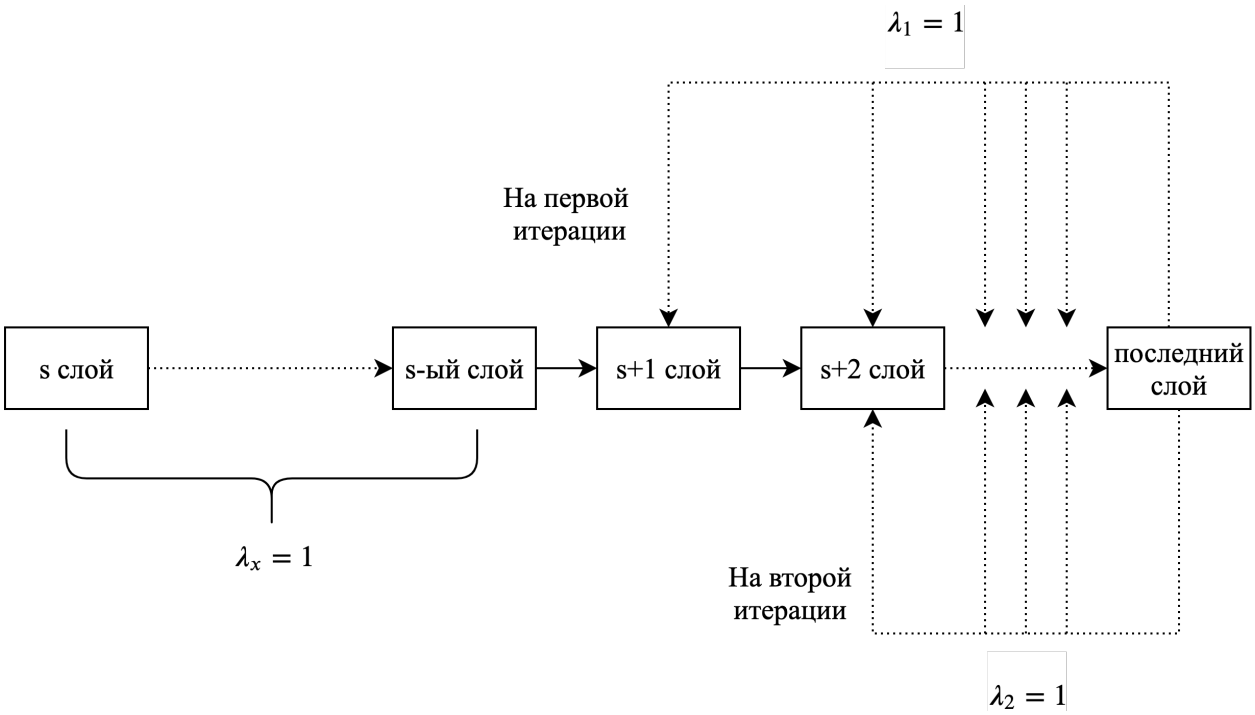

Рис. 1: Экспертное задание расписания оптимизации, вар 1

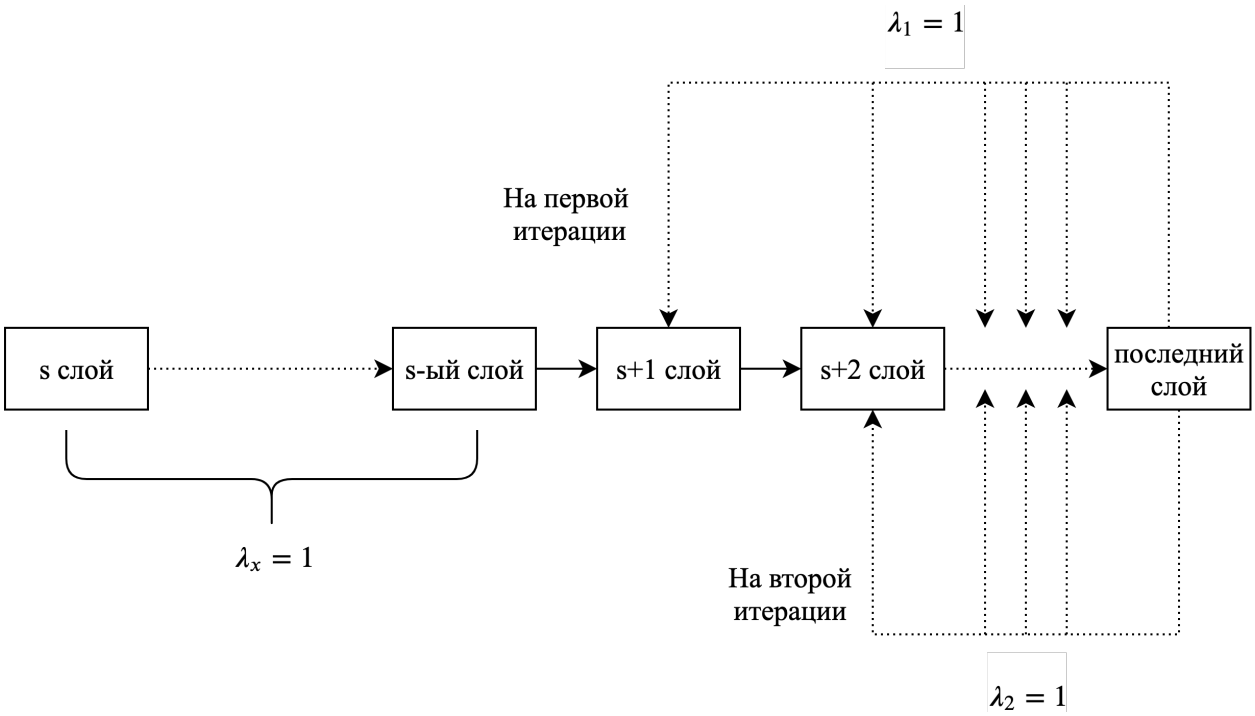

Рис. 2: Экспертное задание расписания оптимизации, вар 1

Эвристика Для оптимального выбора метапараметра λ для регуляризации $L2$ в работе [7] в разделе 6 предложен следующий подход:

