

Содержание

1 Введение	5
2 Априорные распределения для задачи смеси экспертов	7
2.1 Постановка задачи аппроксимации параметров окружности	8
2.2 Вероятностная постановка задачи смеси экспертов	10
2.3 Вычислительный эксперимент по анализу качества аппроксимации радужки глаза смесью экспертов	13
3 Модели привилегированного обучения и дистиляции	19
3.1 Постановка задачи обучения с учителем: Хинтона и Вапника	22
3.2 Обобщенная вероятностная постановка задачи дистиляции	24
3.3 Анализ вероятностного подхода к дистиляции моделей линейных моделей	30
4 Байесовская дистиляция моделей глубокого обучения	34
4.1 Постановка задачи дистиляции в терминах байесовского подхода	35
4.2 Построение априорного распределения параметров ученика на основе параметров учителя	37
4.3 Анализ качества байесовской дистиляции полносвязных нейронных сетей	40
5 Введение отношения порядка на множестве параметров аппроксимирующих моделей	44
5.1 Задача упорядочивания параметров аппроксимирующих моделей	45
5.2 Фиксация параметров модели в процессе обучения	46
5.3 Вычислительный эксперимент по упорядочиванию параметров . .	46
6 Релевантность параметров параметрических моделей	52
6.1 Постановка задачи к назначению релевантности параметрам модели	52
6.2 Анализ разных подходов к определению релевантности	56
7 Оптимальный размер выборки для построения линейных моделей	62
7.1 Постановка задачи определения оптимального размера выборки .	65
7.2 Обзор методов для определения оптимального размера выборки на основе статистических тестов	65
7.3 Эвристические методы определения достаточного размера выборки	68
7.4 Байесовский подход к определению оптимального размера выборки	69
7.5 Вычислительный эксперимент по анализу разных подходов к определению оптимального размера выборки	70

8 Аппроксимация кривых второго порядка при помощи обучения с экспертом	74
8.1 Постановка задачи поиска параметров кривых второго порядка	77
8.2 Композиция кривых второго порядка на изображении	79
8.3 Анализ смеси экспертов для аппроксимации кривых второго порядка на изображении	80
9 Локальные модели в задачах кластеризации временных рядов	86
9.1 Постановка задачи кластеризации точек временного ряда	88
9.2 Кластеризация точек в фазовом пространстве	89
9.3 Анализ фазовых траекторий в задаче кластеризации точек временного ряда	90
10 Заключение	96
Список литературы	97

Аннотация

Исследуется проблема понижения сложности аппроксимирующих моделей с целью повышения их интерпретируемости. В рамках данного исследования рассматриваются методы, которые используют экспертную информацию о данных с целью получения простых, более интерпретируемых моделей. Предлагаются методы, основанные на дистилляции моделей глубокого обучения, где модель учителя рассматривается в качестве эксперта. Также предлагаются методы основанные на экспертном описании задачи, что позволяет строить специальные признаковые описания объектов. Теоретические результаты анализируются в вычислительном эксперименте на синтетических выборках и реальных данных. В качестве реальных данных рассматривается популярные выборки, такие как MNIST, FashionMNIST и Twitter Sentiment Analysis.

Ключевые слова: выбор модели; байесовский вывод; дистилляция модели; локальные преобразования; преобразования вероятностных пространств; релевантность параметров.

1 Введение

В силу высокой вычислительной сложности, время оптимизации нейронных сетей может занимать до нескольких дней [48]. Построение и выбор оптимальной структуры нейронной сети также является вычислительно сложной процедурой, которая значимо влияет на итоговое качество модели. При этом алгоритмы оптимизации сходится по большинству параметров сети уже после небольшого числа итераций [29]. Своевременное определение начала сходимости параметров позволит существенно снизить вычислительные затраты на обучение моделей с большим числом параметров. Примерами моделей, с большим числом параметров, являются AlexNet [141], VGGNet [142], ResNet [148], BERT [144, 143], mT5 [146], GPT3[145] и другие.

Рост числа параметров моделей глубокого обучения влечет снижение интерпретируемости ответов этих моделей. Первые упоминания о данной проблеме рассмотрены А. Г. Ивахненко [154]. Проблема с неинтерпретируемыми моделями широко сейчас рассматривается в классе задач по *adversarial attack* [140].

Другой проблемой моделей с большим числом параметров является высокие требования к вычислителю в момент предсказания. Использование избыточно сложных моделей с избыточным числом неинформативных параметров является препятствием для использования глубоких сетей на мобильных устройствах в режиме реального времени. Для снижения числа параметров в литературе рассматривается метод дистилляции модели на основе предсказаний учителя [150, 162, 163]. Модель с большим числом параметров называется учителем. Модель учителя дистиллируется в модель с малым числом параметров, которая называется ученик. Основные идеи, которые описывают дистилляцию моделей глубокого обучения предложены в работах Дж. Е. Хинтона и В. Н. Вапником [150, 162, 163]. Работы предлагают использовать предсказания модели учителя для повышения качестве модели ученика. В работе [162] В. Н. Вапником вводится понятие привилегированной информации, которое позволяет использовать дополнительную информацию о данных в момент обучения модели. Работа [163] объединяет идеи дистилляции [150] с идеями привилегированного обучения [162], предложив метод дистилляции модели учителя с большим числом признаков в модель ученика с меньшим числом признаков. В предложенном методе [163] решается двухэтапная задача. На первом этапе строится модель учителя с расширенным признаковым описанием. На втором этапе обучается модель ученика в исходном признаковом описании используя дистилляцию [150]. В работе Дж. Е. Хинтона [150] поставлено множество экспериментов по дистилляции моделей глубокого обучения для задачи классификации. Один из экспериментов проводился на выборке MNIST [151], который показал, что предложенный дистилляции позволяет построить нейросетевую модель меньшей сложности на основе модели большей сложности. Второй эксперимент показывал идею по дистилляции ансамбля моделей в одну нейросетевую модель для решения задачи распознания речи. В работе [150] проводится сравнение

дистилляции с моделью смеси экспертов. Дальнейшие работы по дистилляции моделей глубокого обучения рассматривают возможность использования информации о значения параметров модели учителя для оптимизации параметров модели ученика. Работа [139] предлагает метод neuron selectivity transfer, который минимизирует специальную функцию потерь. Данная функция основается на maximum mean discrepancy между выходами слоев модели учителя и модели ученика. В рамках вычислительного эксперимента сравнивалось качество базовой дистилляции с предложенным методом на примере выборок CIFAR [137] и ImageNet [138].

Дистилляция моделей глубокого обучения работает в предположение, что архитектура модели ученика уже известная. Для выборка архитектуры модели ученика предлагается использовать методы прореживания нейросетевых моделей. Существует ряд подходов к построению оптимальной сети. В работах [41, 42] предлагается использовать модель градиентного спуска для оптимизации сети. В [46] используются байесовские методы [47] оптимизации параметров нейронных сетей. Другим методом поиска оптимальной структуры является прореживание избыточно сложной модели [45, 50, 160]. В работе [45] предлагается удалять наименее релевантные параметры на основе значений первой и второй производных функции ошибки. В [161] предложен метод определения релевантности параметров аппроксимирующих моделей при помощи метода Белсли. Релевантность параметров в работе [161] определяется на основе ковариационной матрицы параметров модели. Другим примером задания порядка на множестве параметров служит l_1 -регуляризация [30] и регуляризация ElasticNet [31] для линейных моделей. Порядок, заданный на множестве значений коэффициентов регуляризации, индуцирует порядок на множестве признаковых описаний и указывает на важность признаков. В случае нейросетей для регуляризации параметров используется метод исключения параметров [32, 46]. Данный метод также задает порядок на множестве параметров модели.

Порядок на множестве параметров нейросети можно использовать не только для удаления неимение релевантных параметров, а и для фиксации параметров в процессе оптимизации параметров. Работе [36] посвящена оптимизация структуры нейронной сети, а также выбору параметров, которые можно зафиксировать после некоторой итерации градиентного метода.

2 Априорные распределения для задачи смеси экспертов

В статье исследуется проблема построения смеси экспертов. Смесь экспертов – это мульти модель, которая состоит из множества локальных моделей, которые называются экспертами и шлюзовой функции. Смесь экспертов использует шлюзовую функцию для взвешивания прогнозов каждого эксперта. Весовые коэффициенты шлюзовой функции зависят от объекта, для которого производится прогноз. Примерами мульти моделей являются бэггинг, градиентный бустинг [89] и случайный лес [90]. В статье [106] предполагается, что вклад каждого эксперта в ответ зависит от объекта из набора данных.

Основной проблемой построения мульти моделей является то, что ансамбль зависит от начальной инициализации параметров. Для улучшения устойчивости мульти модели предлагается использовать вероятностную постановку задачи для поиска оптимальных параметров шлюзовой функции и параметров локальной модели. В данной работе задается априорное распределение на параметры локальных моделей, также, для повышения, предлагается учесть зависимость априорных распределений для разных моделей.

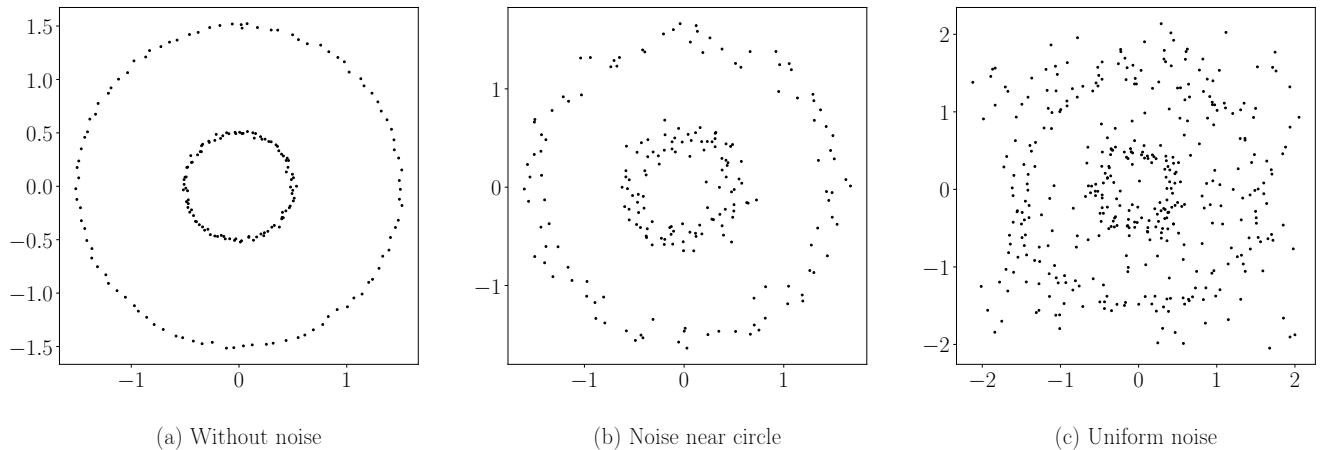


Рис. 1: Пример окружностей с разным уровнем шума: (а) окружности без шума; (б) окружности с зашумленным радиусом; (с) окружности с зашумленным радиусом, а также с равномерным шумом по всему изображению

В данной работе решается задача поиска окружностей на бинаризованном изображении. Предполагается, что радиусы окружностей различаются значимо, а также, что центры почти совпадают. Пример изображений показан на рис. 51а. В данной работе в качестве экспертов рассматриваются линейные модели — каждая модель аппроксимирует одну окружность. В качестве шлюзовой функции рассматривается двухслойная нейронная сеть.

Большое количество работ в области построения смеси экспертов посвящены выбору шлюзовой функции: используется softmax, процесс Дирихле [65], нейронная сеть [66] с функцией softmax на последнем слое. Ряд работ посвящены

выбору моделей в качестве отдельных экспертов. В работах [67, 68] в качестве модели эксперта рассматривается линейная модель. Работы [69, 88] рассматривают модель SVM в качестве модели эксперта. В работе [106] представлен обзор методов и моделей в задачах смеси экспертов.

Смесь экспертов имеет множество приложений в прикладных задачах. Работы [107, 91, 105] посвящены применению смеси экспертов в задачах прогнозирования временных рядов. В работе [94] предложен метод распознавания рукописных цифр. Метод распознания текстов при помощи смеси экспертов исследуется в работах [93], распознание речи [99, 100, 104]. В работе [103] исследуется смесь экспертов для задачи распознавания трехмерных движений человека. В [87] описаны работы по исследованию обнаружения радужки глаза на изображении. В работах [97, 98] в частности описаны методы выделения границ радужки и зрачка.

2.1 Постановка задачи аппроксимации параметров окружности

Задано бинарное изображение

$$\mathbf{M} \in \{0, 1\}^{m_1 \times m_2}, \quad (2.1)$$

где 1 — это черный пиксель, который принадлежит рассматриваемой фигуре на изображении, а 0 — белый пиксель, который является фоном изображения. Пример изображения показан на рис. 51а. Изображение \mathbf{M} отображается в множество координат $\mathbf{C} = \{x_i, y_i\}_{i=1}^N$. Координата (x_i, y_i) является координатой i -го черного пикселя на изображении \mathbf{M} :

$$\mathbf{C} \in \mathbb{R}^{N \times 2}, \quad (2.2)$$

где N — число черных пикселей.

Обозначим точку (x_0, y_0) центром окружности, а r радиусом окружности. Координаты $(x_i, y_i) \in \mathbf{C}$ это геометрическое место точек, которое удовлетворяет системе уравнений:

$$(x_i - x_0)^2 + (y_i - y_0)^2 = r^2 + \varepsilon_i, \quad \forall i \in \{1, \dots, N\}, \quad (2.3)$$

где $\varepsilon_i \in \mathcal{N}(0, \beta^{-1})$ является невязкой i -го уравнения, которая является следствием шума на изображении.

Раскрыв скобки получаем:

$$(2x_0) \cdot x_i + (2y_0) \cdot y_i + (r^2 - x_0^2 - y_0^2) \cdot 1 = x_i^2 + y_i^2 + \varepsilon_i. \quad (2.4)$$

Выражение (2.4) переписывается в задачу линейной регрессии следующим образом:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbf{R}^n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|, \quad \mathbf{X} = [\mathbf{C}, \mathbf{1}], \quad \mathbf{y} = [x_1^2 + y_1^2, x_2^2 + y_2^2, \dots, x_N^2 + y_N^2]^T. \quad (2.5)$$

Используя вектор параметров $\hat{\mathbf{w}} = [w_1, w_2, w_3]^T$ получим параметры окружности x_0, y_0, r :

$$x_0 = \frac{w_1}{2}, \quad y_0 = \frac{w_2}{2}, \quad r = \sqrt{w_3 + x_0^2 + y_0^2}. \quad (2.6)$$

Решение уравнения (2.5) находит параметры единственной окружности на изображении. В случае, когда на изображении несколько окружностей, предлагаются использовать смесь экспертов, которая состоит из линейных модели — экспертов. Каждый эксперт описывает одну окружность на изображении.

Обобщим подход аппроксимации одной окружности на изображении на случай, когда на изображении несколько окружностей. Пусть изображение состоит из K окружностей, тогда множество черных пикселей \mathbf{C} представляется в виде:

$$\mathbf{C} = \sqcup_{k=1}^K \mathbf{C}'_k, \quad (2.7)$$

где \mathbf{C}'_k множество точек принадлежащих k -й окружности. Множеству точек $\mathbf{C}'_k \subset \mathbf{C}$ соответствует задача линейной регрессии для выборки $\mathbf{X}'_k \subset \mathbf{X}, \mathbf{y}'_k \subset \mathbf{y}$. Модель \mathbf{g}_k аппроксимирующая выборку $\mathbf{X}'_k, \mathbf{y}'_k$ является локальной моделью для выборки \mathbf{X}, \mathbf{y} .

Определение 2.1. Модель \mathbf{g} называется локальной моделью для выборки \mathbf{U} , если \mathbf{g} аппроксимирует некоторое не пустое подмножество $\mathbf{U}' \subset \mathbf{U}$.

Определение 2.2. Мультимодель \mathbf{f} называется смесью экспертов, если:

$$\mathbf{f} = \sum_{k=1}^K \pi_k \mathbf{g}_k(\mathbf{w}_k), \quad \pi_k(\mathbf{x}, \mathbf{V}) : \mathbb{R}^{n \times |\mathbf{V}|} \rightarrow [0, 1], \quad \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) = 1, \quad (2.8)$$

где \mathbf{g}_k является k -й локальной моделью, π_k — шлюзовая функция, вектор \mathbf{w}_k является параметрами k -й локальной моделью, а \mathbf{V} — параметры шлюзовой функции.

В данной работе в качестве локальных моделей рассматриваются линейные модели. В качестве шлюзовой функции рассматривается двухслойный перцептрон:

$$\mathbf{g}_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x}, \quad \pi(\mathbf{x}, \mathbf{V}) = \text{softmax}(\mathbf{V}_1^\top \boldsymbol{\sigma}(\mathbf{V}_2^\top \mathbf{x})), \quad (2.9)$$

где $\mathbf{V} = \{\mathbf{V}_1, \mathbf{V}_2\}$ — множество параметров шлюзовой функции.

В статье предлагается использовать вероятностный подход для описания смеси экспертов. Вводится предположение, что \mathbf{y} является случайным вектором, который задается плотностью распределения $p(\mathbf{y}|\mathbf{X})$. Предполагается, что плотность распределения $p(\mathbf{y}|\mathbf{X}, \mathbf{f})$ аппроксимирует истинную плотность распределения $p(\mathbf{y}|\mathbf{X})$:

$$p(\mathbf{y}|\mathbf{X}, \mathbf{f}) = \prod_{i=1}^N \left(\sum_{k=1}^K \pi_k p_k(y_i | \mathbf{g}_k(\mathbf{x}_i)) \right), \quad (2.10)$$

где \mathbf{f} — это смесь экспертов, а $\mathbf{g}_k, \boldsymbol{\pi}$ определяются выражением (9.4).

Пусть \mathbf{w}_k является случайным вектором, который задается плотностью распределения $p^k(\mathbf{w}_k)$. Получим совместное распределения параметров локальных моделей и вектора ответов:

$$p(\mathbf{y}, \mathbf{W}|\mathbf{X}, \mathbf{V}) = \prod_{k=1}^K p^k(\mathbf{w}_k) \prod_{i=1}^N \left(\sum_{k=1}^K \pi_k p_k(y_i | \mathbf{w}_k, \mathbf{x}_i) \right), \quad (2.11)$$

где $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K\}$. Оптимальные параметры находятся при помощи максимизации правдоподобия:

$$\hat{\mathbf{V}}, \hat{\mathbf{W}} = \arg \max_{\mathbf{V}, \mathbf{W}} p(\mathbf{y}, \mathbf{W} | \mathbf{X}, \mathbf{V}). \quad (2.12)$$

2.2 Вероятностная постановка задачи смеси экспертов

Для построения смеси экспертов (9.3, 9.5), введем следующие вероятностные предположения о данных (2.5):

- 1) правдоподобие $p_k(y_i | \mathbf{w}_k, \mathbf{x}_i) = \mathcal{N}(y_i | \mathbf{w}_k^\top \mathbf{x}_i, \beta^{-1})$, где параметр β является уровнем шума,
- 2) априорное распределение параметров $p^k(\mathbf{w}_k) = \mathcal{N}(\mathbf{w}_k | \mathbf{w}_k^0, \mathbf{A}_k)$, где \mathbf{w}_k^0 — вектор размерности $n \times 1$, а \mathbf{A}_k — ковариационная матрица размерности $n \times n$,
- 3) регуляризация априорного распределения $p(\boldsymbol{\varepsilon}_{k,k'} | \boldsymbol{\Xi}) = \mathcal{N}(\boldsymbol{\varepsilon}_{k,k'} | \mathbf{0}, \boldsymbol{\Xi})$, где $\boldsymbol{\Xi}$ — ковариационная матрица, а $\boldsymbol{\varepsilon}_{k,k'} = \mathbf{w}_k^0 - \mathbf{w}_{k'}^0$.

Предположение 2) задает априорное предположения на вектора параметров локальных модели \mathbf{w}_k . Априорное распределение задает ограничения на локальную модель. Например, если $\mathbf{w}_k^0 = [0, 0, 1]$, то k -я локальная модель аппроксимирует окружность с параметрами $x_0 = 0, y_0 = 0, r = 1$ с большей вероятностью.