Рассмотрим функционал

$$Kf(x) = g,$$

где g — искомая функция зависимости, то есть $y_i = g(x_i) + \varepsilon_i, \quad \varepsilon_i$ — некоторый шум. В гильбертовом пространстве можно найти такой η_x, что

$$Kf(x) = (f, \eta_x) = \mathbf{w}^x = (w, x) \Rightarrow \eta_x = id(x).$$

При наличии линейности и ограниченности оператора K существует такая матрица A, что

$$A\boldsymbol{y} = Kf_\lambda = Q\left(Q + n\lambda I\right)^{-1}\boldsymbol{y},$$

где

$$Q_{ij} = q(x_i, x_j) = \left(\eta_{x_i}, \eta_{x_j}\right) = (x_i, x_j),$$

а f_λ — решение задачи

$$\min \frac{1}{n}\sum_{i=1}^{n}(Kf(x_i) - y_i)^2 + \lambda\|f\|^2$$

. Построим функцию

$$V(\lambda) = \frac{n^{-1}\|(I - A)y\|^2}{\left[n^{-1}\operatorname{tr}(I - A)\right]^2},$$

минимизация которой даст искомую λ.

Вычислительный эксперимент

Исследуется процедура (построения расписания оптимизации)оптимизации структуры нейросети. Требуется снизить сложность с сохранением качества аппроксимации. Структура сети оптимизируется с помощью метода аддитивной регуляризации. Цель вычислительного эксперимента состоит в определении оптимальных значений метапараметров регуляризации, а так же исследовании зависимости точности,сложности и устойчивости модели от процедуры регуляризации, задаваемой метапараметрами. Для сравнения сложности структуры и исследования зависимости ошибки от сложности вводится отношение порядка, как было описано в параграфе **Сложность** в части **Постановка задачи**. .

Процедура построения модели включает в себя следующие шаги:

1. Начальная структура сети задается экспертно.
2. Оптимизация параметров методом стохастического градиентного спуска.
3. Оптимизация метапараметров регуляризации.
4. Оптимизация структуры сети генетическим алгоритмом.

Для получения результатов исследования строятся графики зависимости ошибки от сложности модели, в том числе от числа включенных регуляризаторов. Метарпараметры λ определяются для каждого слоя отдельно. В сети используются активационные функции $ReLU$. Матрица метапараметров выглядит следующим образом

$$\{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_k\}^{\mathsf{T}},$$

где $\boldsymbol{\lambda}_i$ — вектор метапараметров для i-ого слоя.

Наборы данных. Качество предлагаемого подхода к построению функции ошибки оценивается на нескольких реальных наборах данных и одном синтетическом наборе. Выборки взяты из открытого репозитория данных для машинного обучения [?]. Описание всех выборок представлено в табл 1. Синтетический набор данных состоит из признаков с различными свойствами ортогональности и коррелированности друг с другом и к целевой переменной. Процедура генерации синтетических данных описана в работе [8]. Возможны следующие конфигурации синтетических данных.

1. Неполный и скоррелированный: набор данных, содержащий коррелирующие признаки, ортогональные с целевому вектору.

2. Адекватный и случайный: набор данных, содержащий случайные признаки, и имеющий один признак, апроксимирующий целевой вектор.

3. Адекватный и избыточный: набор данных, содержащий признаки, коррелирующие с целевым вектором.

4. Адекватный и скорреллированный: набор данных, содержащий ортогональные признаки, и признаки, коррелирующие с ортогональными. Целевой вектор является суммой ортогональных векторов.

Каждый набор данных разбивается на три части.

1. Обучающая выборка — 60% от исходного набора. На этой выборке модель тренируется, и фиксируются значения параметров.

2. Валидационная выборка — 20% от исходного набора. На этой выборке применяется генетический алгоритм, который ищет оптимальную структуру.

3. Тестовая выборка — 20% от исходного набора. Она никак не участвует в оптимизации структуры модели. Эта выборка используется только для контроля качества — сравнение модели исходной и оптимизированной структуры, а так же сравнение с другими алгоритмами прореживания сетей.

Таблица 2: Описание выборок для экспериментов

Выборка $\mathfrak{D}$	Размер train	Размер val	Размер test	Объекты	Признаки
Credit Card	18000	6000	6000	30000	35
Protein	27438	9146	9146	45730	9
Airbnb	6298	2100	2100	10498	16
Wine quality	2938	980	980	4898	11
Synthetic	1200	400	400	2000	30

Реализация Первый шаг : тестовая выборка разбивается на подвыборки заданного (300) размера. На каждом шаге работы сети веса оптимизируются на каждой такой подвыборке. Итого общее число итераций будет равно произведению количества таких подвыборок на экспретно заданное число шагов оптимизации.

Второй шаг : Обучение на сети с заданными параметрами. Так как автокодировщик и нейронная сеть имеют разный смысл: восстановление независимой и зависимой переменной, то для контроля соответствующей ошибки избран следующий метод : к последнему слою добавляются два дополнительных слоя : один на выход автокодировщика, другой на выход нейронной сети. Благодаря этому можно оптимизировать веса сети с учетом выбранных регуляризаторов.

Третий шаг : варьирование гиперпараметров. Изменяется число слоев, число нейронов в слоях, регуляризаторы.

Рассмотрим влияние регуляризации на дисперсию параметров

$$\sigma^2 = \sum_{j=1}^{k} \frac{1}{k} W_k^\intercal W_k$$

:

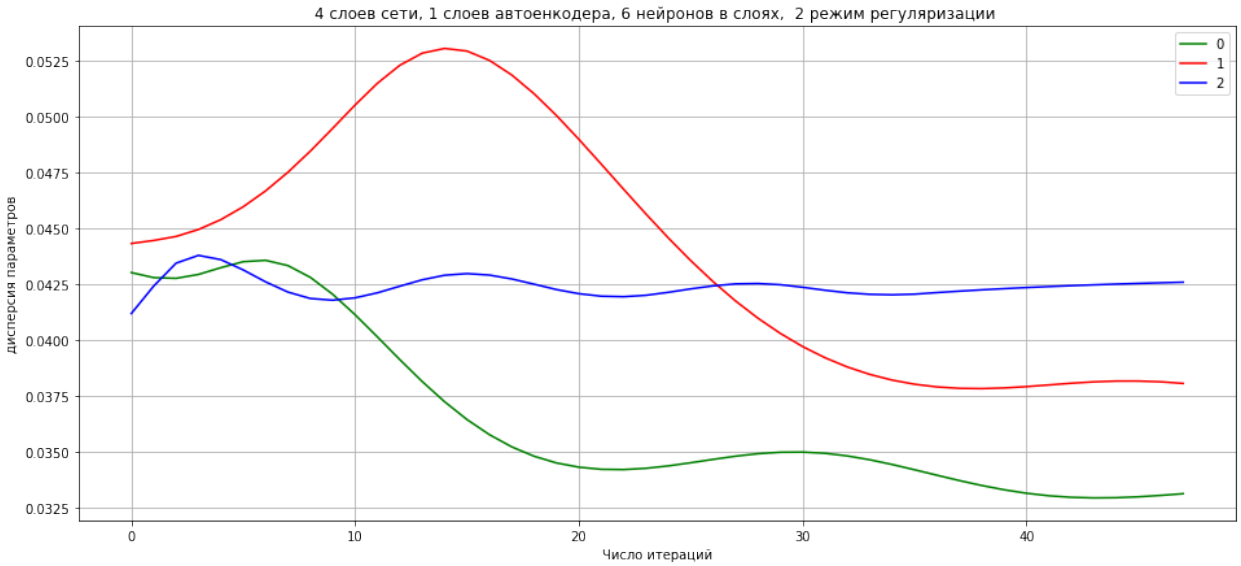

Рис. 3: При низкой сложности регуляризация может привести к повышению дисперсии

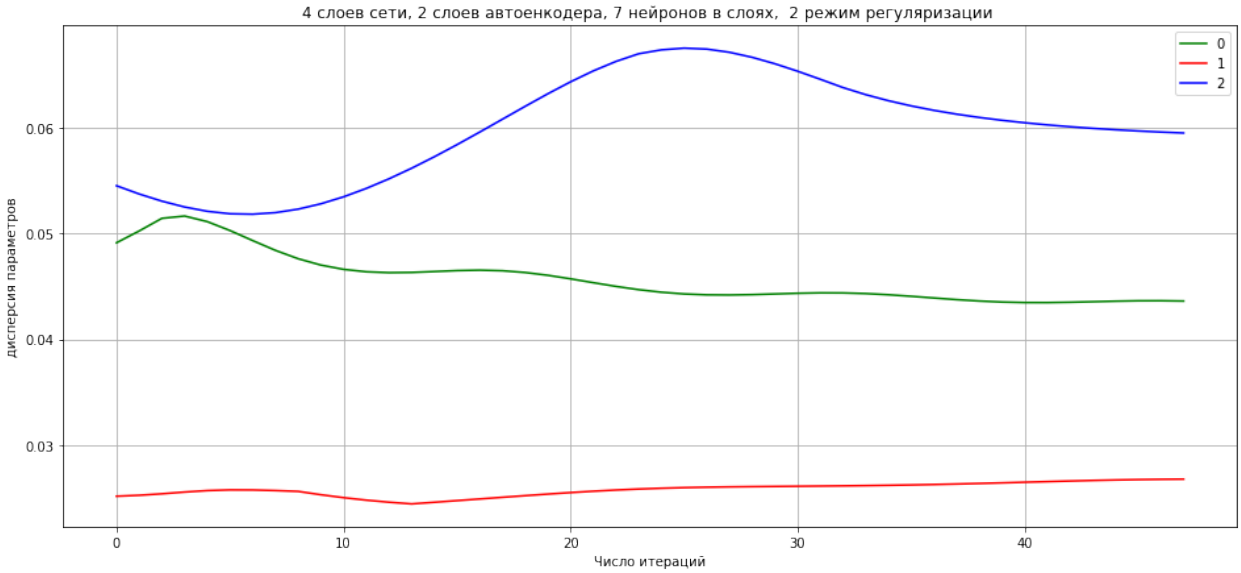

Рис. 4: Одно из расписаний может оказаться неэффективным

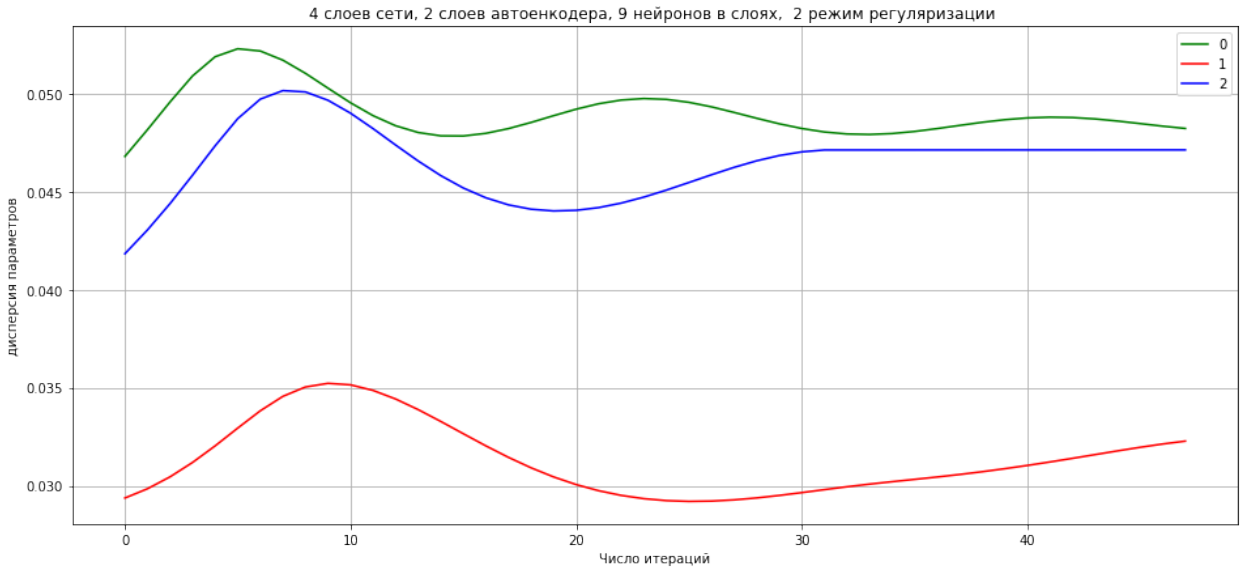

Рис. 5: Обычное поведение – регуляризованные модели показывают меньшую дисперсию

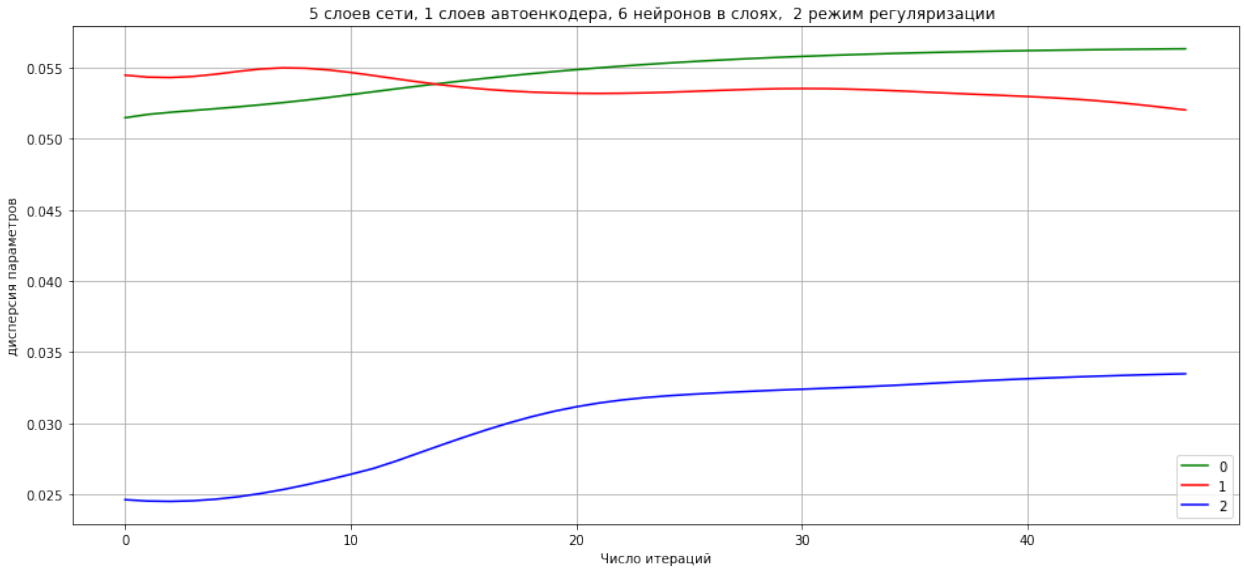

Рис. 6: Одно из расписаний оказывается эффективнее другого

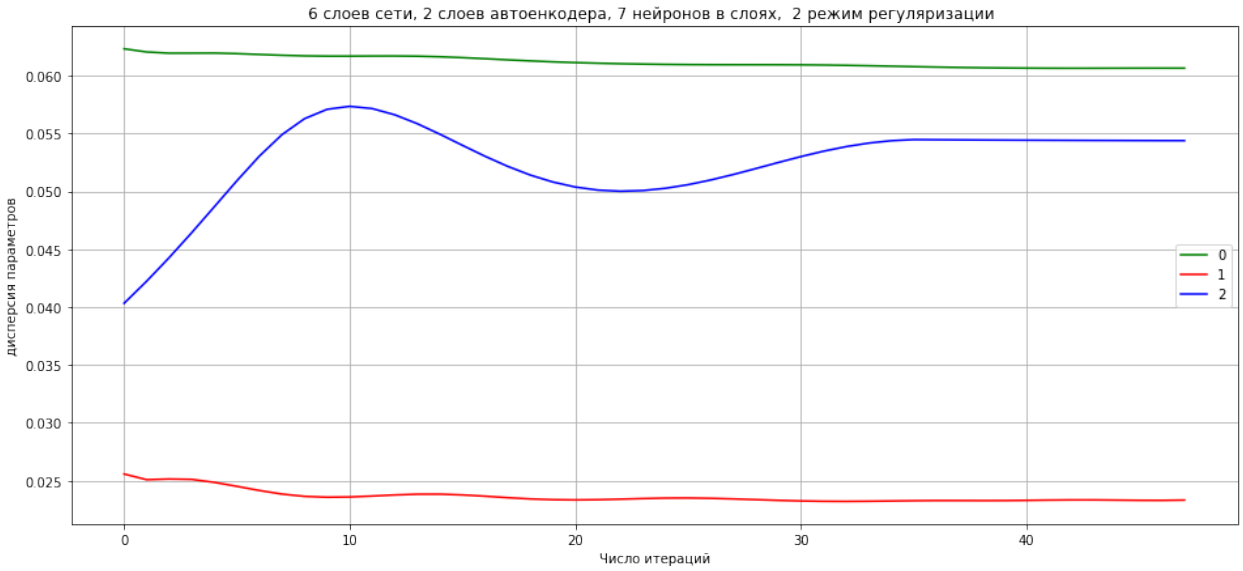