Предположения 3) задает регуляризацию априорных распределений. Данная регуляризация учитывает связь между априорными ограничениями разных локальных моделей. Например, если $\text{diag}(\boldsymbol{\Xi}) = [0.001, 0.001, 1]$, то центры разных окружностей совпадают.

Используя предположения 1), 2), 3) и выражение (9.5) получаем полное правдоподобие:

$$p(\mathbf{y}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) = \prod_{i=1}^N \left(\sum_{k=1}^K \pi_k \mathcal{N}(y_i | \mathbf{w}_k^\top \mathbf{x}_i, \beta^{-1}) \right) \cdot \\ \cdot \prod_{k=1}^K \mathcal{N}(\mathbf{w}_k | \mathbf{w}_k^0, \mathbf{A}_k) \cdot \prod_{k,k'=1}^K \mathcal{N}(\boldsymbol{\varepsilon}_{k,k'} | \mathbf{0}, \boldsymbol{\Xi}), \quad (2.13)$$

где $\mathbf{A} = \{\mathbf{A}_1, \dots, \mathbf{A}_K\}$.

Введем бинарную матрицу \mathbf{Z} . Элемент матрицы z_{ik} равно 1 тогда и только тогда, когда i -й объект аппроксимируется k -й локальной моделью. Подставляя

бинарную матрицу \mathbf{Z} в выражении (2.13), а также взяв логарифм получаем:

$$\begin{aligned} \log p(\mathbf{y}, \mathbf{Z}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) &= \\ &= \sum_{i=1}^N \sum_{k=1}^K z_{ik} \left[\log \pi_k(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} (y_i - \mathbf{w}_k^\top \mathbf{x}_i)^2 + \frac{1}{2} \log \frac{\beta}{2\pi} \right] + \\ &+ \sum_{k=1}^K \left[-\frac{1}{2} (\mathbf{w}_k - \mathbf{w}_k^0)^\top \mathbf{A}_k^{-1} (\mathbf{w}_k - \mathbf{w}_k^0) + \frac{1}{2} \log \det \mathbf{A}_k^{-1} - \frac{n}{2} \log 2\pi \right] + \\ &+ \sum_{k=1}^K \sum_{k'=1}^K \left[-\frac{1}{2} (\mathbf{w}_k^0 - \mathbf{w}_{k'}^0)^\top \boldsymbol{\Xi}^{-1} (\mathbf{w}_k^0 - \mathbf{w}_{k'}^0) + \frac{1}{2} \log \det \boldsymbol{\Xi} - \frac{n}{2} \log 2\pi \right]. \end{aligned} \quad (2.14)$$

Получаем новую задачу оптимизации обоснованности. Функция обоснованности получается при интегрировании выражения (2.14) по параметрам \mathbf{W}, \mathbf{Z} :

$$\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta = \arg \max_{\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta} \int_{\mathbf{W}, \mathbf{Z}} \log p(\mathbf{y}, \mathbf{Z}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) d\mathbf{W} d\mathbf{Z}. \quad (2.15)$$

Рассмотрим вариационную плотность $q(\mathbf{W}, \mathbf{Z})$ для параметров \mathbf{W}, \mathbf{Z} . Тогда функция обоснованности принимает следующий вид:

$$\begin{aligned} \log p(\mathbf{y} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) &= \int_{\mathbf{W}, \mathbf{Z}} q(\mathbf{W}, \mathbf{Z}) \log p(\mathbf{y} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) d\mathbf{W} d\mathbf{Z} = \\ &= \int_{\mathbf{W}, \mathbf{Z}} q(\mathbf{W}, \mathbf{Z}) \log \frac{p(\mathbf{y}, \mathbf{W}, \mathbf{Z} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)}{p(\mathbf{W}, \mathbf{Z} | \mathbf{y}, \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)} d\mathbf{W} d\mathbf{Z} = \\ &= \int_{\mathbf{W}, \mathbf{Z}} q(\mathbf{W}, \mathbf{Z}) \log \frac{p(\mathbf{y}, \mathbf{W}, \mathbf{Z} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) q(\mathbf{W}, \mathbf{Z})}{p(\mathbf{W}, \mathbf{Z} | \mathbf{y}, \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) q(\mathbf{W}, \mathbf{Z})} d\mathbf{W} d\mathbf{Z} = \\ &= \int_{\mathbf{W}, \mathbf{Z}} q(\mathbf{W}, \mathbf{Z}) \frac{p(\mathbf{y}, \mathbf{W}, \mathbf{Z} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)}{q(\mathbf{W}, \mathbf{Z})} d\mathbf{W} d\mathbf{Z} + \\ &+ \int_{\mathbf{W}, \mathbf{Z}} q(\mathbf{W}, \mathbf{Z}) \frac{q(\mathbf{W}, \mathbf{Z})}{p(\mathbf{W}, \mathbf{Z} | \mathbf{y}, \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)} d\mathbf{W} d\mathbf{Z} = \\ &= \mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta) + D_{KL}(q(\mathbf{W}, \mathbf{Z}) || p(\mathbf{W}, \mathbf{Z} | \mathbf{y}, \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)) \end{aligned} \quad (2.16)$$

Используя (2.16) получаем нижнюю оценку обоснованности:

$$\log p(\mathbf{y} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) \geq \mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta), \quad (2.17)$$

где $\mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)$ называется нижней оценкой обоснованности.

Используем ЕМ–алгоритм [92, 86] для решения оптимизационной задачи (2.15). Заметим, что ЕМ–алгоритм вместо оптимизации $\log p(\mathbf{y} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta)$ оптимизирует нижнюю оценку $\mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)$.

E-шаг. Е-шаг решает следующую оптимизационную задачу:

$$\mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta) \rightarrow \max_{q(\mathbf{W}, \mathbf{Z})}, \quad (2.18)$$

где параметры $\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta$ являются зафиксированными.

Пусть совместное распределение $q(\mathbf{Z}, \mathbf{W})$ удовлетворяет условию независимости $q(\mathbf{Z}, \mathbf{W}) = q(\mathbf{Z})q(\mathbf{W})$ [86]. Далее символом \propto обозначим то, что обе стороны выражения равны с точностью до аддитивной константы. Сначала найдем распределение $q(\mathbf{Z})$:

$$\begin{aligned} \log q(\mathbf{Z}) &= \mathbb{E}_{q/\mathbf{Z}} \log p(\mathbf{y}, \mathbf{Z}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) \propto \\ &\propto \sum_{i=1}^N \sum_{k=1}^K z_{ik} \left[\log \pi_k(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} (y_i^2 - \mathbf{x}_i^\top \mathbf{E}\mathbf{w}_k + \mathbf{x}_i^\top \mathbf{E}\mathbf{w}_k \mathbf{w}_k^\top \mathbf{x}_i) + \frac{1}{2} \log \frac{\beta}{2\pi} \right] \quad (2.19) \\ p(z_{ik} = 1) &= \frac{\exp(\log \pi_k(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} (\mathbf{x}_i^\top \mathbf{E}\mathbf{w}_k \mathbf{w}_k^\top \mathbf{x}_i - \mathbf{x}_i^\top \mathbf{E}\mathbf{w}_k))}{\sum_{k'=1}^K \exp(\log \pi_{k'}(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} (\mathbf{x}_i^\top \mathbf{E}\mathbf{w}_{k'} \mathbf{w}_{k'}^\top \mathbf{x}_i - \mathbf{x}_i^\top \mathbf{E}\mathbf{w}_{k'}))}. \end{aligned}$$

Используя выражения (2.19) получаем, что распределение $q(z_{ik})$ является бернулевским распределением с параметром z_{ik} , которое задается выражением (2.19). Далее найдем распределение $q(\mathbf{W})$:

$$\begin{aligned} \log q(\mathbf{W}) &= \mathbb{E}_{q/\mathbf{W}} \log p(\mathbf{y}, \mathbf{Z}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) \propto \\ &\propto \sum_{i=1}^N \sum_{k=1}^K \mathbb{E} z_{ik} \left[\log \pi_k(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} (y_i - \mathbf{w}_k^\top \mathbf{x}_i)^2 + \frac{1}{2} \log \frac{\beta}{2\pi} \right] + \\ &+ \sum_{k=1}^K \left[-\frac{1}{2} (\mathbf{w}_k - \mathbf{w}_k^0)^\top \mathbf{A}_k^{-1} (\mathbf{w}_k - \mathbf{w}_k^0) + \frac{1}{2} \log \det \mathbf{A}_k^{-1} - \frac{n}{2} \log 2\pi \right] \quad (2.20) \\ &\propto \sum_{k=1}^K \left[\mathbf{w}_k^\top \left(\mathbf{A}_k^{-1} \mathbf{w}_k^0 + \beta \sum_{i=1}^N \mathbf{x}_i y_i \mathbb{E} z_{ik} \right) - \frac{1}{2} \mathbf{w}_k^\top \left(\mathbf{A}_k^{-1} + \beta \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^\top \right) \mathbf{w}_k \right]. \end{aligned}$$

Используя выражение (2.20) получаем, что распределение $q(\mathbf{w}_k)$ является нормальным распределением со средним \mathbf{m}_k и ковариационной матрицей \mathbf{B}_k :

$$\mathbf{m}_k = \mathbf{B}_k \left(\mathbf{A}_k^{-1} \mathbf{w}_k^0 + \beta \sum_{i=1}^N \mathbf{x}_i y_i \mathbb{E} z_{ik} \right), \quad \mathbf{B}_k = \left(\mathbf{A}_k^{-1} + \beta \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^\top \mathbb{E} z_{ik} \right)^{-1}. \quad (2.21)$$

M-шаг. M-шаг решает следующую оптимизационную задачу:

$$\mathcal{L}(q, \mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta) \rightarrow \max_{\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta}, \quad (2.22)$$

где $q(\mathbf{W}, \mathbf{Z})$ является известной плотностью распределения. Распределение $q(\mathbf{Z}, \mathbf{W})$ является фиксированным, в то время как вариационная нижняя оценка $\mathcal{L}(\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)$ максимизируется по параметрам $\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta$:

$$\begin{aligned} \mathcal{L}(\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta) &= \mathbb{E}_q \log p(\mathbf{y}, \mathbf{Z}, \mathbf{W} | \mathbf{X}, \mathbf{V}, \mathbf{A}, \mathbf{W}^0, \boldsymbol{\Xi}, \beta) = \\ &= \sum_{i=1}^N \sum_{k=1}^K \mathbb{E} z_{ik} \left[\log \pi_k(\mathbf{x}_i, \mathbf{V}) - \frac{\beta}{2} \mathbb{E} (y_i - \mathbf{w}_k^\top \mathbf{x}_i)^2 + \frac{1}{2} \log \frac{\beta}{2\pi} \right] + \\ &+ \sum_{k=1}^K \left[-\frac{1}{2} \mathbb{E} (\mathbf{w}_k - \mathbf{w}_k^0)^\top \mathbf{A}_k^{-1} (\mathbf{w}_k - \mathbf{w}_k^0) + \frac{1}{2} \log \det \mathbf{A}_k^{-1} - \frac{n}{2} \log 2\pi \right] + \quad (2.23) \\ &+ \sum_{k=1}^K \sum_{k'=1}^K \left[-\frac{1}{2} (\mathbf{w}_k^0 - \mathbf{w}_{k'}^0)^\top \boldsymbol{\Xi}^{-1} (\mathbf{w}_k^0 - \mathbf{w}_{k'}^0) + \frac{1}{2} \log \det \boldsymbol{\Xi} - \frac{n}{2} \log 2\pi \right]. \end{aligned}$$

Во-первых, для нахождения оптимального параметра \mathbf{V} используется градиентный метод оптимизации, который сходится к некоторому локальному экстремуму. Во вторых, используя выражения (2.23) получаем оптимальное значения параметра \mathbf{A}_k

$$\begin{aligned} \frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)}{\partial \mathbf{A}_k^{-1}} &= \frac{1}{2} \mathbf{A}_k - \frac{1}{2} \mathbf{E} (\mathbf{w}_k - \mathbf{w}_k^0) (\mathbf{w}_k - \mathbf{w}_k^0)^T = 0, \\ \mathbf{A}_k &= \mathbf{E} \mathbf{w}_k \mathbf{w}_k^T - \mathbf{w}_k^0 \mathbf{E} \mathbf{w}_k^T - \mathbf{E} \mathbf{w}_k \mathbf{w}_k^{0T} + \mathbf{w}_k^0 \mathbf{w}_k^{0T}. \end{aligned} \quad (2.24)$$

Аналогично получаем оптимальные значения для параметра β и для параметров \mathbf{w}_k^0

$$\begin{aligned} \frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)}{\partial \beta} &= \sum_{k=1}^K \sum_{i=1}^N \left(\frac{1}{\beta} \mathbf{E} z_{ik} - \frac{1}{2} \mathbf{E} z_{ik} [y_i^2 - 2y_i \mathbf{x}_i^T \mathbf{E} \mathbf{w}_k + \mathbf{x}_i^T \mathbf{w}_k \mathbf{w}_k^T \mathbf{x}_i] \right) = 0, \\ \frac{1}{\beta} &= \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K [y_i^2 - 2y_i \mathbf{x}_i^T \mathbf{E} \mathbf{w}_k + \mathbf{x}_i^T \mathbf{E} \mathbf{w}_k \mathbf{w}_k^T \mathbf{x}_i] \mathbf{E} z_{ik}. \end{aligned} \quad (2.25)$$

$$\begin{aligned} \frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{W}^0, \mathbf{A}, \beta)}{\partial \mathbf{w}_k^0} &= \mathbf{A}_k^{-1} (\mathbf{E} \mathbf{w}_k - \mathbf{w}_k^0) + \mathbf{\Xi} \sum_{k'=1}^K [\mathbf{w}_{k'}^0 - \mathbf{w}_k^0] = 0, \\ \mathbf{w}_k^0 &= [\mathbf{A}_k^{-1} + (K-1) \mathbf{\Xi}]^{-1} \left(\mathbf{A}_k^{-1} \mathbf{E} \mathbf{w}_k + \mathbf{\Xi} \sum_{k'=1, k' \neq k}^K \mathbf{w}_{k'}^0 \right). \end{aligned} \quad (2.26)$$

Выражения (2.19–2.26) задают итеративную процедуру, которая сходится к некоторому локальному максимуму оптимизационной задачи (2.15).

2.3 Вычислительный эксперимент по анализу качества аппроксимации радужки глаза смесью экспертов

Для анализа качества различных мультимоделей для аппроксимации окружности проводится вычислительный эксперимент. В эксперимент рассматриваются следующие мультимодели: мультимодель \mathbf{f}_1 без использования априорных распределений, мультимодель \mathbf{f}_2 , которая использует априорные распределения (2.28) для параметров и мультимодель \mathbf{f}_3 , которая использует регуляризацию априорных распределений. Точность аппроксимации мультимодели \mathbf{f}_i задается следующим образом:

$$\mathcal{S}_{\mathbf{f}_i} = \sum_{k=1}^K (x_0^k - x_{\text{pr}}^k)^2 + (y_0^k - y_{\text{pr}}^k)^2 + (r^k - r_{\text{pr}}^k)^2, \quad (2.27)$$

где x_0^k, y_0^k, r^k является истинным центром и радиусом для k -й окружности, $x_{\text{pr}}^k, y_{\text{pr}}^k, r_{\text{pr}}^k$ является предсказанным центром и радиусом для k -й окружности.

Для сравнение модель с разными вероятностными предположениями используется правдоподобие (4.9). В вычислительном эксперименте используется следующее априорное распределение:

$$p^1(\mathbf{w}_1) \sim \mathcal{N}(\mathbf{w}_1^0, \mathbf{I}), \quad p^2(\mathbf{w}_2) \sim \mathcal{N}(\mathbf{w}_2^0, \mathbf{I}), \quad (2.28)$$

где $\mathbf{w}_1^0 = [0, 0, 0.1]$, $\mathbf{w}_2^0 = [0, 0, 2]$.

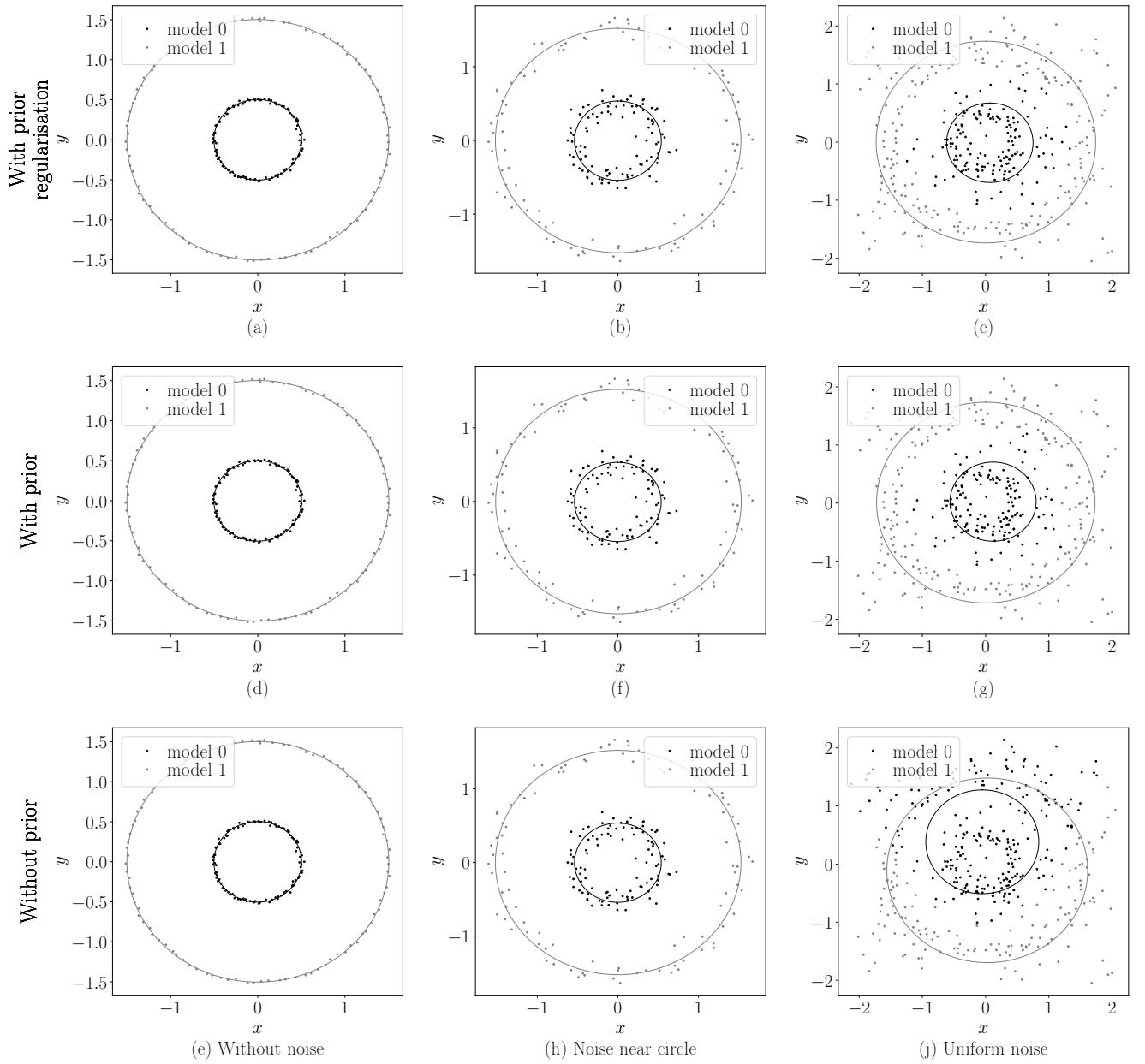


Рис. 2: Мульти модель в зависимости от разных априорных предположений и в зависимости от разного уровня шума: (а)–(с) модель с регуляризацией априорных распределений; (д)–(г) модель с заданными априорными распределениями на параметрах локальных моделей; (е)–(ж) модель без заданных априорных предположений

Синтетические данные с разным типом шума в изображении. В вычислительном эксперименте сравнивается качество следующих мульти моделей f_1, f_2, f_3 на синтетических данных. Синтетические данные являются двумя концентрическими окружностями с разным уровнем шума. Выборка Synthetic 1 является изображением без шума, выборка Synthetic 2 изображение с зачумлённым радиусом окружности, а выборка Synthetic 3 — изображение с равномерным шумом. На рис. 2 показаны результаты для мельтимоделей f_1, f_2, f_3 . Все модели оптимизировались при помощи 50 итераций ЕМ-алгоритма. Мультимо-

дели $\mathbf{f}_2, \mathbf{f}_3$ аппроксимируют окружности лучше чем мульти модель \mathbf{f}_1 . В табл. 2 показано качество аппроксимации (2.27) для всех мульти моделей.