Рис. 7: Одно из расписаний оказывается эффективнее другого

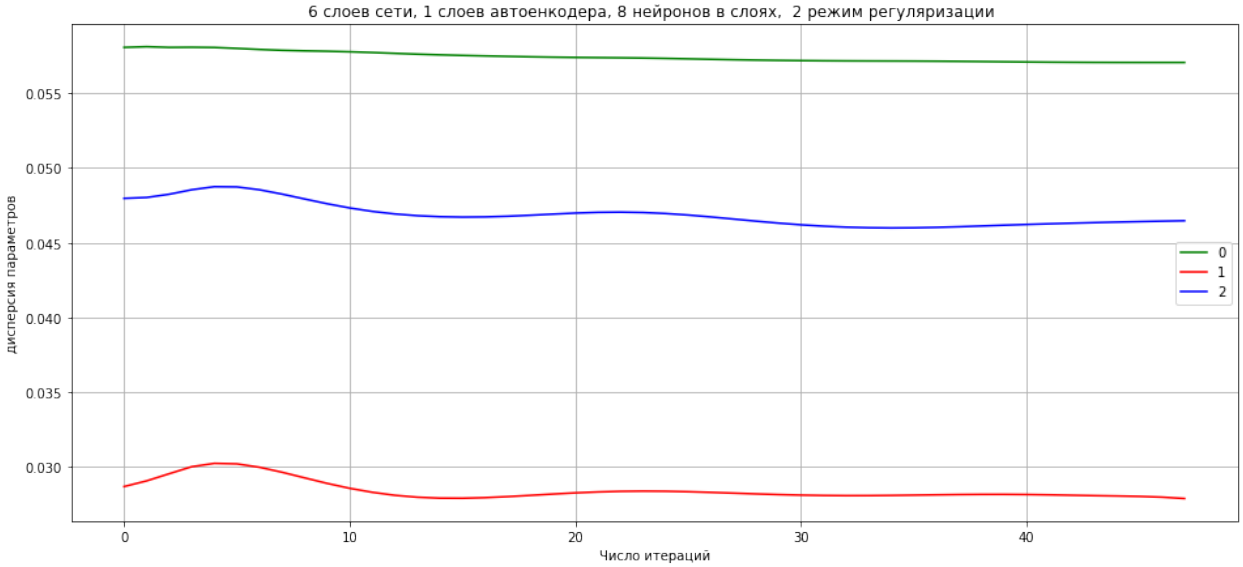

Рис. 8: Дисперсия параметров может оказываться постоянной

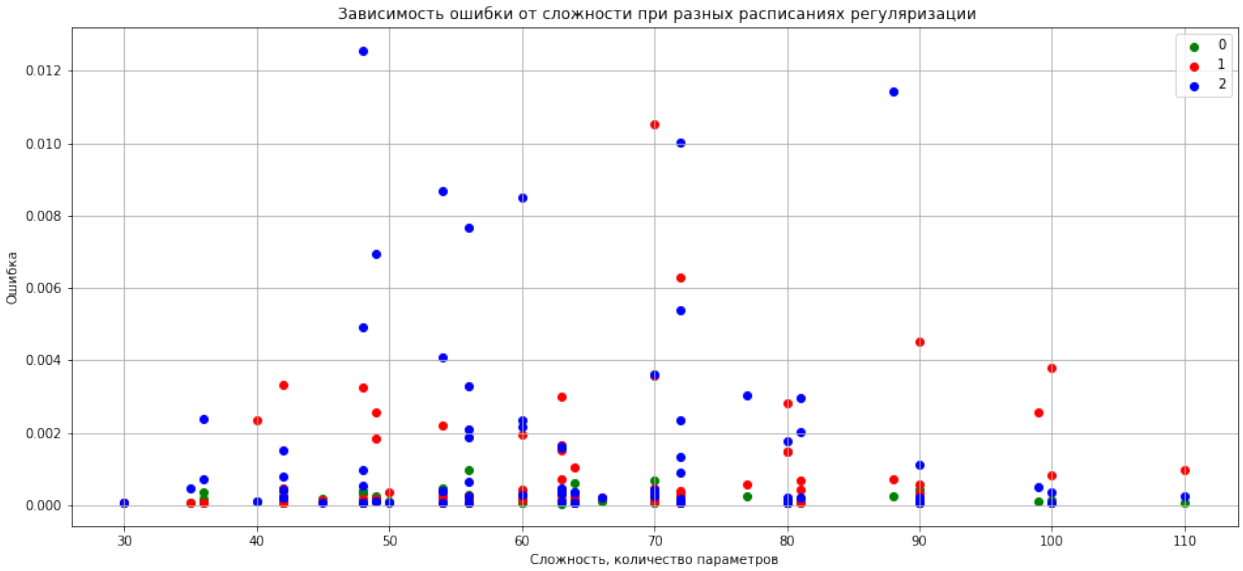

Рис. 9: Зависимость ошибки от сложности

Также была исследована зависимость между сложностью и устойчивостью модели:

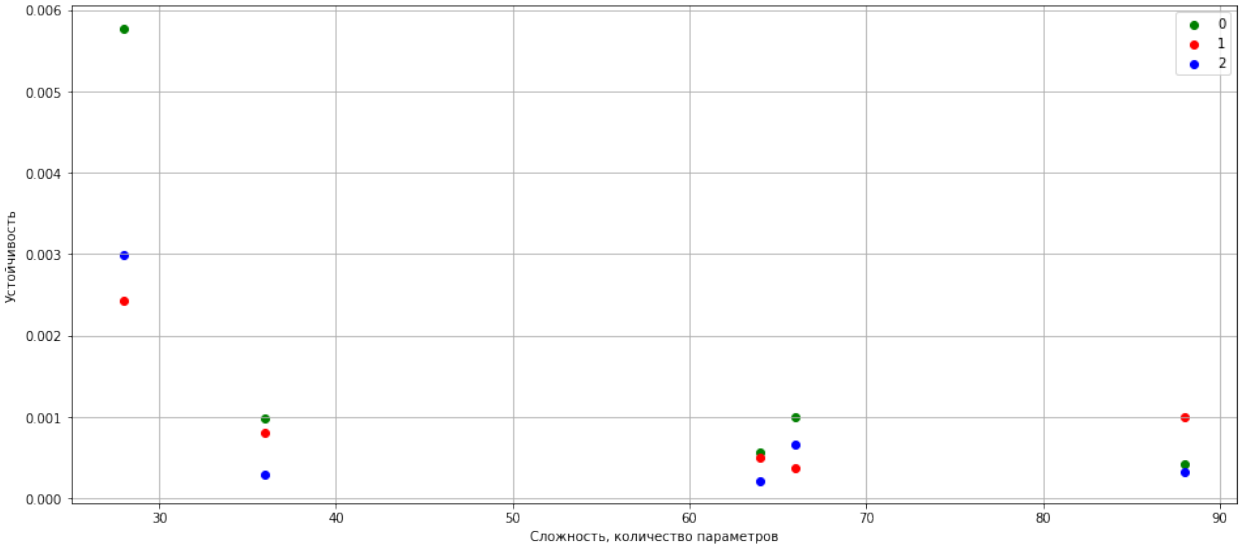

Рис. 10: Зависимость между сложностью и устойчивостью

Результат : Видим, что с добавленной регуляризацией при равной сложности точность выше. Устойчивость, то есть дисперсия ошибки, меньше при добавленной регуляризации, как и ожидалось.

Заключение

Supplementary

Линейная или логистическая регрессия и один нейрон — имеют вид

$$f(\mathbf{x}, \mathbf{w}) = \boldsymbol{\sigma}(\mathbf{w}^\mathsf{T}\mathbf{x}), \tag{11}$$

где $\boldsymbol{\sigma}$ — функция активации, непрерывная монотонная дифференцируемая функция (12), $\mathbf{w}$ — вектор параметров, $\mathbf{x}$ — объект, вектор с присоединенным элементом единица соответствующим аддитивному параметру w_0. При использовании линейной функции активации, получаем линейную регрессию $f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^\mathsf{T}\mathbf{x}$.