Таблица 1: Качество аппроксимации (2.27) для всех мульти моделей

Выборка	$\mathcal{S}_{\mathbf{f}_1}$	$\mathcal{S}_{\mathbf{f}_2}$	$\mathcal{S}_{\mathbf{f}_3}$
Synthetic 1	10^{-5}	10^{-5}	10^{-5}
Synthetic 2	0.6	10^{-3}	10^{-3}
Synthetic 3	0.6	10^{-3}	10^{-3}

Анализ сходимости на синтетической выборке. Данная часть эксперимента анализирует качество сходимости ЕМ-алгоритма для разных мульти моделей $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$. Анализ всех мульти моделей проводиться на выборке Synthetic 3.

На рис. 3 показана зависимость предсказано центра и радиуса в зависимости от номера итерации ЕМ-алгоритма. Мульти модель \mathbf{f}_2 , которая использует априорное распределение аппроксимирует окружность лучше мульти модели \mathbf{f}_1 , которая не использует никакого априорного распределения. Мульти модель \mathbf{f}_3 , которая использует регуляризатор априорных распределений является более стабильной, чем мульти модель \mathbf{f}_2 .

На рис. 4 показана зависимость логарифма правдоподобия (4.9) от номера итерации ЕМ-алгоритма. Логарифм правдоподобия мульти модели $\mathbf{f}_2, \mathbf{f}_3$ растет быстрее чем логарифм правдоподобия мульти модели \mathbf{f}_1 . После 20-й итерации все мульти модели имеют одинаковое правдоподобие.

На рис. 5-7 показан процесс сходимости для разных мульти моделей $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$. На рис. 7 показана мульти модель \mathbf{f}_1 , которая аппроксимирует окружности не верно. На рис. 5-6 показаны мульти модели $\mathbf{f}_2, \mathbf{f}_3$, которые аппроксимируют окружности верно.

Вычислительный эксперимент показывает, что мульти модели $\mathbf{f}_2, \mathbf{f}_3$, которые используют априорные распределения на параметры экспертов аппроксимируют окружности лучше чем мульти модель \mathbf{f}_1 , которая работает без априорных распределений.

Анализ мульти моделей в зависимости от уровня шума. Данная часть эксперимента анализирует зависимость разных мульти моделей $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$ от уровня шума. Анализ всех мульти моделей проводиться на выборке Synthetic 1, с добавлением разного уровня шума. Минимальный уровень шума равен 0, когда числа шумовых точек равно 0. Максимальный уровень шума равен 1, когда число шумовых точек равно числу точек на изображении. На рис. 8 показан график зависимости центра окружности и ее радиус в зависимости от уровня шума. Из графика видно, что радиус окружности увеличивается при увеличении уровня шума. Мульти модели $\mathbf{f}_2, \mathbf{f}_3$ аппроксимируют центр окружности

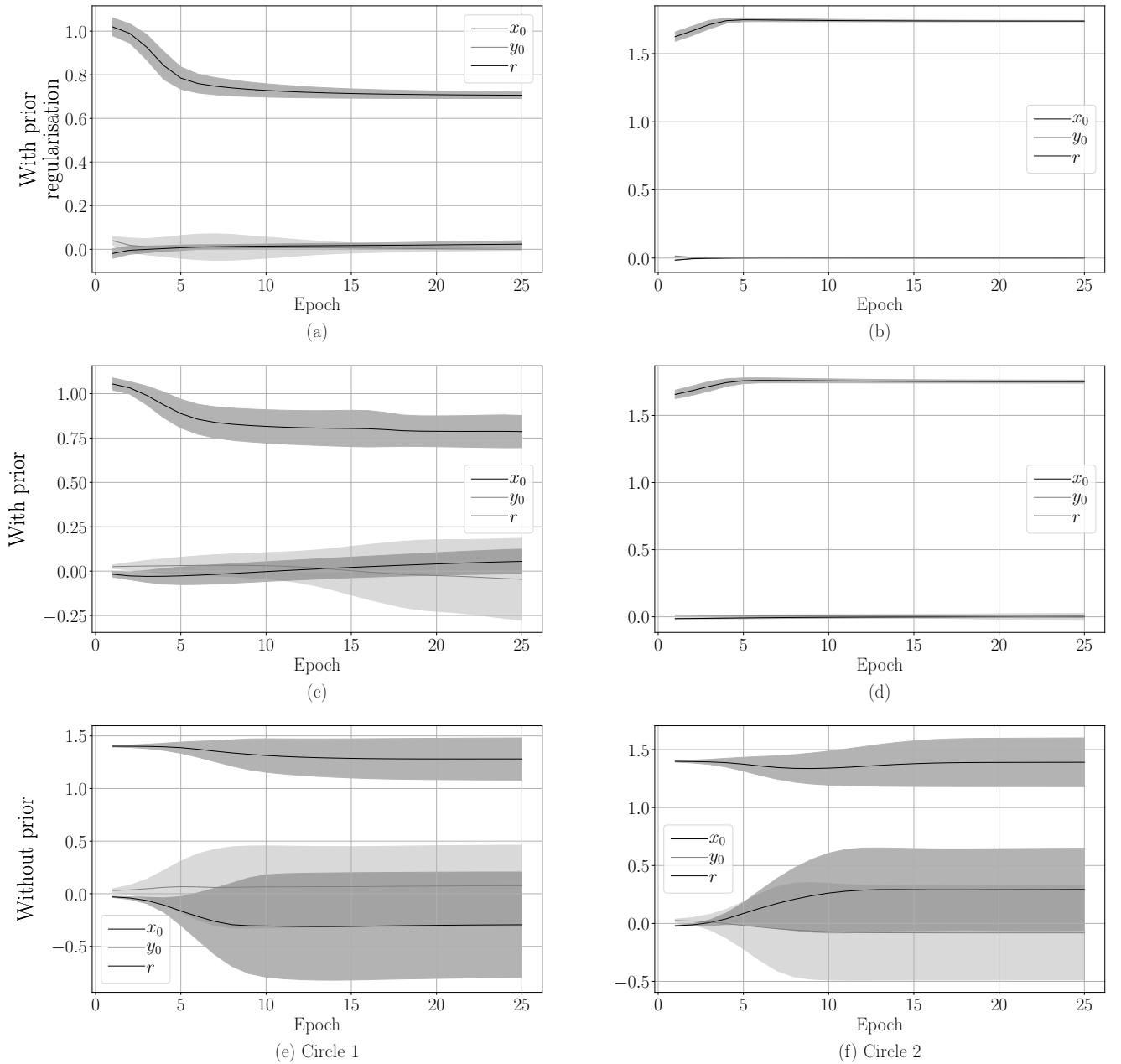


Рис. 3: График зависимости центра и радиуса окружностей от номера итерации: (a)–(b) модель с регуляризацией априорных распределений; (c)–(d) модель с заданными априорными распределениями на параметры моделей; (e)–(f) модель без задания априорных распределений

верно, но мульти модель f_3 более устойчива к шуму . На рис. 9 показана зависимость логарифма правдоподобия (4.9) от уровня шума. Из графика видно, что логарифм правдоподобия (4.9) эквивалентный для всех мульти моделей, но на рис. 8 видно, что качество аппроксимации (2.27) зависит от мульти модели. Данная часть вычислительного эксперимента показывает, что мульти модель f_3 с регуляризацией априорного распределения является более устойчива к шуму, чем остальные.

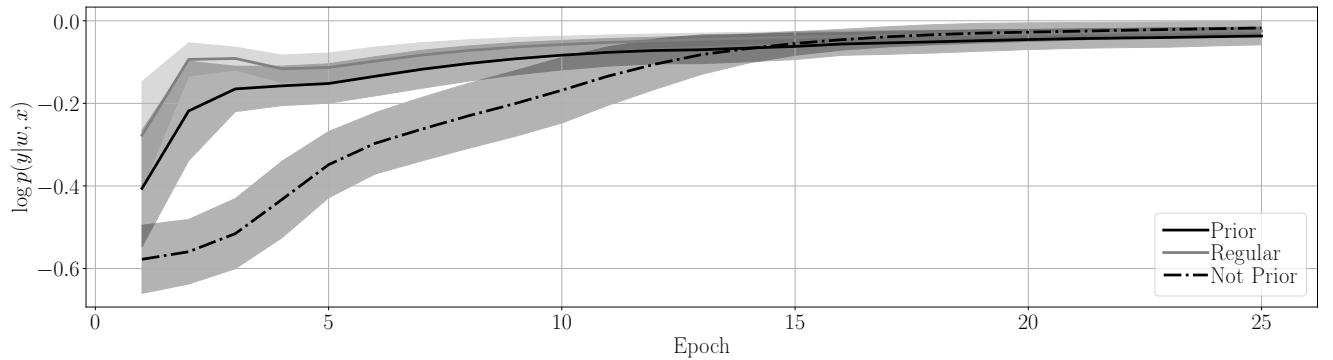


Рис. 4: График зависимости логарифма правдоподобия (4.9) от номера итерации.

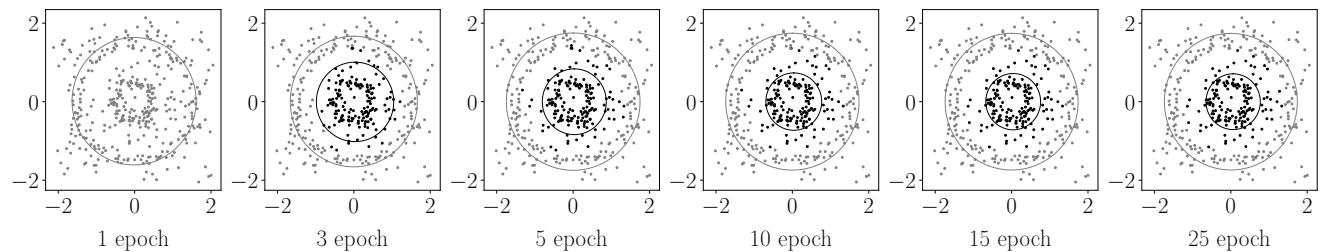


Рис. 5: Визуализации процесса сходимости мультимодели с использованием априорной регуляризации.

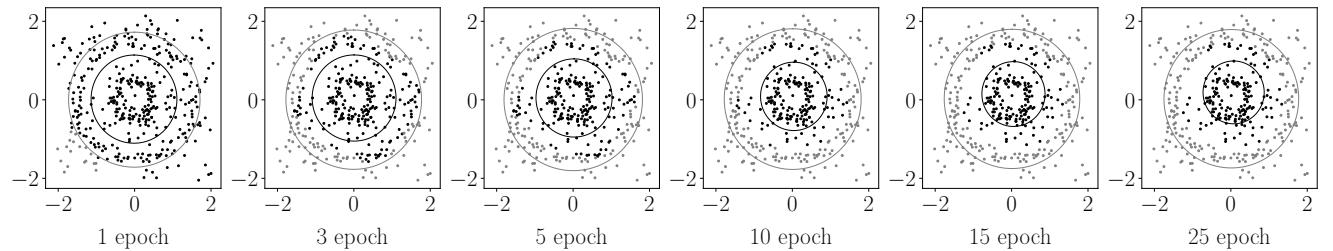


Рис. 6: Визуализации процесса сходимости мультимодели с использованием априорного распределения.

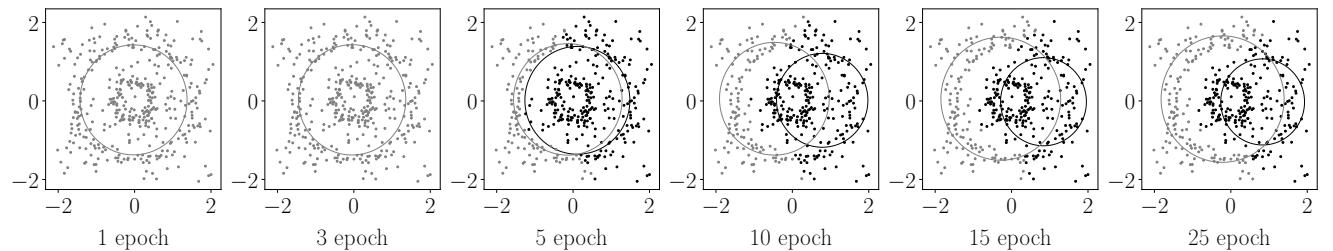


Рис. 7: Визуализации процесса сходимости мультимодели без использования априорного распределения.

Реальные данные. Данная часть эксперимента анализирует разные мультимодели f_1, f_2, f_3 на реальной выборке. На рис. 10 показан результат работы разных мультимоделей. Мультимодель f_1 не верно аппроксимирует меньшую

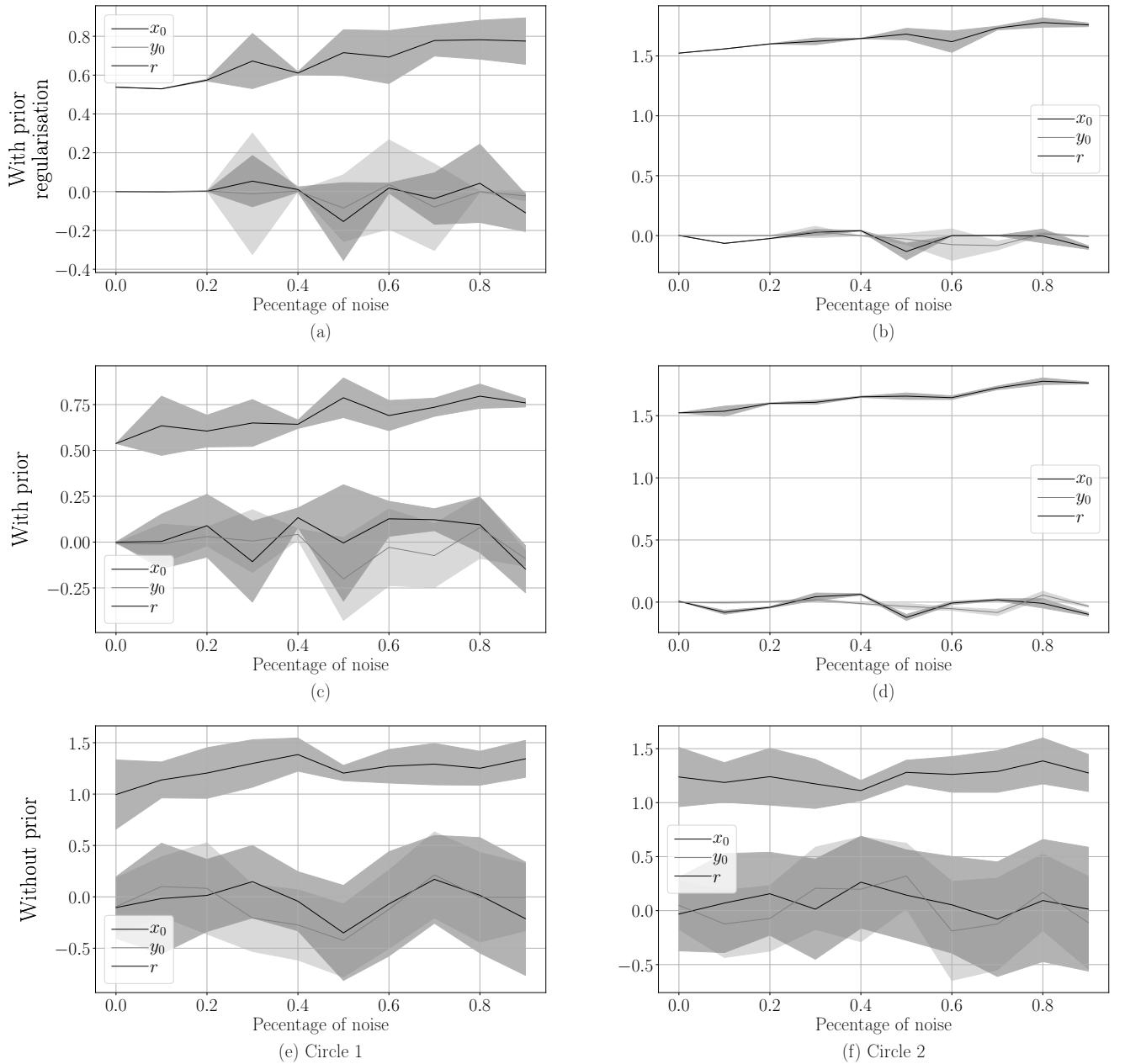


Рис. 8: график зависимости центра и радиуса окружностей от номера итерации: (a)–(b) модель с регуляризацией априорных распределений; (c)–(d) модель с заданными априорными распределениями на параметры моделей; (e)–(f) модель без задания априорных распределений

окружность. Мультимодели $\mathbf{f}_2, \mathbf{f}_3$ аппроксимируют обе окружности верно.

На рис. 11-13 показан процесс аппроксимации для разных мультимоделей $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$.

Данная часть эксперимента показывает, что мультимодели $\mathbf{f}_2, \mathbf{f}_3$ аппроксимируют окружности на реальных изображениях лучше, чем мультимодель \mathbf{f}_1 .

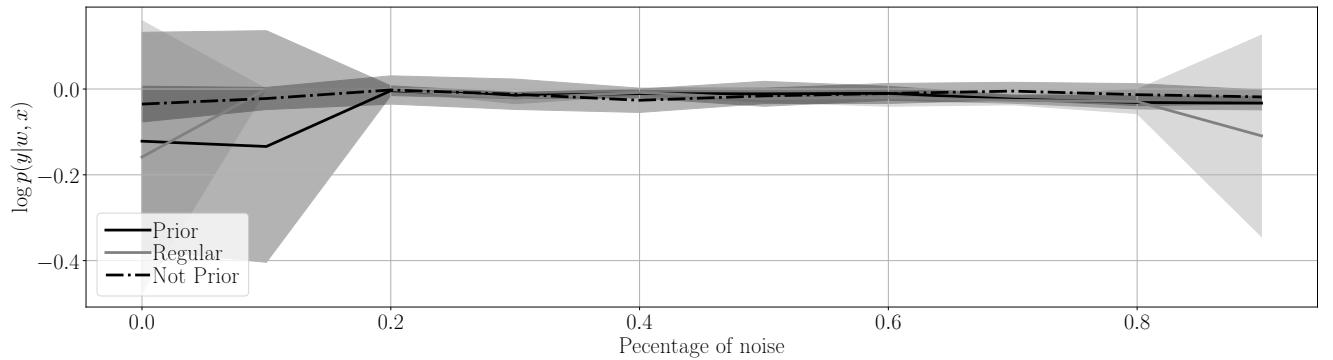


Рис. 9: График зависимости логарифма правдоподобия (4.9) от уровня шума.

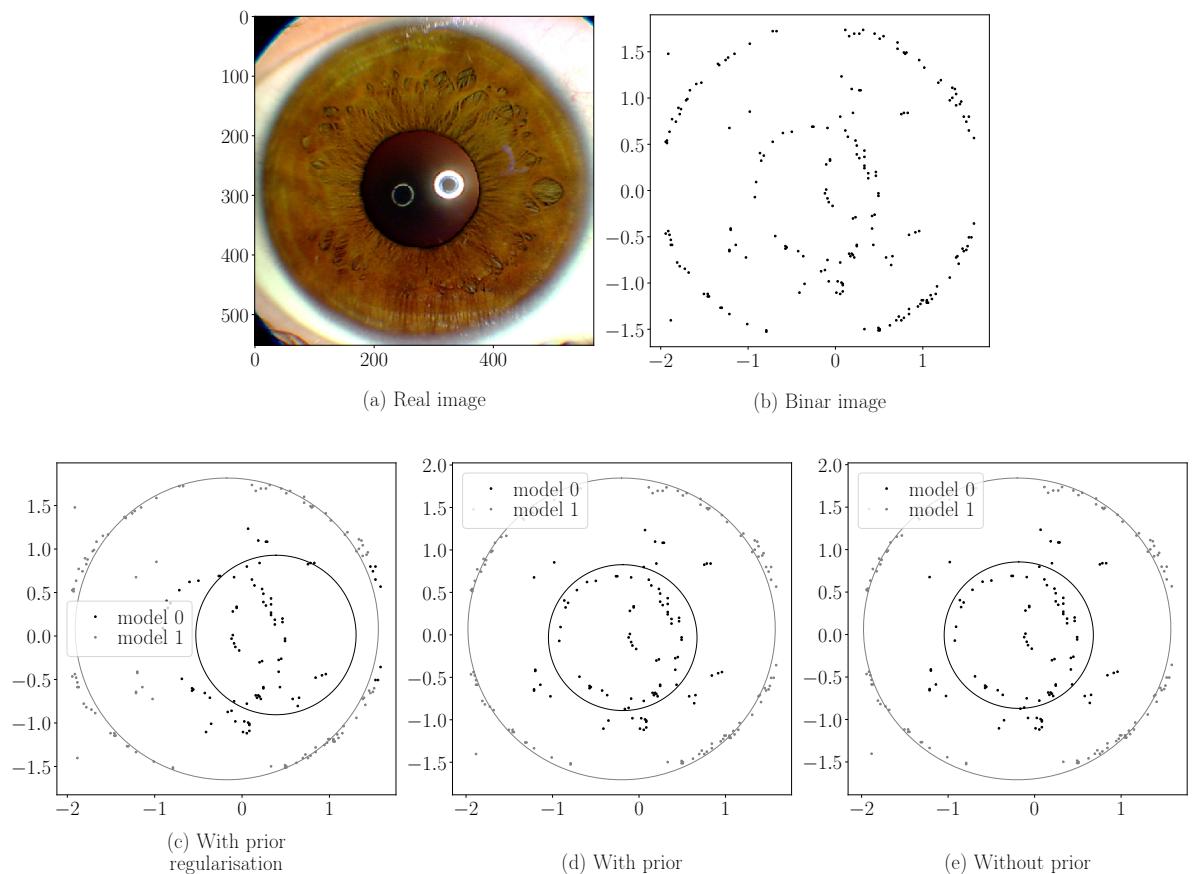


Рис. 10: Мульти модель в зависимости от разных априорных предположений на реальном изображении: (a) исходное изображение; (b) бинаризованное изображение; (c) мульти модель без априорных предположений; (d) мульти модель с априорными распределениями на параметрах локальных моделей; (e) мульти модель с регуляризацией на априорных распределениях параметров локальных моделей.

3 Модели привилегированного обучения и дистилляции

Повышение точности аппроксимации в задачах машинного обучения влечет за собой повышение сложности моделей и как следствие снижает их интерпрети-

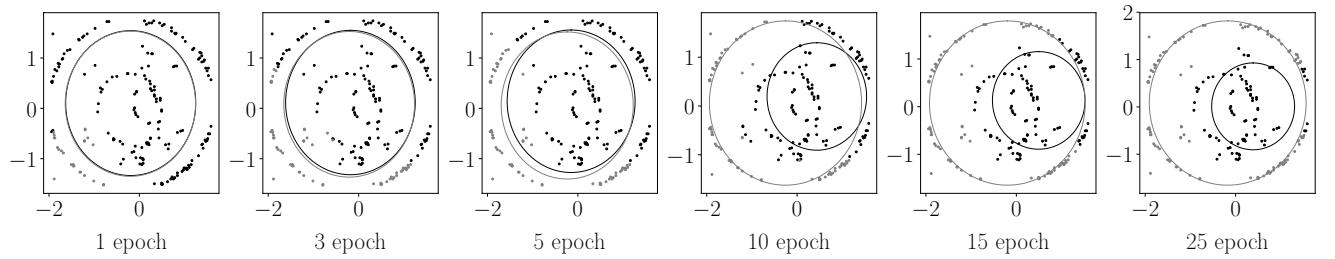


Рис. 11: Визуализации процесса сходимости мультимодели без использования априорного распределения.

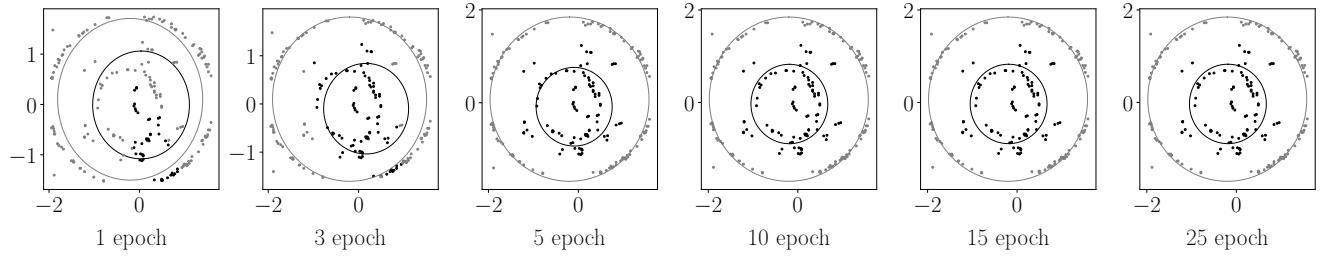


Рис. 12: Визуализации процесса сходимости мультимодели с использованием априорного распределением.

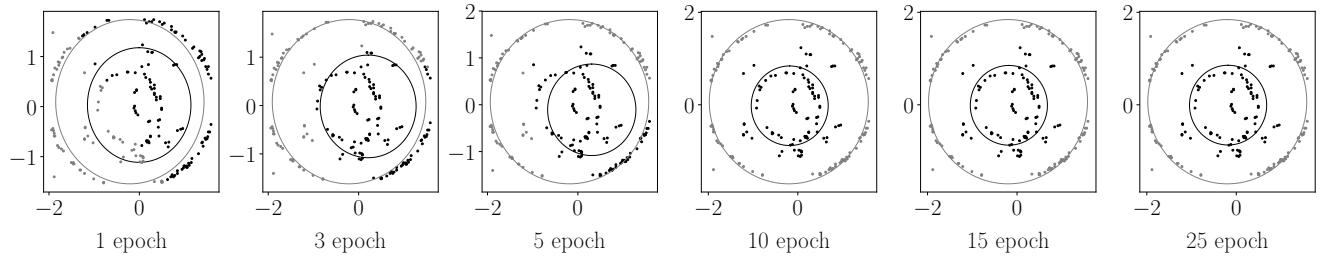


Рис. 13: Визуализации процесса сходимости мультимодели с использованием априорной регуляризации.

руемость. Примером такого усложнения являются следующие модели: трансформеры [143], BERT [144], ResNet [148] а также ансамбли этих моделей.

При построении модели машинного обучения используется два свойства: сложность модели и точность аппроксимации модели. Сложность влияет на время, которое модель требуется для принятия решения, а также на интерпретируемость модели, следовательно модель которая имеют меньшую сложность является более предпочтительной [149]. С другой стороны точность аппроксимации модели нужно максимизировать. В данной работе рассматривается метод *дистилляции* модели. Данные метод позволяет строить новые модели на основе ранее обученных моделей.

Определение 3.1. *Дистилляция модели — уменьшение сложности модели путем выбора модели в множестве более простых моделей с использованием ответов более сложной модели.*

В работе [150] Дж. Е. Хинтоном рассматривается метод дистилляции мо-

делей машинного обучения для задачи классификации. В работе проведен ряд экспериментов, в которых проводилась дистилляция моделей для разных задач машинного обучения. Эксперимент на выборке MNIST [151], в котором избыточно сложно нейросеть была дистиллирована в нейросеть меньшей сложности. Эксперимент по Speech Recognition, в котором ансамбль моделей был *дистиллирован* в одну модель. Также в работе [150] был проведен эксперимент по обучению экспертных моделей на основе одной большой модели.

Определение 3.2. *Привилегированная информация — множество признаков, которые доступны только в момент выбора модели, но не в момент тестирования.*

В работе [162] В. Н. Вапником введено понятия *привилегированной информации*. В работе [163] метод дистилляции [150] используется вместе с привилегированным обучением [162]. В предложенном методе на первом этапе обучается модель *учителя* в пространстве привилегированной информации, после чего обучается модель *ученика* в исходном признаковом пространстве используя *дистилляцию* [150]. Для обучения строится функция ошибки специального вида, анализируемая в данной работе. Эта функция состоит из нескольких слагаемых, включая ошибки учителя, ученика и регуляризирующие элементы. Первые варианты подобной функции ошибки были предложены А. Г. Ивахненко [154].

Определение 3.3. *Учитель — фиксируемая модель, ответы которой используются при выборе модели ученика.*

Определение 3.4. *Ученик — модель, которая выбирается согласно какого-либо критерия.*

В данной работе предлагается рассмотреть вероятностный подход к решению задачи дистилляции модели и задачи привилегированного обучения. Подход обобщается на случай, когда привилегированная информация доступна не для всех объектов из обучающей выборки. В рамках вероятностного подхода предлагается анализ и обобщение функции ошибки [150, 163]. Рассматриваются частные задачи классификации и регрессии [154].

В рамках вычислительного эксперимента анализируются методы использующие и не использующие модель учителя при обучение модели ученика. Для анализа используются реальные выборки для задачи классификации изображений FashionMNIST [155] и для задачи классификации текстов Twitter Sentiment Analysis [156]. Выборка FashionMNIST использовалась вместо общепринятой выборки MNIST, так как последняя имеет приемлемое качество аппроксимации даже для линейного классификатора. Вычислительный эксперимент использует модели разной сложности: линейная модель, полносвязная нейронная сеть, сверточная нейронная сеть [157], модель Bi-LSTM [158] и модель BERT [144].

3.1 Постановка задачи обучения с учителем: Хинтона и Вапника

Задано множество объектов Ω и множество целевых переменных \mathbb{Y} . Множество $\mathbb{Y} = \{1, \dots, K\}$ для задачи классификации, где K число классов, множество $\mathbb{Y} = \mathbb{R}$ для задачи регрессии. Для каждого объекта из $\omega_i \in \Omega$ задана целевая переменная $y_i = y(\omega_i)$. Множество целевых переменных для всех объектов обозначим \mathbf{Y} . Для множества Ω задано отображение в некоторое признаковое пространство \mathbb{R}^n :

$$\varphi : \Omega \rightarrow \mathbb{R}^n, \quad |\Omega| = m, \quad (3.1)$$

где n размерность признакового пространства, а m количество объектов в множестве Ω . Отображение φ отображает объект $\omega_i \in \Omega$ в соответствующий ему вектор признаков $\mathbf{x}_i = \varphi(\omega_i)$. Пусть для объектов $\Omega^* \subset \Omega$ задана привилегированная информация:

$$\varphi^* : \Omega^* \rightarrow \mathbb{R}^{n^*}, \quad |\Omega^*| = m^*, \quad (3.2)$$

где $m^* \leq m$ — число объектов с привилегированной информацией, n^* — число признаков в пространстве привилегированной информации. Отображение φ^* отображает объект $\omega_i \in \Omega^*$ в соответствующий ему вектор признаков $\mathbf{x}_i^* = \varphi^*(\omega_i)$.

Множество индексов объектов, для которых известна привилегированная информация, обозначим \mathcal{I} :

$$\mathcal{I} = \{1 \leq i \leq m \mid \text{для } i\text{-го объекта задана привилегированная информация}\}, \quad (3.3)$$

а множество индексов объектов, для которых не известна привилегированная информация, обозначим $\{1, \dots, m\} \setminus \mathcal{I} = \bar{\mathcal{I}}$.

Пусть на множестве привилегированных признаков задана функция учителя $\mathbf{f}(\mathbf{x}^*)$:

$$\mathbf{f} : \mathbb{R}^{n^*} \rightarrow \mathbb{Y}^*, \quad (3.4)$$

где $\mathbb{Y}^* = \mathbb{Y}$ для задачи регрессии и \mathbb{Y}^* является единичным симплексом \mathcal{S}_K в пространстве размерности K для задачи классификации. Модель учителя \mathbf{f} ставит объекты \mathbf{X}^* в соответствие объектам \mathbf{S} , то есть $\mathbf{f}(\mathbf{x}_i^*) = \mathbf{s}_i$.

Требуется выбрать модель ученика $\mathbf{g}(\mathbf{x})$ из множества:

$$\mathfrak{G} = \{\mathbf{g} | \mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{Y}^*\}, \quad (3.5)$$

например для задачи классификации множество \mathfrak{G} может быть параметрическим семейством функций линейных моделей:

$$\mathfrak{G}_{\text{lin,cl}} = \{\mathbf{g}(\mathbf{W}, \mathbf{x}) | \mathbf{g}(\mathbf{W}, \mathbf{x}) = \text{softmax}(\mathbf{W}\mathbf{x}), \quad \mathbf{W} \in \mathbb{R}^{n \times K}\}. \quad (3.6)$$

Рассмотрим описание метода предложенного в работах [150, 163]. В рамках данных работ предполагается, что для всех данных доступна привилегированная информация $\mathcal{I} = \{1, 2, \dots, m\}$. В работе [150] решается задача классификации вида:

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m, \quad \mathbf{x}_i \in \mathbb{R}^n, \quad y_i \in \mathbb{Y} = \{1, \dots, K\}, \quad (3.7)$$

где y_i — это класс объекта, также будем обозначать \mathbf{y}_i вектором вероятности для класса y_i .

В постановке Хинтона рассматривается параметрическое семейство функций:

$$\mathfrak{G}_{\text{cl}} = \{\mathbf{g} | \mathbf{g} = \text{softmax}(\mathbf{z}(\mathbf{x})/T), \quad \mathbf{z} : \mathbb{R}^n \rightarrow \mathbb{R}^K\}, \quad (3.8)$$

где \mathbf{z} — это дифференцируемая параметрическая функция заданной структуры, T — параметр температуры. В качестве модели учителя \mathbf{f} рассматривается функция из множества \mathfrak{F}_{cl} :

$$\mathfrak{F}_{\text{cl}} = \{\mathbf{f} | \mathbf{f} = \text{softmax}(\mathbf{v}(\mathbf{x})/T), \quad \mathbf{v} : \mathbb{R}^n \rightarrow \mathbb{R}^K\}, \quad (3.9)$$

где \mathbf{v} — это дифференцируемая параметрическая функция заданной структуры, T — параметр температуры. Параметр температуры T имеет следующие свойства:

1. при $T \rightarrow 0$ получаем вектор, в котором один из классов имеет единичную вероятность;
2. при $T \rightarrow \infty$ получаем равновероятные классы.

Функция потерь \mathcal{L} в которой учитывается перенос информации от модели учителя \mathbf{f} к модели ученика \mathbf{g} имеет следующий вид:

$$\begin{aligned} \mathcal{L}_{st}(\mathbf{g}) = & - \underbrace{\sum_{i=1}^m \sum_{k=1}^K y_i^k \log \mathbf{g}(\mathbf{x}_i) \big|_{T=1}}_{\text{исходная функция потерь}} \\ & - \underbrace{\sum_{i=1}^m \sum_{k=1}^K \mathbf{f}(\mathbf{x}_i) \big|_{T=T_0} \log \mathbf{g}(\mathbf{x}_i) \big|_{T=T_0}}_{\text{слагаемое дистилляция}}, \end{aligned} \quad (3.10)$$

где $\cdot \big|_{T=t}$ обозначает, что параметр температуры T в предыдущей функции равняется t .

Получаем оптимизационную задачу:

$$\hat{\mathbf{g}} = \arg \min_{\mathbf{g} \in \mathfrak{G}_{\text{cl}}} \mathcal{L}_{st}(\mathbf{g}). \quad (3.11)$$

Работа [163] обобщает метод предложенный в работе [150]. Решение задачи оптимизации (3.11) зависит только от вектора ответов модели учителя \mathbf{f} . Следовательно признаковые пространства учителя и ученика могут различаться. В этом случае получаем следующую постановку задачи:

$$\mathcal{D} = \{(\mathbf{x}_i, \mathbf{x}_i^*, y_i)\}_{i=1}^m, \quad \mathbf{x}_i \in \mathbb{R}^n, \quad \mathbf{x}_i^* \in \mathbb{R}^{n^*}, \quad y_i \in \{1, \dots, K\}, \quad (3.12)$$

где \mathbf{x}_i это информация доступна на этапах обучения и контроля, а \mathbf{x}_i^* это информация доступна только на этапе обучения. Модель учителя принадлежит множеству моделей $\mathfrak{F}_{\text{cl}}^*$:

$$\mathfrak{F}_{\text{cl}}^* = \{\mathbf{f} | \mathbf{f} = \text{softmax}(\mathbf{v}^*(\mathbf{x}^*)/T), \quad \mathbf{v}^* : \mathbb{R}^{n^*} \rightarrow \mathbb{R}^K\}, \quad (3.13)$$

где \mathbf{v}^* — это дифференцируемая параметрическая функция заданной структуры, T — параметр температуры. Множество моделей \mathfrak{F}_{cl}^* отличается от множества моделей \mathfrak{F}_{cl} из выражения (3.9). В множестве \mathfrak{F}_{cl} модели используют пространство исходных признаков, а в множестве \mathfrak{F}_{cl}^* модели используют пространство привилегированных признаков. Функция потерь (3.10) в случае модели учителя $\mathbf{f} \in \mathfrak{F}_{cl}^*$ переписывается в следующем виде:

$$\mathcal{L}_{st}(\mathbf{g}) = - \sum_{i=1}^m \sum_{k=1}^K y_i^k \log \mathbf{g}(\mathbf{x}_i) \Big|_{T=1} - \sum_{i=1}^m \sum_{k=1}^K \mathbf{f}(\mathbf{x}_i^*) \Big|_{T=T_0} \log \mathbf{g}(\mathbf{x}_i) \Big|_{T=T_0}, \quad (3.14)$$

где $\cdot \Big|_{T=t}$ обозначает, что параметр температуры T в предыдущей функции равняется t .

Требуется построить модель, которая использует привилегированную информацию \mathbf{x}_i^* при обучении. Для этого рассмотрим двухэтапную модель обучения предложенную в работе [163]:

1. выбираем оптимальную модель учителя $\mathbf{f} \in \mathfrak{F}_{cl}^*$;
2. выбираем оптимальную модель ученика $\mathbf{g} \in \mathfrak{G}_{cl}$ используя дистилляцию [150].

Модель ученика — это функция, которая минимизирует (3.14). Модель учителя — это функция, которая минимизирует кросс-энтропийную функцию ошибки:

$$\mathcal{L}_{th}(\mathbf{f}) = - \sum_{i=1}^m \sum_{k=1}^K y_i^k \log \mathbf{f}(\mathbf{x}_i^*). \quad (3.15)$$

3.2 Обобщенная вероятностная постановка задачи дистилляции

Задано распределения целевой переменной $p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g})$. Для поиска $\hat{\mathbf{g}}$ воспользуемся методом максимального правдоподобия. В качестве $\hat{\mathbf{g}}$ выбирается функция, которая максимизирует правдоподобие модели:

$$\hat{\mathbf{g}} = \arg \max_{\mathbf{g} \in \mathfrak{G}} \prod_{i=1}^N p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}), \quad (3.16)$$

где множество \mathfrak{G} задается (3.5). Рассмотрим вероятностную постановку, в которой должны быть выполнены ограничения:

1. задано распределение целевой переменной $p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g})$;
2. задано совместное распределение целевой переменной и ответов модели учителя $p(\mathbf{y}_i, \mathbf{s}_i | \mathbf{x}_i, \mathbf{g})$;
3. для всех $\omega \in \Omega^*$ элементы $\mathbf{y}(\omega)$ и $\mathbf{s}(\omega)$ являются зависимыми величинами, так как ответы учителя должны коррелировать с истинными ответами;

4. если $|\Omega^*| = 0$ то решение должно соответствовать решению (9.8).