Такую функцию активации мы обозначим $\boldsymbol{\sigma} = \mathbf{id}$. При использовании сигмоидной функции активации, получаем модель логистической регрессии

$$f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{w}^\mathsf{T}\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^\mathsf{T}\mathbf{x})}. \tag{12}$$

Двухслойная нейронная сеть, состоящая из линейной комбинации нейронов, однослойных нейронных сетей

$$f(\mathbf{x}, \mathbf{w}) = \sigma^{(2)}\left(\sum_{i=1}^{n_2} w_i^{(2)} \cdot \sigma^{(1)}\left(\sum_{i=1}^{n} w_{ij}^{(1)} x_j + w_{i0}^{(1)}\right) + w_0^{(2)}\right) = \sigma \circ \mathbf{w}^\mathsf{T}\boldsymbol{\sigma} \circ \mathbf{W}\mathbf{x}.$$

Метод главных компонент. Модель допускает вращения признакового пространства, то есть координаты (признаки) преобразовываются только с помощью поворотов:

$$\mathbf{h} = \mathbf{W}\mathbf{x},$$

где $\mathbf{W}$ — матрица поворота. Она ортогональна:

$$\mathbf{W}\mathbf{W}^\mathsf{T} = \mathbf{I}_n. \tag{13}$$

Полученное пространство образов $\mathbf{h}$ называется скрытым. Происходит преобразование без потерь.

При удалении нескольких строк матрицы $\mathbf{W}$, например их число $u < n$, полученный вектор $\mathbf{h}$ имеет размер $u \times 1$. Получается проекция $\mathbf{h}$ вектора $\mathbf{x}$. Согласно теореме Рао С.Р. [**?**], первые u главных компонент восстанавливают $\mathbf{h}$ оптимальным способом,

$$\mathbf{r}(\mathbf{x}) = \mathbf{W}^\mathsf{T}\mathbf{h}.$$

Автокодировщик $\mathbf{h}$ — это монотонное нелинейное отображение входного вектора свободных переменных $\mathbf{x} \in \mathbb{R}^n$ в скрытое представление $\mathbf{h} \in \mathbb{R}^u$ вида:

$$\mathbf{h}(\mathbf{x}) = \boldsymbol{\sigma}(\underset{u\times n}{\mathbf{W}}\mathbf{x} + \mathbf{b}).$$

В случае $\boldsymbol{\sigma} = \mathbf{id}$ и (13) автокодировщик тождественен методу главных компонент. Скрытое представление $\mathbf{h}$ реконструирует вектор $\mathbf{x}$ линейно:

$$\mathbf{r}(\mathbf{x}) = \underset{n\times u}{\mathbf{W}'}\mathbf{h} + \mathbf{w}_0'.$$

1. Lasso или L_1 регуляризация вида:

$$\mathfrak{r}_1(w) = \|w\|_1$$

2. Штраф за количество слоев в нейронной сети:

$$\mathfrak{r}_2(k) = k$$

3. Штраф за неортогональность матрицы:

$$\mathfrak{r}_3(W) = \|WW^T - I\|$$

4. Несколько видов Тихоновской регуляризации:

   4.1. Ridge или L_2 регуляризация вида:

$$\mathfrak{r}_4(w) = \|w\|_2^2$$

   4.2. Штраф за частоту появления весов

$$A = \frac{1}{3}\begin{bmatrix} \frac{2}{3} & \frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & \frac{2}{3} & \frac{2}{3} \end{bmatrix}$$

$$\mathfrak{r}_5(W) = \|(I - A)W\|$$

   4.3. Штраф за локальную разницу в весах

$$B = \begin{bmatrix} -2 & 2 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 2 \end{bmatrix}$$

$$\mathfrak{r}_6(W) = \|BW\|$$

Помимо аддитивной регуляризации были разработаны и другие методы. В статье [9] предлагается алгоритм вычисления ошибки при использовании контроля по отдельным объектам.

Задана центрированная нормированная выборка :

$$\{(\mathbf{x}_i, y_i), i = 1, \ldots, N\}, \mathbf{x}_i = (x_{1i} \cdots x_{ni})^T \in \mathbb{R}^n, y_i \in \mathbb{R}$$
$$\sum_{i=1}^{N} \mathbf{x}_j = \mathbf{0}, \sum_{i=1}^{N} y_j = 0, \frac{1}{N}\sum_{i=1}^{N} x_{ij}^2 = 1, i \in I = \{1, \ldots, n\},$$

Задача – минимизировать вектор $\mathbf{a}$, параметры регрессии $\hat{y}(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$:

$$\begin{aligned} J_{\text{NEN}}(\mathbf{w}|\lambda_1, \lambda_2) &= \lambda_2 \sum_{i=1}^{n} a_i^2 + \lambda_1 \sum_{i=1}^{n} |a_i| + \sum_{i=1}^{N} \left(y_j - \sum_{i=1}^{n} a_i x_{ij} \right)^2 \\ &= \lambda_2 \mathbf{w}^T\mathbf{w} + \lambda_1 \|\mathbf{w}\|_1 + (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \to \min(\mathbf{w}) \\ \mathbf{y} &= (y_1 \cdots y_N) \in \mathbb{R}^N, \mathbf{X} = (\mathbf{x}_1 \cdots \mathbf{x}_N)^T (N \times n) \\ \hat{\mathbf{w}}_{\lambda_1,\lambda_2} &= (\hat{a}_{i,\lambda_1,\lambda_2}, i \in I) = \arg\min J_{\text{NEN}}(\mathbf{w}|\lambda_1, \lambda_2) \in \mathbb{R}^n \end{aligned}$$