Рассмотрим совместное правдоподобие истинных меток и меток учителя:

$$p(\mathbf{Y}, \mathbf{S} | \mathbf{X}, \mathbf{g}, \mathcal{I}) = \prod_{i \notin \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{y}_i, \mathbf{s}_i | \mathbf{x}_i, \mathbf{g}). \quad (3.17)$$

Распишем $p(\mathbf{y}_i, \mathbf{s}_i | \mathbf{x}_i, \mathbf{g})$ по формуле условной вероятности:

$$p(\mathbf{y}_i, \mathbf{s}_i | \mathbf{x}_i, \mathbf{g}) = p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) p(\mathbf{s}_i | \mathbf{y}_i, \mathbf{x}_i, \mathbf{g}) \quad (3.18)$$

Подставляя выражения (3.18) в (3.17) получаем.

$$p(\mathbf{Y}, \mathbf{S} | \mathbf{X}, \mathbf{g}, \mathcal{I}) = \prod_{i \notin \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{s}_i | \mathbf{y}_i, \mathbf{x}_i, \mathbf{g}). \quad (3.19)$$

Заметим, что \mathbf{y}_i и \mathbf{s}_i зависимы только через переменную \mathbf{x}_i , тогда $p(\mathbf{s}_i | \mathbf{y}_i, \mathbf{x}_i, \mathbf{g}) = p(\mathbf{s}_i | \mathbf{x}_i, \mathbf{g})$. Получаем совместное правдоподобие:

$$p(\mathbf{Y}, \mathbf{S} | \mathbf{X}, \mathbf{g}, \mathcal{I}) = \prod_{i \notin \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{s}_i | \mathbf{x}_i, \mathbf{g}). \quad (3.20)$$

Используя (3.20) получаем следующую оптимизационную задачу для поиска $\hat{\mathbf{g}}$

$$\hat{\mathbf{g}} = \arg \max_{\mathbf{g} \in \mathcal{G}} \prod_{i \notin \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) \prod_{i \in \mathcal{I}} p(\mathbf{s}_i | \mathbf{x}_i, \mathbf{g}). \quad (3.21)$$

Для удобства, будем минимизировать логарифм, тогда из (3.21) получаем:

$$\hat{\mathbf{g}} = \arg \max_{\mathbf{g} \in \mathcal{G}} \sum_{i \notin \mathcal{I}} \log p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) + (1 - \lambda) \sum_{i \in \mathcal{I}} \log p(\mathbf{y}_i | \mathbf{x}_i, \mathbf{g}) + \lambda \sum_{i \in \mathcal{I}} \log p(\mathbf{s}_i | \mathbf{x}_i, \mathbf{g}), \quad (3.22)$$

где параметр $\lambda \in [0, 1]$ введен для взвешивания ошибок на истинных ответах и ошибок относительно ответов учителя.

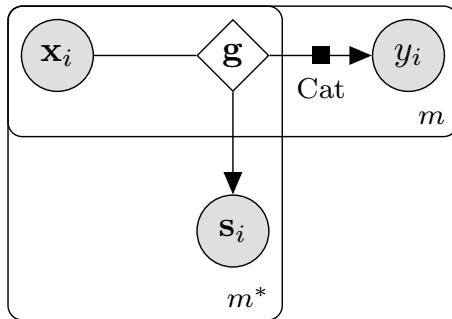


Рис. 14: Вероятностная модель в формате плоских нотаций.

На рис. 14 показан вид вероятностной модели в графовой нотации, для произвольной функции \mathbf{g} . Для каждой реализации \mathbf{g} соответствующий блок требует уточнение. На рис 16 показана более подробная реализация в случае, когда модель \mathbf{g} это линейная модель.

Для задачи многоклассовой классификации рассматриваются следующие вероятностные предположения:

1. рассматривается функция учителя $\mathbf{f} \in \mathfrak{F}_{\text{cl}}^*$ (3.13);
2. рассматривается функция ученика следующего вида $\mathbf{g} \in \mathfrak{G}_{\text{cl}}$ (3.8);
3. для истинных меток рассматривается категориальное распределение $p(y|\mathbf{x}, \mathbf{g}) = \text{Cat}(\mathbf{g}(\mathbf{x}))$, где $\mathbf{g}(\mathbf{x})$ задает вероятность каждого класса;
4. для меток учителя введем плотность распределения

$$p(\mathbf{s}|\mathbf{x}, \mathbf{g}) = C \prod_{k=1}^K g_k(\mathbf{x})^{s^k}, \quad (3.23)$$

где g^k обозначает вероятность класса k , которую предсказывает модель ученика, а s^k — вероятность класса k , которую предсказывает модель учителя.

Теорема 1. *Пусть вероятность каждого класса отделима от нуля и единицы, то есть для всех k выполняется $1 > 1 - \varepsilon > g_k(\mathbf{x}) > \varepsilon > 0$, тогда при*

$$C = (-1)^K \frac{K^{K/2}}{2^{K(K-1)/2}} \prod_{k=1}^K g_k(\mathbf{x}) \log g_k(\mathbf{x}) \quad (3.24)$$

функция $p(\mathbf{s}|\mathbf{x}, \mathbf{g})$ определенная в (3.23) является плотностью распределения.

Доказательство. Во-первых покажем, что для произвольного вектора ответов $\mathbf{s} \in \mathcal{S}_K$ выполняется $p(\mathbf{s}|\mathbf{x}, \mathbf{g}) \geq 0$. Заметим, что для всех k выполняется, что $\log g_k(\mathbf{x}) < 0$, тогда

$$C = \underbrace{\frac{K^{K/2}}{2^{K(K-1)/2}}}_{>0} \prod_{k=1}^K \underbrace{g_k(\mathbf{x})}_{>\varepsilon} \underbrace{(-\log g_k(\mathbf{x}))}_{>0} > 0, \quad (3.25)$$

тогда с учетом того, что $g_k(\mathbf{x}) > 0$ и $C > 0$ получаем, что $p(\mathbf{s}|\mathbf{x}, \mathbf{g}) \geq 0$. Во-вторых покажем, что интеграл по всему пространству ответов \mathcal{S}_K является

конечным:

$$\begin{aligned}
\int_{S_K} p(\mathbf{s}|\mathbf{x}, \mathbf{g}) ds &= \int_{S_K} \prod_{k=1}^K g_k(\mathbf{x})^{s^k} ds = \prod_{k=1}^K \int_{S_K} g_k(\mathbf{x})^{s^k} ds \\
&= \prod_{k=1}^K \int_0^1 \frac{r^{K-1} \sqrt{K}}{(K-1)! \sqrt{2^{K-1}}} g_k(\mathbf{x})^r dr = \prod_{k=1}^K \underbrace{\frac{\sqrt{K}}{(K-1)! \sqrt{2^{K-1}}}}_D \int_0^1 r^{K-1} g_k(\mathbf{x})^r dr \\
&= D^K \prod_{k=1}^K \int_0^1 r^{K-1} \exp(r \log g_k(\mathbf{x})) dr \\
&= (-D)^K \prod_{k=1}^K \log g_k(\mathbf{x}) (\Gamma(K) - \Gamma(K, -\log g_k(\mathbf{x}))) \\
&= (-D)^K (K-1)!^K \prod_{k=1}^K \log g_k(\mathbf{x}) (1 - g_k(\mathbf{x}) \exp_{K-1}(-\log g_k(\mathbf{x})) + g_k(\mathbf{x})) \\
&= \frac{(-\sqrt{K})^K}{2^{K(K-1)/2}} \prod_{k=1}^K \log g_k(\mathbf{x}) (1 - g_k(\mathbf{x}) \exp_{K-1}(-\log g_k(\mathbf{x})) + g_k(\mathbf{x})) < \infty,
\end{aligned} \tag{3.26}$$

где $\Gamma(K)$ является гамма функцией, $\Gamma(K, -\log g_k(\mathbf{x}))$ является неполной гамма функцией, $\exp_n(x)$ является суммой Тейлора из первых n слагаемых. В рамках приближенных расчетов будем считать, что $\exp_n(x) \approx \exp(x)$, тогда с учетом (3.26) получаем:

$$C(\mathbf{g}, \mathbf{x}) = \int_{S_K} p(\mathbf{s}|\mathbf{x}, \mathbf{g}) ds \approx (-1)^K \frac{K^{K/2}}{2^{K(K-1)/2}} \prod_{k=1}^K g_k(\mathbf{x}) \log g_k(\mathbf{x}) \tag{3.27}$$

Полученное выражение (3.27) заканчивает доказательство теоремы. \square

Из теоремы 1 следует, что плотность введенная для меток учителя является плотностью распределения, следовательно можно воспользоваться выражением (3.22). Используя предположения 1)–4) и подставляя в (3.22) получаем следующую оптимизационную задачу:

$$\begin{aligned}
\hat{\mathbf{g}} &= \arg \max_{\mathbf{g} \in \mathcal{G}} \sum_{i \notin \mathcal{I}} \sum_{k=1}^K y_i^k \log g_k(\mathbf{x}_i) \Big|_{T=1} \\
&\quad + (1-\lambda) \sum_{i \in \mathcal{I}} \sum_{k=1}^K y_i^k \log g_k(\mathbf{x}_i) \Big|_{T=1} + \lambda \sum_{i \in \mathcal{I}} \sum_{k=1}^K s_{i,k} \log g_k(\mathbf{x}_i) \Big|_{T=T_0} \\
&\quad + \lambda \sum_{i \in \mathcal{I}} \sum_{k=1}^K \left(\log g_k(\mathbf{x}_i) \Big|_{T=T_0} + \log \log \frac{1}{g_k(\mathbf{x}_i)} \Big|_{T=T_0} \right),
\end{aligned} \tag{3.28}$$

где выражение $\cdot \Big|_{T=t}$ обозначает, что в предыдущую функцию softmax требуется подставить значение температуры T равное некоторому значению t .

Проанализировав выражение (3.28) получаем, что первые три слагаемые совпадают со слагаемыми в выражении (3.10) при $\mathcal{I} = \{1, \dots, m\}$, и $\lambda = \frac{1}{2}$,

а третье слагаемое является некоторым регуляризатором, который получен из вида распределения.

Анализируя первые три слагаемых в выражении (3.28) получаем, что при $T_0 = 1$ получаем сумму кросс энтропий между двумя распределениями для каждого объекта:

1. первое распределение это выпуклая комбинация с весом $1 - \lambda$ и λ : распределения задаваемое метками объектов $\text{Cat}(\mathbf{y})$ и распределения задаваемого моделью учителя $\text{Cat}(\mathbf{s})$
2. второе распределение это распределение задаваемое плотностью распределения ответов ученика $\text{Cat}(\mathbf{g}(\mathbf{x}))$.

Получаем, что модель ученика восстанавливает плотность не исходных меток, а новую плотность, которая является выпуклой комбинацией плотности исходных меток и меток учителя. Для задачи регрессии рассматриваются следующие вероятностные предположения:

1. рассматривается функция учителя $\mathbf{f} \in \mathfrak{F}_{\text{rg}}^*$:

$$\mathfrak{F}_{\text{rg}}^* = \{\mathbf{f} | \mathbf{f} = \mathbf{v}^*(\mathbf{x}^*), \quad \mathbf{v}^* : \mathbb{R}^{n^*} \rightarrow \mathbb{R}\}, \quad (3.29)$$

где \mathbf{v}^* — это дифференцируемая параметрическая функция;

2. рассматривается функция ученика $\mathbf{g} \in \mathfrak{G}_{\text{rg}}$:

$$\mathfrak{G}_{\text{rg}} = \{\mathbf{g} | \mathbf{g} = \mathbf{z}(\mathbf{x}), \quad \mathbf{z} : \mathbb{R}^n \rightarrow \mathbb{R}^K\}, \quad (3.30)$$

где \mathbf{z} — это дифференцируемая параметрическая функция;

3. истинные метки имеют нормальное распределение

$$p(y|\mathbf{x}, \mathbf{g}) = \mathcal{N}(y|\mathbf{g}(\mathbf{x}), \sigma); \quad (3.31)$$

4. метки учителя распределены

$$p(s|\mathbf{x}, \mathbf{g}) = \mathcal{N}(s|\mathbf{g}(\mathbf{x}), \sigma_s); \quad (3.32)$$

Используя предположения 1)–4) и подставляя в (3.22) получаем следующую оптимизационную задачу:

$$\begin{aligned} \hat{g} = \arg \min_{g \in \mathcal{G}} & \sum_{i \notin \mathcal{I}} \sigma^2 (y_i - \mathbf{g}(\mathbf{x}_i))^2 \\ & + (1 - \lambda) \sum_{i \in \mathcal{I}} \sigma^2 (y_i - \mathbf{g}(\mathbf{x}_i))^2 + \lambda \sum_{i \in \mathcal{I}} \sigma_s^2 (s_i - \mathbf{g}(\mathbf{x}_i))^2. \end{aligned} \quad (3.33)$$

Выражение (3.33) записано с точностью до аддитивной константы относительно \mathbf{g} .

Теорема 2. Пусть множество \mathcal{G} описывает класс линейных функций вида $\mathbf{g}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$. Тогда решение оптимизационной задачи (3.33) эквивалентно решению следующей задачи линейной регрессии:

$$\mathbf{y}'' = \mathbf{X}\mathbf{w} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \quad (3.34)$$

где $\boldsymbol{\Sigma}^{-1} = \text{diag}(\boldsymbol{\sigma}')$ и \mathbf{y}'' имеют следующий вид:

$$\begin{aligned} \sigma'_i &= \begin{cases} \sigma^2, & \text{если } i \notin \mathcal{I} \\ (1 - \lambda)\sigma^2 + \lambda\sigma_s^2, & \text{иначе} \end{cases}, \\ \mathbf{y}'' &= \boldsymbol{\Sigma}\mathbf{y}', \\ y'_i &= \begin{cases} \sigma^2 y_i, & \text{если } i \notin \mathcal{I} \\ (1 - \lambda)\sigma^2 y_i + \lambda\sigma_s^2 s_i, & \text{иначе} \end{cases}. \end{aligned} \quad (3.35)$$

Доказательство. Обозначим $\mathbf{a}_{\mathcal{J}} = [a_i | i \in \mathcal{J}]^\top$, где \mathbf{a} произвольный вектор, а \mathcal{J} произвольное не пустое индексное множество. Подвектор вектора ответов \mathbf{y} , для элементов которого доступна привилегированная информация обозначим $\mathbf{y}_{\bar{\mathcal{I}}} = [y_i | i \in \bar{\mathcal{I}}]^\top$. Аналогично обозначим матрицу $\mathbf{X}_{\bar{\mathcal{I}}} = [\mathbf{x}_i | i \in \bar{\mathcal{I}}]^\top$.

В случае линейной модели $\mathbf{g}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ выражение (3.33) принимает вид:

$$\begin{aligned} \hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathcal{W}} & \sigma^2 (\mathbf{y}_{\bar{\mathcal{I}}} - \mathbf{X}_{\bar{\mathcal{I}}}\mathbf{w})^\top (\mathbf{y}_{\bar{\mathcal{I}}} - \mathbf{X}_{\bar{\mathcal{I}}}\mathbf{w}) \\ & + \sigma^2 (1 - \lambda) (\mathbf{y}_{\mathcal{I}} - \mathbf{X}_{\mathcal{I}}\mathbf{w})^\top (\mathbf{y}_{\mathcal{I}} - \mathbf{X}_{\mathcal{I}}\mathbf{w}) + \sigma_s^2 \lambda (\mathbf{s}_{\mathcal{I}} - \mathbf{X}_{\mathcal{I}}\mathbf{w})^\top (\mathbf{s}_{\mathcal{I}} - \mathbf{X}_{\mathcal{I}}\mathbf{w}). \end{aligned} \quad (3.36)$$

Раскроем скобки и сгруппируем:

$$\begin{aligned} \hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathcal{W}} & \sigma^2 (\mathbf{w}^\top \mathbf{X}_{\bar{\mathcal{I}}}^\top \mathbf{X}_{\bar{\mathcal{I}}} \mathbf{w} - 2\mathbf{y}_{\bar{\mathcal{I}}}^\top \mathbf{X}_{\bar{\mathcal{I}}} \mathbf{w}) \\ & + (1 - \lambda) \sigma^2 (\mathbf{w}^\top \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} \mathbf{w} - 2\mathbf{y}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} \mathbf{w}) + \lambda \sigma_s^2 (\mathbf{w}^\top \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} \mathbf{w} - 2\mathbf{s}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} \mathbf{w}) \end{aligned} \quad (3.37)$$

Продифференцируем выражение, приравняем к нулю и сгруппируем элементы:

$$\begin{aligned} (\sigma^2 \mathbf{X}_{\bar{\mathcal{I}}}^\top \mathbf{X}_{\bar{\mathcal{I}}} + (1 - \lambda) \sigma^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} + \lambda \sigma_s^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}}) \mathbf{w} = & 2\sigma^2 \mathbf{X}_{\bar{\mathcal{I}}}^\top \mathbf{y}_{\bar{\mathcal{I}}} \\ & + 2(1 - \lambda) \sigma^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{y}_{\mathcal{I}} + 2\lambda \sigma_s^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{s}_{\mathcal{I}}. \end{aligned} \quad (3.38)$$

Воспользуемся следующими равенствами:

$$\begin{aligned} \sigma^2 \mathbf{X}_{\bar{\mathcal{I}}}^\top \mathbf{X}_{\bar{\mathcal{I}}} + (1 - \lambda) \sigma^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} + \lambda \sigma_s^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{X}_{\mathcal{I}} &= \mathbf{X}^\top \boldsymbol{\Sigma}^{-1} \mathbf{X}, \\ 2\sigma^2 \mathbf{X}_{\bar{\mathcal{I}}}^\top \mathbf{y}_{\bar{\mathcal{I}}} + 2(1 - \lambda) \sigma^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{y}_{\mathcal{I}} + 2\lambda \sigma_s^2 \mathbf{X}_{\mathcal{I}}^\top \mathbf{s}_{\mathcal{I}} &= 2\mathbf{X}\mathbf{y}', \end{aligned} \quad (3.39)$$

где $\boldsymbol{\Sigma}$ и \mathbf{y}' из условия задачи (3.35).

Подставляя (3.39) в (3.38) получаем:

$$\mathbf{w} = 2(\mathbf{X}^\top \boldsymbol{\Sigma}^{-1} \mathbf{X})^{-1} \mathbf{X} \boldsymbol{\Sigma}^{-1} \mathbf{y}'', \quad (3.40)$$

что соответствует решению задачи (3.34). \square

Теорема 2 показывает, что обучения с учителем для задачи регрессии можно свести к классической задачи оптимизации для задачи линейной регрессии.

3.3 Анализ вероятностного подхода к дистилляции моделей линейных моделей

Проводится вычислительный эксперимент для анализа качества моделей, которые получены путем дистилляции модели учителя в модель ученика. Как показано в теореме 2 задачу регрессии с учителем можно свести к задачи регрессии без учителя, поэтому в эксперименте более подробно рассматривается случай классификации. Во всех частях вычислительного эксперимента для поиска оптимальных параметров нейросетей использовался градиентный метод оптимизации Адам [159].

В данной части проводится эксперимент для задачи классификации для выборки FashionMNIST [155]. В качестве модели учителя \mathbf{f} рассматривается модель нейросети с двумя сверточными слоями и с тремя полно связанными слоями, в качестве функции активации рассматривается ReLu. Модель учителя содержит 30 тысяч обучаемых параметров. В качестве модели ученика рассматривается модель логистической регрессии для многоклассовой классификации. Модель ученика содержит 7850 обучаемых параметров.

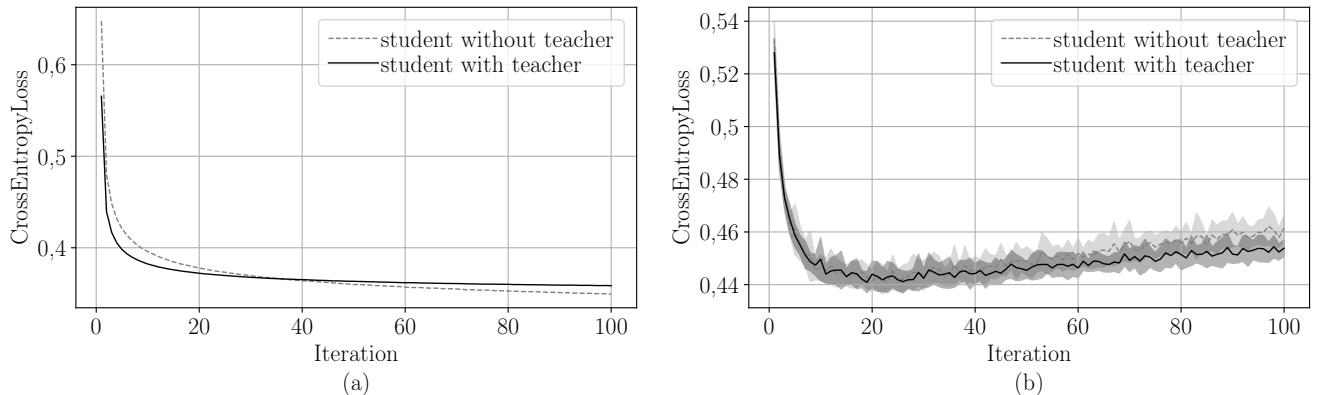


Рис. 15: Зависимость кросс-энтропии между истинными метками и предсказанными учеников вероятностями классов: а) на обучающей выборке; б) на тестовой выборке.

На рис. 15 показан график зависимости кросс-энтропии между истинными метками объектов и вероятностями, которые предсказывает модель ученика. На графике сравнивается модель, которая обучалась без учителя (в задаче оптимизации (3.28) присутствует только первое слагаемое) с моделью, которая была получена путем дистилляции модели нейросети в линейную модель. На графике видно, что обе модели начинают переобучаться после 30-й итерации, но модель, которая получена путем дистилляции переобучивается не так быстро, что следует из того, что ошибка на тестовой выборке растет медленней, а на обучающей выборке падает также медленней.

Проанализируем модель на синтетической выборке. Выборка построенная

следующим образом:

$$\begin{aligned} \mathbf{W} &= [\mathcal{N}(w_{jk}|0, 1)]_{n \times K}, & \mathbf{X} &= [\mathcal{N}(x_{ij}|0, 1)]_{m \times n}, \\ \mathbf{S} &= \text{softmax}(\mathbf{X}\mathbf{W}), & \mathbf{y} &= [\text{Cat}(y_i|\mathbf{s}_i)], \end{aligned} \quad (3.41)$$

где функция softmax берется построчно. Строки матрицы \mathbf{S} будем рассматривать как предсказание учителя, то есть учитель знает истинные вероятности каждого класса. На рис. 16 показана вероятностная модель в графовой нотации. В эксперименте число признаков $n = 10$, число классов $K = 3$, для обучения было сгенерировано $m_{\text{train}} = 1000$ и $m_{\text{test}} = 100$ объектов.

На рис. 17 показано распределение по классам для каждого объекта обучающей выборки. Видно, что все классы являются равновероятными.

Построим в качестве ученика простую линейную модель, которая минимизирует крос-энтропийную (первое слагаемое в формуле (3.28)). Представление данной модели в виде графовой модели показано на рис. 16.

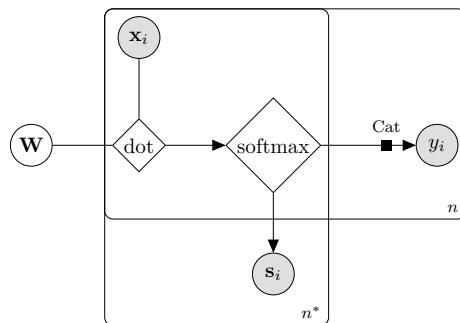


Рис. 16: Вероятностная модель используемая в синтетическом эксперименте.

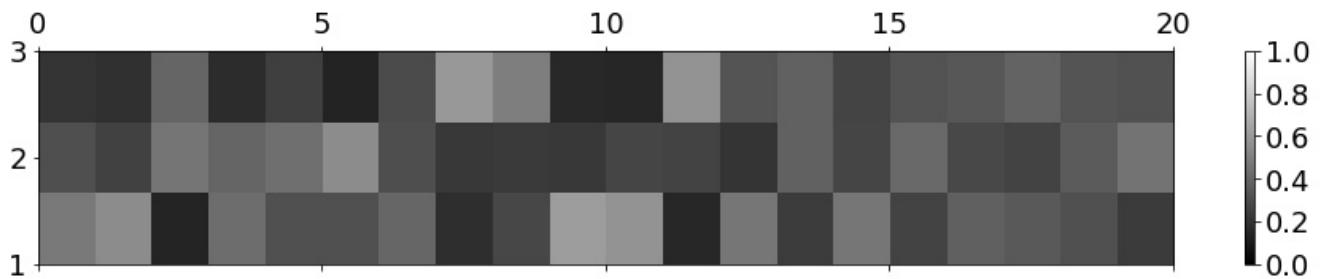


Рис. 17: Истинное распределение объектов по классам.

На рис. 18 показано распределение вероятностей классов, которое предсказала модель. Видно, что данное распределение является не соответствует истинному, так как модель сосредотачивает всю вероятность в одном классе.

Рассмотрим модель, которая учитывает информацию о истинных распределениях на классах для каждого объекта. Для этого будем минимизировать первые три слагаемых в формуле (3.28), при $T_0 = 1$ и $\lambda = 0,75$. В качестве методов учителя $s_{i,k}$ использовались истинные вероятности для каждого класса для данного объекта. На рис 19 показано распределение, которое дала модель в данном случае, видно, что распределения являются сглаженными и концентрации всей вероятности в одном классе не наблюдается.

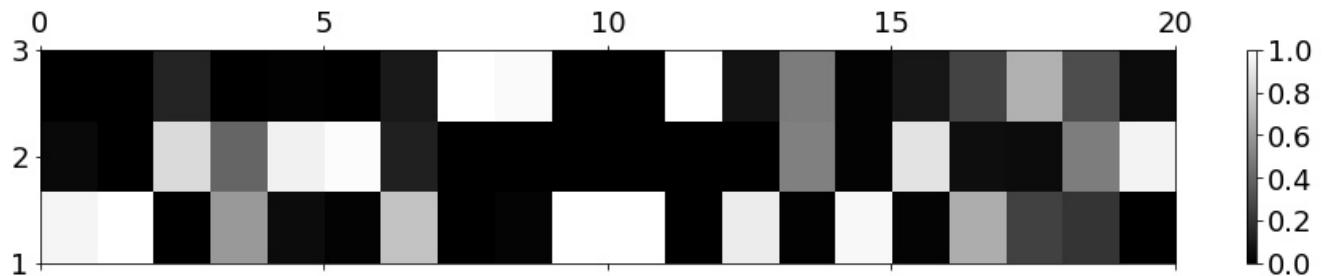


Рис. 18: Распределение предсказанное моделью без использования информации об истинном распределении на классах.

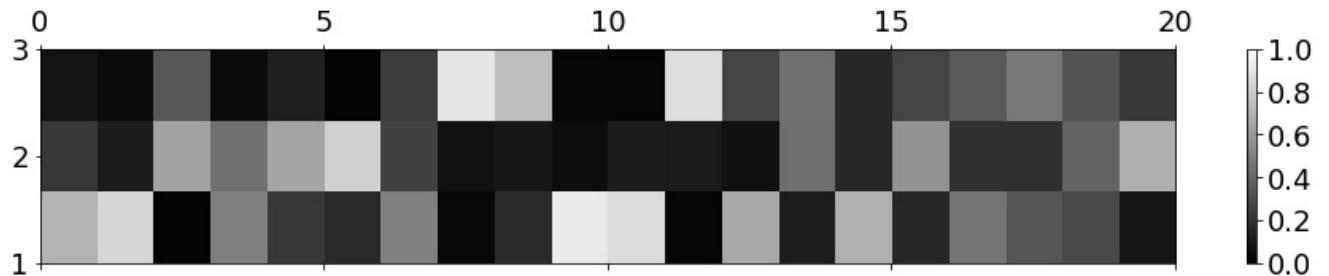


Рис. 19: Распределение предсказанное моделью с использованием информации об истинном распределении на классах.

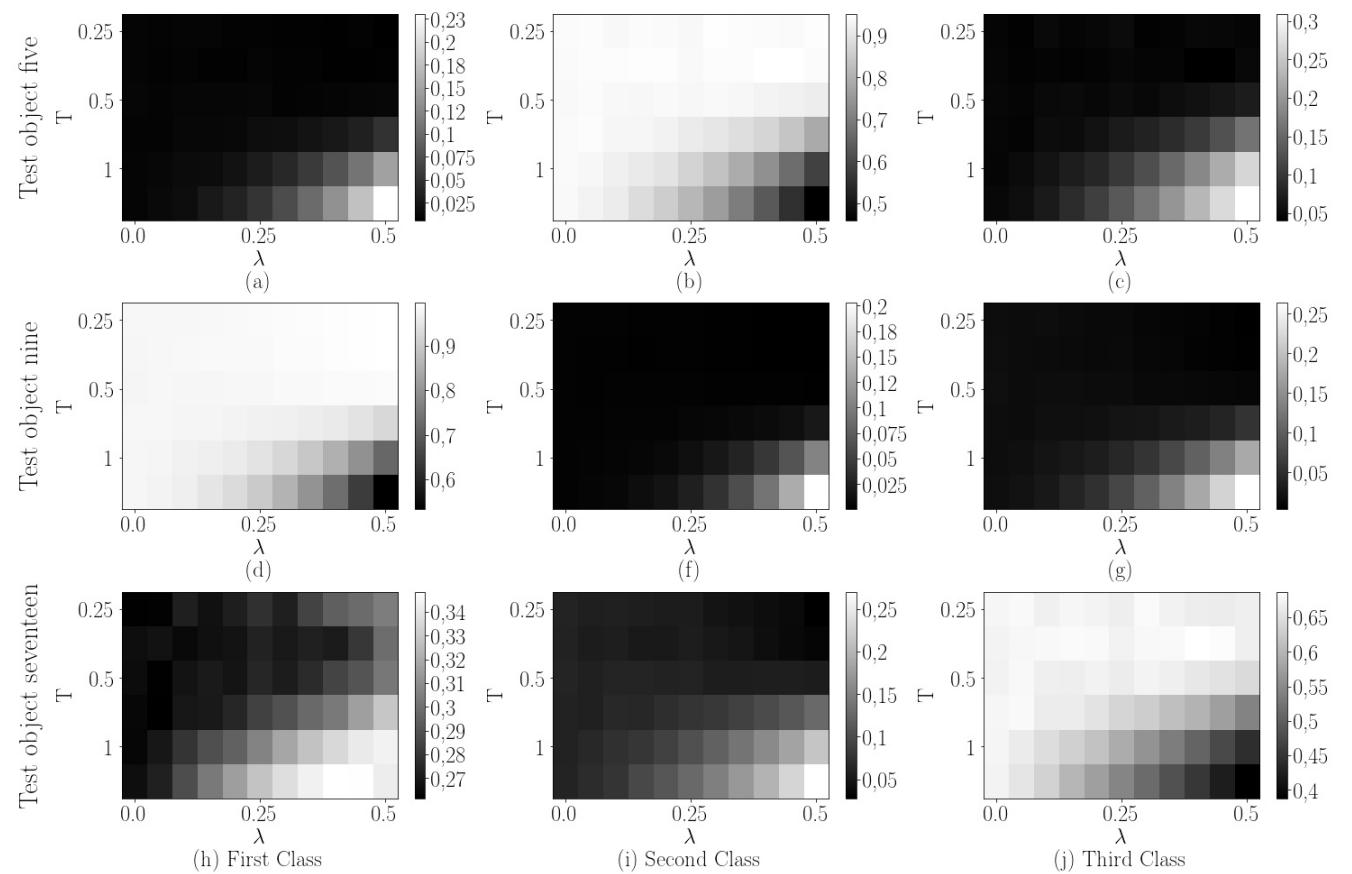


Рис. 20: Вероятности классов для разных объектов.

Таблица 2: Сводная таблица результатов вычислительного эксперимента.

Dataset	Model	CrossEntropyLoss	Accuracy	StudentSize
FashionMnist	without teacher	$0,461 \pm 0,005$	$0,841 \pm 0,002$	7850
	with teacher	$0,453 \pm 0,003$	$0,842 \pm 0,002$	7850
Synthetic	without teacher	$0,225 \pm 0,002$	$0,831 \pm 0,002$	33
	with teacher	$0,452 \pm 0,001$	$0,828 \pm 0,001$	33
Twitter	without teacher	$0,501 \pm 0,006$	$0,747 \pm 0,005$	1538
	with teacher	$0,489 \pm 0,003$	$0,764 \pm 0,004$	1538

Заметим, что в данном примере предполагается, что модель учителя учитывает не только метки классов, а и распределение на метках классов, в то время как в выборке $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$, имеются только точечные оценки в виде меток.

В данном примере используются истинные распределения в качестве предсказаний учителя, но их можно заменить предсказаниями модели учителя, которая предсказывает не только сами метки, а и их распределение для каждого объекта.

На рис. 20 показана зависимость вероятности верного класса от температуры T и параметра доверия λ для одного из объекта из тестовой выборке. Видно, что при увеличении температуры распределение на классах становится более равномерным.

В данной части проводится эксперимент на выборке Twitter Sentiment Analysis. Данная выборка содержит короткие сообщения, для которых нужно предсказать эмоциональный окрас: содержит твит позитивный окрас или негативный. Выборка разделена на 1,18 миллиона твитов для обучения и 0,35 миллиона твитов для тестирования. В твитах была выполнена следующая предобработка:

- все твиты были переведены в нижний регистр;
- все никнеймы вида “@andrey” были заменены на токен “name”;
- все цифры были заменены на токен “number”.

Результаты данной части эксперимента показаны в табл. 2. В качестве модели учителя использовалась модель Bi-LSTM с 170 тысячами параметров для обучения. В качестве эмбедингов обучалась матрица из 30 миллионов параметров в единой процедуре с моделью BI-LSTM. Обученная модель предсказывает с точностью 0,835. В качестве модели ученика рассматривается модель с 1538 параметрами, но в качестве эмбедингов рассматривается переобученная модель BERT.

4 Байесовская дистилляция моделей глубокого обучения

Исследуется проблема снижения числа обучаемых параметров моделей машинного обучения. Примерами таких моделей, с избыточным числом параметров, являются AlexNet [141], VGGNet [142], ResNet [148], BERT [144, 143], mT5 [146], GPT3[145] и другие. Табл. 3 описывает глубокие модели машинного обучения.

Таблица 3: Число параметров в моделях машинного обучения.

Название	AlexNet	VGGNet	ResNet	BERT	mT5	GPT3
Год	2012	2014	2015	2018	2020	2020
Тип данных	изображение	изображение	изображение	текст	текст	текст
Число параметров, млрд	0,06	0,13	0,06	0,34	13	175

Число параметров моделей машинного обучения с годами растет. Это влечет снижение интерпретируемости моделей. Данная проблема рассматривается в специальном классе задач по состязательным атакам (англ. adversarial attack) [140]. Большое число параметров требует больших вычислительных ресурсов. Из-за этого данные модели не могут быть использованы в мобильных устройствах. Для снижения числа параметров предложен метод дистилляции модели [150, 162, 163]. Дистиллируемая модель с большим числом параметров называется *учителем*, а модель получаемая путем дистилляции называется *учеником*. При оптимизации параметров модели ученика используется модель учителя с фиксированными параметрами.

Определение 4.1. *Дистилляция модели – снижение сложности модели путем выбора модели в множестве более простых моделей на основе параметров и ответов более сложной фиксированной модели.*

Идея дистилляции предложена в работах Дж. Е. Хинтона и В. Н. Вапником [150, 162, 163]. В этих работах предлагается использовать ответы учителя в качестве целевой переменной для обучения модели ученика. Поставлен ряд экспериментов, в которых проводилась дистилляция моделей для задачи классификации машинного обучения. Базовый эксперимент на выборке MNIST [151] показал применимость метода для дистилляции избыточно сложной нейросетевой модели в нейросетевую модель меньшей сложности. Эксперимент по дистилляции ансамбля моделей в одну модель для решения задачи распознания речи. Также в работе [150] был проведен эксперимент по обучению экспертных моделей на основе одной модели с большим числом параметров при помощи предложенного метода дистилляции на ответах учителя.

В работе [139] предложен метод передачи селективности нейронов (англ. neuron selectivity transfer) основанный на минимизации специальной функции

потерь основаной на максимальном среднем отклонении (англ. maximum mean discrepancy) между выходами всех слоев модели учителя и ученика. Вычислительный эксперимент показал эффективность данного метода для задачи классификации изображений на примере выборок CIFAR [137] и ImageNet [138].

В данной работе предлагается методы основанный на байесовском выводе. В качестве априорного распределения параметров модели ученика предлагается использовать апостериорное распределение параметров модели учителя. Решается задача сопоставления пространства параметров модели учителя и модели ученика. Авторы предлагают подход, основанный на последовательном сопоставлении пространств параметров модели ученика и учителя.

Определение 4.2. *Структура модели — множество структурных параметров модели, которые задают вид суперпозиции.*

Определение 4.3. *Сопоставление параметрических моделей — изменение структуры модели (одной или нескольких моделей) в результате которого векторы параметров различных моделей лежат в одном пространстве.*

В результате сопоставления, параметры модели учителя и модели ученика лежат в одном пространстве. Как следствие, в качестве априорного распределения параметров модели ученика выбирается апостериорное распределение параметров модели учителя. В данной работе в качестве параметрических моделей рассматривается полносвязная нейронная сеть. В качестве структурных параметров модели выбраны число слоев, а также размер каждого скрытого слоя.

В рамках предложенного метода сопоставления параметрических моделей не оговорен выбор порядка на множестве параметров модели учителя. Для этого предлагается упорядочивать параметры модели учителя на основе их значимости. Первый нейрон является наиболее значимым, а последний нейрон наименее значимым. Порядок задается на основе отношения плотности распределения упорядочиваемого параметра к плотности распределения параметра в нуле [160] или на основе метода Белсли [161]. В рамках данной работы порядок на параметрах в рамках одного слоя задается случайным образом.

В рамках вычислительного эксперимента проводится теоретический анализ. Предложенный метод дистилляции анализируется на примере синтетической выборки, а также на реальной выборке FashionMnist [155].

4.1 Постановка задачи дистилляции в терминах байесовского подхода

Задана выборка

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m, \quad \mathbf{x}_i \in \mathbb{R}^n, \quad y_i \in \mathbb{Y}, \quad (4.1)$$

где \mathbf{x}_i, y_i — признаковое описание и целевая переменная i -го объекта, число объектов в обучающей выборке обозначается m . Размер признакового описания

объектов обозначается n . Множество $\mathbb{Y} = \{1, \dots, K\}$ для задачи классификации, где K число классов, множество $\mathbb{Y} = \mathbb{R}$ для задачи регрессии.

Задана модель учителя в виде суперпозиций линейных и нелинейных преобразований:

$$f = \sigma \circ \mathbf{U}_T \sigma \circ \mathbf{U}_{T-1} \circ \dots \circ \mathbf{U}_2 \sigma \circ \mathbf{U}_1, \quad (4.2)$$

где T — число слоев модели учителя, σ — функция активации, а \mathbf{U}_t обозначает матрицу линейного преобразования. Матрицы \mathbf{U} соединяются в вектор параметров \mathbf{u} модели учителя f :

$$\mathbf{u} = \text{vec}([\mathbf{U}_T, \mathbf{U}_{T-1}, \dots, \mathbf{U}_1]), \quad (4.3)$$

где vec операция векторизации соединенных матриц. Каждая матрица \mathbf{U}_t имеет размер $n_t \times n_{t-1}$, где $n_0 = n$, а $n_T = 1$ для задачи регрессии и $n_T = K$ для задачи классификации на K классов. Число параметров N_{tr} учителя f

$$N_{\text{tr}} = \sum_{t=1}^T n_t n_{t-1}. \quad (4.4)$$

Для построения вектора параметров \mathbf{u} задается полный порядок на элементов матриц \mathbf{U}_t . Для полносвязной нейронной сети вводится естественный порядок, индуцированный номером слоя t , номером нейрона, и номером элемента вектора параметров нейрона: выбирается матрица \mathbf{U}_t , строка этой матрицы и элемент строки.

Например, для модели учителя в задаче регрессии:

$$f(\mathbf{x}) = \sigma \circ \mathbf{U}_3 \sigma \circ \mathbf{U}_2 \sigma \circ \mathbf{U}_1 \mathbf{x}, \quad (4.5)$$

вектор параметров \mathbf{u} принимает вид

$$\mathbf{u} = [u_1^{1,1}, \dots, u_1^{1,n}, \dots, u_1^{n_1,1}, \dots, u_1^{n_1,n}, u_2^{1,1}, \dots, u_2^{1,n_1}, \dots, u_2^{n_2,1}, \dots, u_2^{n_2,n_1}, u_3^{1,1}, \dots, u_3^{1,n_2}]. \quad (4.6)$$

Пусть для вектора параметров учителя f известно апостериорное распределение параметров $p(\mathbf{u}|\mathfrak{D})$. На основе выборки \mathfrak{D} и апостериорного распределения параметров учителя f требуется выбрать модель ученика из параметрического семейства функций:

$$g = \sigma \circ \mathbf{W}_L \sigma \circ \dots \circ \mathbf{W}_1, \quad \mathbf{W}_l \in \mathbb{R}^{n_l \times n_{l-1}}, \quad (4.7)$$

где L число слоев модели ученика. Число параметров N_{st} модели ученика g вычисляется аналогично выражению (4.4). Вектор параметров модели ученика \mathbf{w} строится аналогичным образом (4.3). Модель g задается своим вектором параметров \mathbf{w} . Следовательно, задача выбора модели g эквивалентна задаче оптимизации вектора параметров $\mathbf{w} \in \mathbb{R}^{N_{\text{st}}}$.