Предлагается следующий подход:

$$\begin{aligned} J_{\text{EN}}(\mathbf{w}|\lambda_1, \lambda_2) &= \lambda_2 \textstyle\sum_{i=1}^{n} (a_i - a_i^*)^2 + \lambda_1 \sum_{i=1}^{n} |a_i| + \sum_{i=1}^{N} (y_j - \sum_{i=1}^{n} a_i x_{ij})^2 \\ &= \lambda_2 \left(\mathbf{w} - \tfrac{1}{N}\mathbf{X}^T\mathbf{y} \right)^T \left(\mathbf{w} - \tfrac{1}{N}\mathbf{X}^T\mathbf{y} \right) + \lambda_1 \|\mathbf{w}\|_1 + (\mathbf{y} - \mathbf{X}\mathbf{w})^T(\mathbf{y} - \mathbf{X}\mathbf{w}) \to \min(\mathbf{w}) \\ \hat{\mathbf{w}}_{\lambda_1,\lambda_2} &= (\hat{a}_{i,\lambda_1,\lambda_2}, i \in I) = \arg\min J_{\text{EN}}(\mathbf{w}|\lambda_1, \lambda_2) \in \mathbb{R}^n, \end{aligned}$$

где $\mathbf{w}^* = (1/N)\mathbf{X}^T\mathbf{y}$. Таким образом, можно переписать задачу в следующем виде:

$$\frac{\lambda_1}{1 + \lambda_2/N}\|\mathbf{w}\|_1 + \left[\mathbf{w}^T \frac{\mathbf{X}^T\mathbf{X} + \lambda_2\mathbf{I}}{1 + \lambda_2/N}\mathbf{w} - 2\mathbf{y}^T\mathbf{X}\mathbf{w} \right] \to \min(\mathbf{w}),$$

Вводится разбиение множества параметров по знаку:

$$\begin{cases} \hat{I}_{11}^-, \lambda_2 = \{i \in I : \hat{a}_{i,\lambda_1,\lambda_2} < 0\} \\ \hat{I}_{1_1,\lambda_2}^0 = \{i \in I : \hat{a}_{i,\lambda_1,\lambda_2} = 0\}, I = \hat{I}_{\lambda_1,\lambda_2}^- \cup \hat{I}_{\lambda_1,\lambda_2}^0 \cup \hat{I}_{\lambda_1\lambda_2}^+ \\ \hat{I}_{\lambda_1^+,\lambda_2}^+ = \{i \in I : \hat{a}_{i,\lambda_1,\lambda_2} > 0\} \end{cases}$$

. После некоторых преобразований:

$$\begin{aligned} \hat{S}_{\text{LOO}}(\lambda_1, \lambda_2) &= \tfrac{1}{N} \textstyle\sum_{k=1}^{N} \left(\hat{\delta}_{k,\lambda_1,\lambda_2}^{(k)} \right)^2 \\ \hat{\delta}_{k,\lambda_1,\lambda_2}^{(k)} &= y_k - \hat{y}_{k,\lambda_1,\lambda_2}^{(k)} = y_k - \tilde{\mathbf{x}}_k^T \hat{\mathbf{w}}_{\lambda_1,\lambda_2}^{(k)} \end{aligned}$$

и

$$\hat{S}_{\text{LOO}}^{\text{EN}}(\lambda_1, \lambda_2) = \frac{1}{N}\sum_{k=1}^{N} \left(\frac{\hat{\delta}_{k,\lambda_1,\lambda_2} + \frac{1}{N-1}\lambda_2 (y_k q_{k,\lambda_1,\lambda_2} - h_{k,\lambda_1,\lambda_2})}{1 - q_{k,\lambda_1,\lambda_2}} \right)^2 \text{(ElasticNet)}$$

$$\begin{aligned} q_{k,\lambda_1,\lambda_2} &= \tilde{\mathbf{x}}_k^T \left(\tilde{\mathbf{X}}_{\lambda_1,\lambda_2}^T \tilde{\mathbf{X}}_{\lambda_1,\lambda_2} + \lambda_2 \tilde{\mathbf{I}}_{\hat{n}_{\lambda_1,\lambda_2}} \right)^{-1} \tilde{\mathbf{x}}_k \\ h_{k,\lambda_1,\lambda_2} &= \tilde{\mathbf{x}}_k^T \left(\tilde{\mathbf{X}}_{\lambda_1,\lambda_2}^T \tilde{\mathbf{X}}_{\lambda_1,\lambda_2} + \lambda_2 \tilde{\mathbf{I}}_{\hat{n}_{\lambda_1,\lambda_2}} \right)^{-1} \tilde{\mathbf{w}}^* \end{aligned}$$

Список литературы

[1] Andrey N Tikhonov and Vasiliy Y Arsenin. Solutions of ill-posed problems. *New York*, pages 1–30, 1977.

[2] Markus Svensén and Christopher M Bishop. Pattern recognition and machine learning. 2007.

[3] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[4] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

[5] Eldad Haber and Lars Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34(1):014004, 2017.

[6] Jiaxuan Wang, Jeeheh Oh, Haozhu Wang, and Jenna Wiens. Learning credible models. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2417–2426, 2018.

[7] Mark A Lukas et al. Methods for choosing the regularization parameter. In *Mini Conference on Inverse Problems in Partial Differential Equations*, pages 89–110. Centre for Mathematics and its Applications, Mathematical Sciences Institute . . . , 1992.

[8] AM Katrutsa and VV Strijov. Stress test procedure for feature selection algorithms. *Chemometrics and Intelligent Laboratory Systems*, 142:172–183, 2015.

[9] Elena Chernousova, Nikolay Razin, Olga Krasotkina, Vadim Mottl, and David Windridge. Linear regression via elastic net: Non-enumerative leave-one-out verification of feature selection. In *Clusters, Orders, and Trees: Methods and Applications*, pages 377–390. Springer, 2014.