Параметры $\hat{\mathbf{w}} \in \mathbb{R}^{N_{\text{st}}}$ оптимизируются при помощи вариационного вывода на основе совместного правдоподобия модели и данных:

$$\mathcal{L}(\mathfrak{D}, \mathbf{A}) = \log p(\mathfrak{D}|\mathbf{A}) = \log \int_{\mathbf{w} \in \mathbb{R}^{N_{\text{st}}}} p(\mathfrak{D}|\mathbf{w}) p(\mathbf{w}|\mathbf{A}) d\mathbf{w}, \quad (4.8)$$

где $p(\mathbf{w}|\mathbf{A})$ — априорное распределение вектора параметров модели ученика. Так как взятие интеграла (4.8) является вычислительно сложной задачей, используется вариационный подход [160, 161]. Для этого задается вариационное

распределение параметров модели ученика $q(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, которое аппроксимирует неизвестное апостериорное распределение $p(\mathbf{w}|\mathfrak{D})$

$$q(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \approx p(\mathbf{w}|\mathfrak{D}). \quad (4.9)$$

Далее распределение $q(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ обозначается просто $q(\mathbf{w})$. Оптимизация параметров \mathbf{w} сводится к решению задачи:

$$\hat{\mathbf{w}} = \arg \min_{\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{w}} D_{\text{KL}}(q(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) || p(\mathbf{w}|\mathbf{A})) - \log p(\mathbf{y}|\mathbf{X}, \mathbf{w}). \quad (4.10)$$

Выражение (9.8) не учитывает параметры учителя f . Для использования информации о распределении параметров учителя предлагается рассмотреть параметры априорного распределения $p(\mathbf{w}|\mathbf{A})$ как функцию от апостериорного распределения учителя $p(\mathbf{u}|\mathfrak{D})$.

4.2 Построение априорного распределения параметров ученика на основе параметров учителя

Апостериорное распределение параметров модели учителя предполагается нормальным:

$$p(\mathbf{u}|\mathfrak{D}) = \mathcal{N}(\mathbf{m}, \boldsymbol{\Sigma}), \quad (4.11)$$

где \mathbf{m} и $\boldsymbol{\Sigma}$ параметры этого распределения. На основе параметров \mathbf{m} и $\boldsymbol{\Sigma}$ требуется задать параметры \mathbf{A} априорного распределения $p(\mathbf{w}|\mathbf{A})$. В случае, когда структура моделей учителя и ученика задаются числом слоев и размером этих слоев, то возможны следующие варианты: 1) число слоев и размер каждого слоя совпадает; 2) число слоев совпадает, а размеры различаются; 3) не совпадает число слоев.

Учитель и ученик принадлежат одному семейству. Рассмотрим следующие условия:

1. число слоев модели учителя равняется числу слоев модели ученика $L = T$;
2. размеры соответствующих слоев совпадают, другими словами, для всех t, l таких, что $t = l$ выполняется $n_l = n_t$, где n_t обозначает размер t -го слоя учителя, а n_l размер l -го слоя ученика.

В случае выполнения этих условий, априорное распределение параметров модели ученика приравнивается к апостериорному распределению параметров учителя $p(\mathbf{w}|\mathbf{A}) = p(\mathbf{u}|\mathfrak{D})$.

Удаление нейрона в слое учителя. Проведем согласование модели учителя и модели ученика, согласно определению 4.3 при помощи последовательных преобразований параметров \mathbf{u} . Рассмотрим преобразование:

$$\phi(t, \mathbf{u}) : \mathbb{R}^{N_{\text{tr}}} \rightarrow \mathbb{R}^{N_{\text{tr}} - 2n_t} \quad (4.12)$$

вектора \mathbf{u} , которое описывает удаление одного нейрона из t -го слоя учителя. Обозначим новый вектор параметров $\mathbf{v} = \phi(t, \mathbf{u})$, а элементы вектора, которые были удалены как $\bar{\mathbf{v}}$. Заметим, что векторы \mathbf{v} и $\bar{\mathbf{v}}$ являются случайными величинами.

Теорема 3. Пусть выполняются следующие условия:

1. апостериорное распределение $p(\mathbf{u}|\mathcal{D})$ параметров модели учителя является нормальным распределением (4.11);
2. число слоев модели учителя равняется числу слоев модели ученика $L = T$;
3. размеры соответствующих слоев не совпадают, другими словами, для всех t, l таких, что $t = l$ выполняется $n_t \leq n_l$.

Тогда апостериорное распределения параметров модели учителя $p(\mathbf{v}|\mathcal{D})$ также является нормальным распределением.

Доказательство. Не уменьшая общности, пусть $\phi(t, \mathbf{u})$ удаляет j -й нейрон в t -м слое, что является удалением j -й строки матрицы \mathbf{U}_t . Заметим, что удаление j -й строки матрицы \mathbf{U}_t влечет удаление j -й компоненты вектора \mathbf{z}_{t+1} , где

$$\mathbf{z}_t = \boldsymbol{\sigma} \circ \mathbf{U}_{t-1} \boldsymbol{\sigma} \circ \cdots \circ \mathbf{U}_2 \boldsymbol{\sigma} \circ \mathbf{U}_1 \mathbf{x}. \quad (4.13)$$

Удаление j -й компоненты вектора \mathbf{z}_{t+1} эквивалентно занулению j -го столбца матрицы \mathbf{U}_{t+1} . Заметим, что тогда предсказание модели не зависит от параметров j -й строки матрицы \mathbf{U}_t , а следовательно донными параметрами также можно пренебречь.

Найдем распределение вектора \mathbf{v} . Для поиска распределения вектора параметров после зануления j -го столбца матрицы \mathbf{U}_{t+1} воспользуемся формулой условной вероятности $p(\bar{\mathbf{v}}_1|\mathcal{D}, \boldsymbol{\nu}_1 = \mathbf{0})$, а для удаления j -й строки матрицы \mathbf{U}_t воспользуемся маргинализацией распределения $p(\bar{\mathbf{v}}_1|\mathcal{D}, \boldsymbol{\nu}_1 = \mathbf{0})$. Обозначим зануляемые параметры модели как $\boldsymbol{\nu}_1$, а удаляемые параметры как $\boldsymbol{\nu}_2$. Также обозначим все параметры, которые не были занулены как $\bar{\boldsymbol{\nu}}_1 = [\mathbf{v}^\top, \boldsymbol{\nu}_2^\top]$. Итоговое распределение параметров принимает вид:

$$p(\mathbf{v}|\mathcal{D}) = \int_{\boldsymbol{\nu}_2} p(\bar{\mathbf{v}}_1|\mathcal{D}, \boldsymbol{\nu}_1 = \mathbf{0}) d\boldsymbol{\nu}_2. \quad (4.14)$$

Из свойств нормального распределения следует, что распределение

$$p(\bar{\mathbf{v}}_1|\mathcal{D}, \boldsymbol{\nu}_1 = \mathbf{0}) \quad (4.15)$$

также является нормальным распределением с параметрами $\boldsymbol{\mu}, \boldsymbol{\Xi}$:

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{m}_{\bar{\boldsymbol{\nu}}_1} + \boldsymbol{\Sigma}_{\bar{\boldsymbol{\nu}}_1, \boldsymbol{\nu}_1} \boldsymbol{\Sigma}_{\boldsymbol{\nu}_1, \boldsymbol{\nu}_1}^{-1} (\mathbf{0} - \mathbf{m}_{\boldsymbol{\nu}_1}), \\ \boldsymbol{\Xi} &= \boldsymbol{\Sigma}_{\bar{\boldsymbol{\nu}}_1, \bar{\boldsymbol{\nu}}_1} - \boldsymbol{\Sigma}_{\bar{\boldsymbol{\nu}}_1, \boldsymbol{\nu}_1} \boldsymbol{\Sigma}_{\boldsymbol{\nu}_1, \boldsymbol{\nu}_1}^{-1} \boldsymbol{\Sigma}_{\boldsymbol{\nu}_1, \bar{\boldsymbol{\nu}}_1}, \end{aligned} \quad (4.16)$$

где введены обозначения $\mathbf{m}_{\bar{\nu}_1}, \mathbf{m}_{\nu_1}$ соответствует подвектору вектора \mathbf{m} , который относится к параметрам $\bar{\nu}_1$ и ν_1 соответственно. Ковариационная матрица $\Sigma_{\bar{\nu}_1, \nu_1}$ обозначает подматрицу матрицы Σ , которая соответствует ковариационной матрицей между параметрами $\bar{\nu}_1$ и ν_1 .

Распределение $p(\mathbf{v}|\mathfrak{D})$ найдем при помощи маргинализации распределения (4.15) по параметрам ν_2 . Используя свойства нормального распределения получаем распределения:

$$p(\mathbf{v}|\mathfrak{D}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{v}}, \boldsymbol{\Xi}_{\mathbf{v}, \mathbf{v}}), \quad (4.17)$$

где $\boldsymbol{\mu}_{\mathbf{v}}$ обозначает подвектор вектора $\boldsymbol{\mu}$, который относится к параметру \mathbf{v} , а матрица $\boldsymbol{\Xi}_{\mathbf{v}, \mathbf{v}}$ является подматрицей матрицы $\boldsymbol{\Xi}$, которая относится к вектору параметров \mathbf{v} . \square

Теорема 3 задает апостериорное распределение параметров (4.17) после зануления нейронов в модели нейросети — учителя. Заметим, что аналогичным образом можно удалить сразу подмножество нейронов в рамках одного слоя. В случае, если число нейронов отличается в нескольких слоях модели нейросети учителя, то выполняется последовательно применения отображения $\phi(t, \mathbf{u})$ для каждого t -го слоя.

Удаление слоя учителя. Проведем согласование модели учителя и модели ученика, согласно определению 4.3 при помощи последовательных преобразований вектора параметров \mathbf{u} . Рассмотрим преобразование:

$$\psi(t, \mathbf{u}) : \mathbb{R}^{N_{\text{tr}}} \rightarrow \mathbb{R}^{N_{\text{tr}} - n_t n_{t-1}} \quad (4.18)$$

вектора \mathbf{u} которое описывает удаление одного t -го слоя. Обозначим новый вектор параметров $\mathbf{v} = \psi(t, \mathbf{u})$, а элементы вектора, которые были удалены как $\bar{\mathbf{v}}$.

Теорема 4. Пусть выполняются следующие условия:

1. апостериорное распределение параметров $p(\mathbf{u}|\mathfrak{D})$ модели учителя является нормальным распределением (4.11);
2. соответствующие размеры слоев совпадают, $n_t = n_{t-1}$, то есть матрица \mathbf{U}_t является квадратной;
3. функция активации удовлетворяет свойству идемпотентности $\sigma \circ \sigma = \sigma$.

Тогда апостериорное распределения также описывается нормальным распределением со следующей плотностью распределения:

$$p(\mathbf{v}|\mathfrak{D}) = \mathcal{N}(\mathbf{m}_{\mathbf{v}} + \Sigma_{\mathbf{v}, \bar{\mathbf{v}}} \Sigma_{\bar{\mathbf{v}}, \bar{\mathbf{v}}}^{-1} (\mathbf{i} - \bar{\mathbf{v}}), \Sigma_{\mathbf{v}, \mathbf{v}} - \Sigma_{\mathbf{v}, \bar{\mathbf{v}}} \Sigma_{\bar{\mathbf{v}}, \bar{\mathbf{v}}}^{-1} \Sigma_{\bar{\mathbf{v}}, \mathbf{v}}), \quad (4.19)$$

где вектор \mathbf{i} задается следующим образом:

$$\mathbf{i} = [\underbrace{1, 0, \dots, 0}_{n_t}, \underbrace{0, 1, \dots, 0}_{n_t}, \underbrace{0, 0, 1, \dots, 0}_{n_t}, \underbrace{0, \dots, 1}_{n_t}]^T. \quad (4.20)$$

Доказательство. Рассмотрим структуру нейронной сети с T слоями и $T + 1$ слоем. Не уменьшая общности для удаления рассматривается t -й слой, для которого выполняются условия этой теоремы. Заметим, что если t -й слой нейронной сети с $T + 1$ слоем приравнять к единичной матрице, то он будет эквивалентным архитектуре с T слоями:

$$\begin{aligned} f &= \sigma \circ \mathbf{U}_{T+1} \sigma \circ \mathbf{U}_T \cdots \sigma \circ \mathbf{U}_t \sigma \circ \cdots \mathbf{U}_2 \sigma \circ \mathbf{U}_1 = \\ &= \sigma \circ \mathbf{U}_{T+1} \sigma \circ \mathbf{U}_T \cdots \sigma \circ \mathbf{I} \sigma \circ \cdots \mathbf{U}_2 \sigma \circ \mathbf{U}_1 = \\ &= \sigma \circ \mathbf{U}_{T+1} \sigma \circ \mathbf{U}_T \cdots \sigma \circ \sigma \circ \cdots \mathbf{U}_2 \sigma \circ \mathbf{U}_1 =^1 \\ &=^1 \sigma \circ \mathbf{U}_{T+1} \sigma \circ \mathbf{U}_T \cdots \sigma \circ \cdots \mathbf{U}_2 \sigma \circ \mathbf{U}_1. \end{aligned} \tag{4.21}$$

Получаем, что удаление t -го слоя нейросети эквивалентно приравнивают матрицы параметров t -го слоя к единичной матрице. Распределение параметров после приравнивая к единичной матрице вычисляется при помощи условного распределения. В силу общих свойств нормального распределения условное распределения также является нормальным распределением с параметрами $\boldsymbol{\mu}, \boldsymbol{\Sigma}$:

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{m}_{\mathbf{v}} + \boldsymbol{\Sigma}_{\mathbf{v}, \bar{\mathbf{v}}} \boldsymbol{\Sigma}_{\bar{\mathbf{v}}, \bar{\mathbf{v}}}^{-1} (\mathbf{i} - \bar{\mathbf{v}}), \\ \boldsymbol{\Sigma} &= \boldsymbol{\Sigma}_{\mathbf{v}, \mathbf{v}} - \boldsymbol{\Sigma}_{\mathbf{v}, \bar{\mathbf{v}}} \boldsymbol{\Sigma}_{\bar{\mathbf{v}}, \bar{\mathbf{v}}}^{-1} \boldsymbol{\Sigma}_{\bar{\mathbf{v}}, \mathbf{v}}), \end{aligned} \tag{4.22}$$

где вектор $\mathbf{m}_{\mathbf{v}}$ является подвектором вектора \mathbf{m} соответствующий параметрам \mathbf{v} , а матрица $\boldsymbol{\Sigma}_{\mathbf{v}, \bar{\mathbf{v}}}$ является подматрицей ковариационной матрицы $\boldsymbol{\Sigma}$ соответствующий векторам параметров \mathbf{v} и $\bar{\mathbf{v}}$. \square

Теорема 4 задает апостериорное распределение (4.19) параметров после удаления слоя нейросети. Полученное распределение $p(\mathbf{v}|\mathfrak{D})$ является оценкой апостериорного распределения модели без одного слоя.

Выполнение последовательных преобразований. Преобразования ϕ, ψ согласовывают пространства параметров учителя f и ученика g . После сопоставления параметрических моделей получаем, что параметры модели учителя и модели ученика принадлежат одному семейству 4.2.

4.3 Анализ качества байесовской дистилляции полно связанных нейронных сетей

Проводится вычислительный эксперимент для анализа предложенного метода дистилляции на основе апостериорного распределения параметров модели учителя.

Проанализируем модель на синтетической выборке. Выборка построенная следующим образом:

$$\begin{aligned} \mathbf{w} &= [w_j : w_j \sim \mathcal{N}(0, 1)]_{n \times 1}, \quad \mathbf{X} = [x_{ij} : x_{ij} \sim \mathcal{N}(0, 1)]_{m \times n}, \\ \mathbf{y} &= [y_i : y_i \sim \mathcal{N}(\mathbf{x}_i^\top \mathbf{w}, \beta)]_{m \times 1}, \end{aligned} \tag{4.23}$$

¹Выполняется в силу условия теоремы об идемпотентности функции активации.

где $\beta = 0,1$ — уровень шума в данных. В эксперименте число признаков $n = 10$, для обучения и тестирования было сгенерировано $m_{\text{train}} = 900$ и $m_{\text{test}} = 124$ объекта.

В качестве модели учителя рассматривалась модель — многослойный перцептрон с двумя скрытыми слоями (9.4). Матрицы линейных преобразований имеют размер:

$$\mathbf{U}_1 \in \mathbb{R}^{100 \times 10}, \quad \mathbf{U}_2 \in \mathbb{R}^{50 \times 100}, \quad \mathbf{U}_3 \in \mathbb{R}^{1 \times 50}. \quad (4.24)$$

В качестве функции активации была выбрана функция активации ReLu. Модель учителя предварительно обучена на основе вариационного вывода (9.8), где в качестве априорного распределения параметров выбрано стандартное нормальное распределение.

В качестве модели ученика были выбраны две конфигурации. Первая конфигурация получается путем удаления нейронов в модели учителя:

$$g = \sigma \circ \mathbf{W}_3 \sigma \circ \mathbf{W}_2 \sigma \circ \mathbf{W}_1, \quad (4.25)$$

где σ является нелинейной функцией активации, а матрицы линейных преобразований имеют размер:

$$\mathbf{W}_1 \in \mathbb{R}^{10 \times 10}, \quad \mathbf{W}_2 \in \mathbb{R}^{10 \times 10}, \quad \mathbf{W}_3 \in \mathbb{R}^{1 \times 10}. \quad (4.26)$$

В качестве функции активации была выбрана функция активации ReLu.

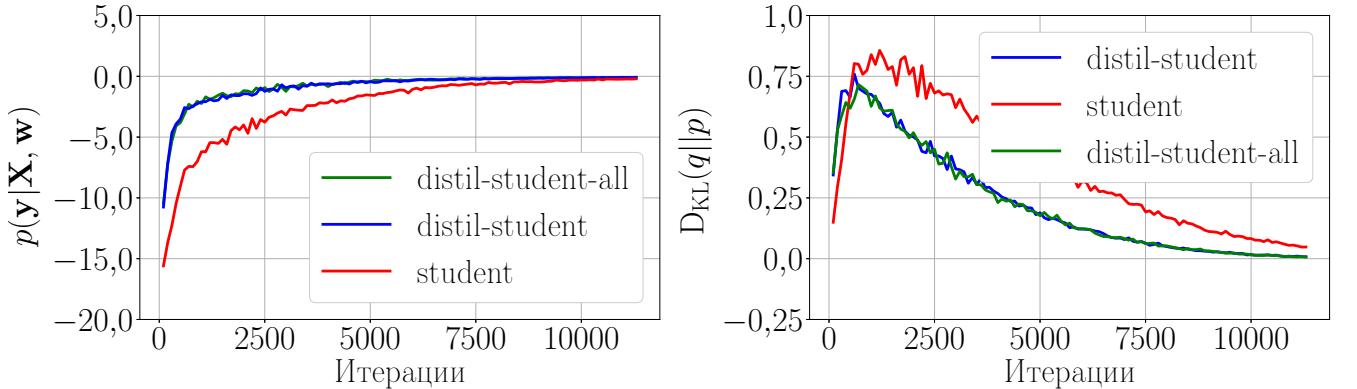


Рис. 21: Структура (4.25) модели ученика g . Слева: правдоподобие выборки в зависимости от номера итерации при обучении. Справа: KL-дивергенция между вариационным и априорным распределениями параметров модели.

Рис. 21 сравнивает модели ученика, со структурой (4.25). Представлено сравнение разных моделей: модель без дистилляции, где в качестве априорного распределения выбирается стандартное нормальное распределение (на легенде обозначается *student*); модель с частичной дистилляцией, где в качестве среднего значения параметров выбираются параметры согласно выражения (4.17), а ковариационная матрица была приравнена к единичной матрице (на легенде обозначается *distil-student*); модель с полной дистилляцией согласно выражения (4.17) (на легенде обозначается *distil-student-all*). Видно, что модели ученика, где в качестве априорного распределения выбраны распределения, основанные на апостериорном распределении учителя имеют большее правдоподобие, чем

модель где в качестве априорного распределения выбрано стандартное нормальное. Также заметим, что использования параметра среднего из апостериорного распределения дает основной вклад при дистилляции, так как качество моделей distil-student и distil-student-all совпадает.

Вторая конфигурация получается путем удаления слоя модели учителя:

$$g = \sigma \circ \mathbf{W}_2 \sigma \circ \mathbf{W}_1, \quad (4.27)$$

где σ является нелинейной функцией активации, а матрицы линейных преобразований имеют размер:

$$\mathbf{W}_1 \in \mathbb{R}^{1 \times 50}, \quad \mathbf{W}_2 \in \mathbb{R}^{50 \times 10}. \quad (4.28)$$

В качестве функции активации была выбрана функция активации ReLu.

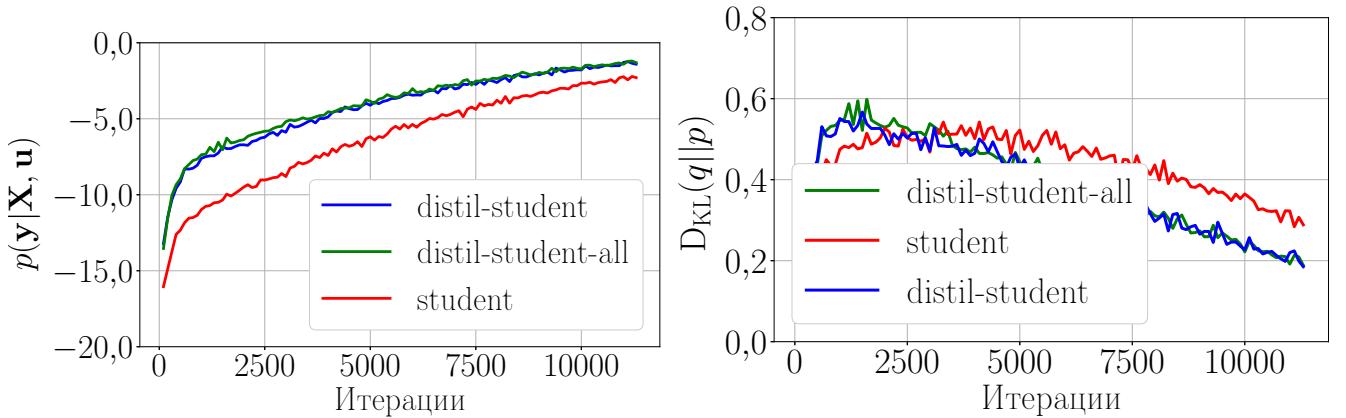


Рис. 22: Структура (4.27) модели ученика g . Слева: правдоподобие выборки в зависимости от номера итерации при обучении. Справа: KL–дивергенция между вариационным и априорным распределениями параметров модели.

Рис. 22 сравнивает модели ученика со структурой (4.27). Аналогично рис. 21, на рис. 22 представлено сравнение модели без дистилляции (student), модели с дистилляцией параметра среднего значение (distil-student) и модели с полной дистилляцией (distil-student-all). В рамках данного эксперимента, по дистилляции модели учителя в модель ученика с меньшим числом параметров получены результаты, которые подтверждают, что задание априорного распределения параметров ученика позволяет улучшить число итераций при выборе оптимальных параметров модели ученика.

В рамках данного эксперимента проводился анализ байесовского подхода к дистилляции на реальных данных. В качестве реальных данных выбрана выборка FashionMnist [155] которая является задачей классификации изображений на 10 классов.

В качестве модели учителя рассматривалась модель многослойный перцептрон с двумя скрытыми слоями (9.4). Матрицы линейных преобразований имеют размер:

$$\mathbf{U}_1 \in \mathbb{R}^{800 \times 784}, \quad \mathbf{U}_2 \in \mathbb{R}^{50 \times 800}, \quad \mathbf{U}_3 \in \mathbb{R}^{10 \times 50}, \quad (4.29)$$

В качестве функции активации была выбрана функция активации ReLu. Модель учителя предварительно обучена на основе вариационного вывода (9.8),

где в качестве априорного распределения параметров выбрано стандартное нормальное распределение.

В качестве модели ученика были выбрана конфигурация с одним скрытым слоем (4.27), где матрицы линейных преобразований имеют размер:

$$\mathbf{W}_1 \in \mathbb{R}^{50 \times 784}, \quad \mathbf{W}_2 \in \mathbb{R}^{50 \times 10}. \quad (4.30)$$

В качестве функции активации была выбрана функция активации ReLu.

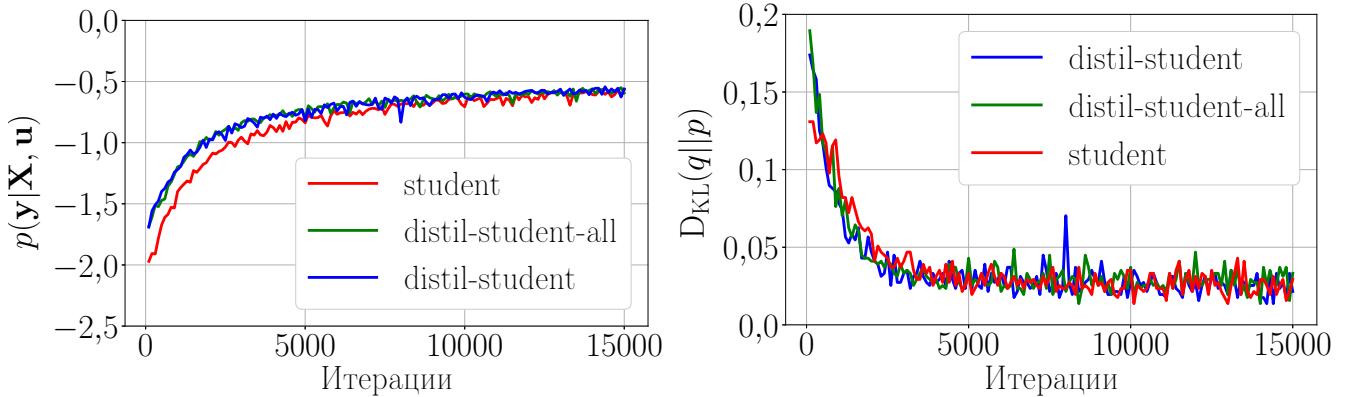


Рис. 23: Слева: правдоподобие выборки в зависимости от номера итерации при обучении. Справа: KL-дивергенция между вариационным и априорным распределениями параметров модели.

Рис. 23 сравнивает модели ученика с разными априорными распределениями параметров. Аналогично синтетическому эксперименту, модель, где в качестве априорного распределения использовалось стандартное нормальное распределение, сравнивалась с моделью, где параметры распределения определялись на основе формулы (4.19). Видно, что у моделей с заданием априорного распределения на основе апостериорного распределения параметров учителя правдоподобие выборки выше, чем у модели, где в качестве априорного распределения выбрано стандартное нормальное распределение.

В табл. 4 представлен результат вычислительного эксперимента. Для численного сравнения качества моделей выбрана разность площадей графика правдоподобия $p(\mathbf{y}|\mathbf{X}, \mathbf{u})$ между моделью student и моделями distil-student и distil-student-all соответственно:

$$S = \sum_s p(\mathbf{y}|\mathbf{X}, \mathbf{u}_s^s) - p(\mathbf{y}|\mathbf{X}, \mathbf{u}_{ds}^s), \quad (4.31)$$

где $\mathbf{u}_s^s, \mathbf{u}_{ds}^s$ обозначает параметры модели студента и модели дистиллированного студента после s -й итерации оптимизационного процесса. Заметим, что площадь S имеет знак: чем большее положительное число, тем дистиллированная модель лучше, чем модель построенная без учителя. В случае, если площадь S принимает отрицательное значение, то значит модель без дистилляции является лучше чем модель с дистилляцией. В рамках вычислительного эксперимента видно, что площадь S под графиками принимает положительные значения, то есть модели ученика полученные при помощи дистилляции являются лучше чем модель ученика без дистилляции.

Таблица 4: Сводная таблица результатов вычислительного эксперимента.

	teacher	student	distil-student	distil-student-all
Эксперимент на синтетической выборке (удаление нейрона)				
Структура	[10, 100, 50, 1]	[10, 10, 10, 1]	[10, 10, 10, 1]	[10, 10, 10, 1]
Число параметров	6050	210	210	210
Разность площадей	-	0	16559	16864
Эксперимент на синтетической выборке (удаление слоя)				
Структура	[10, 100, 50, 1]	[10, 50, 1]	[10, 50, 1]	[10, 50, 1]
Число параметров	6050	550	550	550
Разность площадей S	-	0	23310	25506
Эксперимент на выборке FashionMnist				
Структура	[784, 800, 50, 10]	[784, 50, 10]	[784, 50, 10]	[784, 50, 10]
Число параметров	667700	39700	39700	39700
Разность площадей S	-	0	1165	1145

5 Введение отношения порядка на множестве параметров аппроксимирующих моделей

В данной работе предлагается метод введения отношения порядка на множестве параметров сложных параметрических моделей, таких как нейросеть. Рассматривается порядок, заданный при помощи ковариационной матрицы градиентов функции ошибки по параметрам модели [37]. В работе [29] предложен итерационный метод для поиска ковариационной матрицы градиентов. Данный итерационный метод интегрируется в градиентный метод оптимизации Adam [38].

Множество параметров упорядочивается по возрастанию дисперсии: от параметра с минимальной дисперсией до параметра с максимальной дисперсией градиента функции ошибки по соответствующему параметру модели. Предполагается, что малая дисперсия градиента указывает на то, что соответствующий параметр можно зафиксировать.

Для задания порядка на множестве параметров при помощи ковариационной матрицы вводится предположение о том, что фиксация параметров происходит в момент, когда все параметры модели находятся в некоторой окрестности локального минимума функции ошибки. Данное условие накладывается для корректного использования итерационного метода поиска ковариационной матрицы градиентов.

Заданный порядок на множестве параметров модели используется для фиксации тех параметров модели, которые оказываются предстоящими с точки зрения заданного порядка. Сначала фиксируются те параметры, которые имеют минимальную дисперсию градиента в окрестности локального минимума функции ошибки.

Для анализа свойств предложенного метода задания порядка на множестве параметров проводился вычислительный эксперимент. В качестве моделей

рассматривались модели различной структурной сложности: линейные модели, нейросетевые модели. Предложенный метод задания порядка сравнивается с методом, в котором порядок задан произвольным образом.

5.1 Задача упорядочивания параметров аппроксимирующих моделей

Задана выборка:

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m, \quad \mathbf{x}_i \in \mathbb{X} = \mathbb{R}^n, \quad y_i \in \mathbb{Y}, \quad (5.1)$$

где n — размерность признакового пространства, m — число объектов в выборке. Пространство ответов $\mathbb{Y} = \mathbb{R}$ в случае задачи регрессии и $\mathbb{Y} = \{1, \dots, K\}$ в случае задачи классификации, где K — число классов.

Задано семейство моделей параметрических функций с наперед заданной структурой:

$$\begin{aligned} \mathfrak{F} &= \{f(\mathbf{w}) : \mathbb{X} \rightarrow \mathbb{Y} | \mathbf{w} \in \mathbb{R}^p\}, \\ \mathbf{h}(\mathbf{w}, \mathbf{x}) &= \mathbf{W}_1 \boldsymbol{\sigma}(\mathbf{W}_2 \boldsymbol{\sigma}(\dots \boldsymbol{\sigma}(\mathbf{W}_r \mathbf{x}) \dots)), \\ f_{\text{cl}}(\mathbf{w}, \mathbf{x}) &= \arg \max_{j \in \{1, \dots, K\}} \text{softmax}(\mathbf{h}(\mathbf{w}, \mathbf{x}))_j, \\ f_{\text{reg}}(\mathbf{w}, \mathbf{x}) &= \mathbf{h}(\mathbf{w}, \mathbf{x}), \end{aligned} \quad (5.2)$$

где p — размерность пространства параметров, r — число слоев нейросети, $\mathbf{w} = \text{vec}[\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_r]$, а $\boldsymbol{\sigma}$ — функция активации. В случае задачи регрессии структура модели имеет вид f_{reg} , а в случае классификации имеет вид f_{cl} . Задана функция потерь:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, \mathfrak{D}) &= \frac{1}{m} \sum_{i=1}^m l(\mathbf{x}_i, y_i, \mathbf{w}), \\ l_{\text{reg}}(\mathbf{x}, y, \mathbf{w}) &= (y - f(\mathbf{w}, \mathbf{x}))^2, \\ l_{\text{cl}}(\mathbf{x}, y, \mathbf{w}) &= - \sum_{j=1}^K ([y = j] \ln \text{softmax}_j(\mathbf{h}(\mathbf{w}, \mathbf{x}))), \end{aligned} \quad (5.3)$$

где l_{reg} — это функция ошибки на одном элементе для задачи регрессии, l_{cl} — для задачи классификации. Оптимальный вектор параметров $\hat{\mathbf{w}}$ получим минимизацией функции потерь:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^p} \mathcal{L}(\mathbf{w}, \mathfrak{D}). \quad (5.4)$$

Для поиска оптимальных параметров модели используется градиентный метод оптимизации:

$$\mathbf{w}_t = \mathbf{w}_{t-1} + \Delta \mathbf{w}(\mathbf{g}_{S,t}, \mathbf{w}_{t-1}, \mathbf{w}_{t-2}, \dots), \quad \mathbf{g}_{S,t} = \frac{\partial \mathcal{L}(\mathbf{w}_t, \mathbf{X}_S, \mathbf{Y}_S)}{\partial \mathbf{w}}, \quad (5.5)$$

где t — номер итерации, $\mathbf{g}_{S,t}$ — значение градиента на подвыборке размера S , $\Delta \mathbf{w}$ — приращение вектора параметров.

Порядок на множестве параметров модели задается при помощи ковариационной матрицы \mathbf{C} градиентов функции ошибки \mathcal{L} по параметрам модели \mathbf{w} .

Для вычисления ковариационной матрицы \mathbf{C} используется итерационная формула [29], которая вычисляется на каждой итерации (9.5) градиентного метода оптимизации параметров:

$$\mathbf{C}_t = (1 - \kappa_t) \mathbf{C}_{t-1} + \kappa_t (\mathbf{g}_{1,t} - \mathbf{g}_{S,t}) (\mathbf{g}_{1,t} - \mathbf{g}_{S,t})^\top, \quad (5.6)$$

где t — номер итерации, $\mathbf{g}_{S,t}$ — значение градиента на подвыборке размера S , $\mathbf{g}_{1,t}$ — значение градиента на первом элементе подвыборки, $\kappa_t = \frac{1}{t}$ — параметр сглаживания, \mathbf{C}_0 инициализируются из равномерного распределения.

Пусть известно t_0 — число итераций, после которого все параметры находятся в некоторой локальной окрестности минимума, тогда, как показано в работе [29], матрица \mathbf{C}_{t_0} аппроксимирует истинную ковариационную матрицу \mathbf{C} . Ковариационная матрица \mathbf{C}_{t_0} используется для упорядочения параметров модели \mathbf{w}_{t_0} .

Пусть \mathcal{I} — упорядоченный вектор индексов $[1, 2, \dots, p]$. Обозначим $\mathcal{I}_{\mathbf{w}_{t_0}}$ вектор индексов, порядок которого задан при помощи ковариационной матрицы \mathbf{C}_{t_0} .

Например, если ковариационная матрица \mathbf{C}_{t_0} имеет вид

$$\begin{bmatrix} 0,3 & 0 & 0 \\ 0 & 0,2 & 0 \\ 0 & 0 & 0,25 \end{bmatrix},$$

то вектор индексов $\mathcal{I}_{\mathbf{w}_{t_0}} = [3, 1, 2]$.

5.2 Фиксация параметров модели в процессе обучения

Для фиксации параметров \mathbf{w}_{t_0} при помощи вектора индексов $\mathcal{I}_{\mathbf{w}_{t_0}}$ используется бинарный вектор $\boldsymbol{\alpha}(k)$:

$$\alpha_i(k) = \begin{cases} 1, & \text{если } \mathcal{I}_{\mathbf{w}_{t_0}}[j] \leq k; \\ 0 & \text{иначе,} \end{cases} \quad (5.7)$$

где k — число фиксирующих параметров.

Учитывая (9.7), уравнение (9.5) приводится к виду

$$\mathbf{w}_t = \mathbf{w}_{t-1} + \boldsymbol{\alpha}(k) \cdot \Delta \mathbf{w}(\mathbf{g}_{S,t}, \mathbf{w}_{t-1}, \mathbf{w}_{t-2}, \dots), \quad (5.8)$$

где t — номер итерации, $\mathbf{g}_{S,t}$ — значение градиента на подвыборке размера S , $\Delta \mathbf{w}$ — приращение вектора параметров. После умножения на бинарный вектор $\boldsymbol{\alpha}$ часть параметров не оптимизируется, что приводит к фиксации параметров.

5.3 Вычислительный эксперимент по упорядочиванию параметров

Для анализа результатов, полученных предложенным алгоритмом, проводится вычислительный эксперимент. В качестве данных используются синтетические

Таблица 5: Описание выборок, используемых в эксперименте

Выборка, \mathfrak{D}	Тип	Число признаков, n	Модель	Число параметров, p
Boston Housing	Регрессия	13	Нейросеть	301
MNIST	Классификация	784	Нейросеть	7960
Synthetic 3	Регрессия	200	Линейная	200
Synthetic 2	Классификация	200	Линейная	200
Synthetic 1	Регрессия	200	Нейросеть	4041

и реальные данные, которые описаны в табл. 5. Выборки MNIST [151] и Boston Housing [39] рассматриваются в качестве реальных данных, для которых решается задача классификации и регрессии соответственно. Синтетические выборки задаются следующим образом:

$$\begin{aligned}\mathfrak{D}_{\text{reg}} &= \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n), y_i \sim \mathcal{N}(\mathbf{w}^\top \mathbf{x}_i, \mathbf{I}_n), \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)\}, \\ \mathfrak{D}_{\text{cl}} &= \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n), y_i \sim \mathcal{Be}(\mathbf{w}^\top \mathbf{x}_i), \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n)\}.\end{aligned}\quad (5.9)$$

В качестве аппроксимирующих моделей рассматриваются линейные и нейросетевые модели (9.3). В качестве функции ошибки для задачи регрессии рассматривается MSELoss , а для задачи классификации — CrossEntropyLoss (9.4).

Предварительно для каждой модели и выборки определяется число t_0 — номер итерации, после которой все параметры модели находятся в некоторой окрестности локального минимума. Параметр t_0 устанавливается экспериментальным путем для каждой модели и выборки отдельно из условия, что качество модели меняется незначительно при числе итераций $t > t_0$.

После t_0 шагов алгоритма оптимизации часть параметров модели фиксируется в соответствии с формулами (9.7), (9.8). Результат работы получается усреднением по 25 независимым запускам оптимизации модели. Значение функции ошибки \mathcal{L} усредняется по разным запускам алгоритма оптимизации. В ходе эксперимента проводится анализ вектора $\boldsymbol{\alpha}$, который также усредняется по разным запускам алгоритма оптимизации. Усредненное значение бинарного вектора $\boldsymbol{\alpha}$ обозначим $\hat{\boldsymbol{\alpha}}$.

Выборка Synthetic 1. Эксперимент проводился на синтетически построенных данных. В качестве модели использовалась двухслойная нейросеть — перцептрон. На рис. 24 показаны графики зависимости функции потерь \mathcal{L} от числа фиксируемых параметров. В случае фиксации параметров предложенным методом функция потерь \mathcal{L} растет медленней, чем в случае фиксации параметров произвольным образом.

На рис. 25 показана зависимость векторов $\hat{\boldsymbol{\alpha}}(k)$ от числа фиксируемых параметров. Каждый столбец соответствует одному вектору $\hat{\boldsymbol{\alpha}}(k)$. На рис. 25a, 25c видно, что $\hat{\boldsymbol{\alpha}}(k)$ имеет большое число компонент вектора, близких к 1. Так

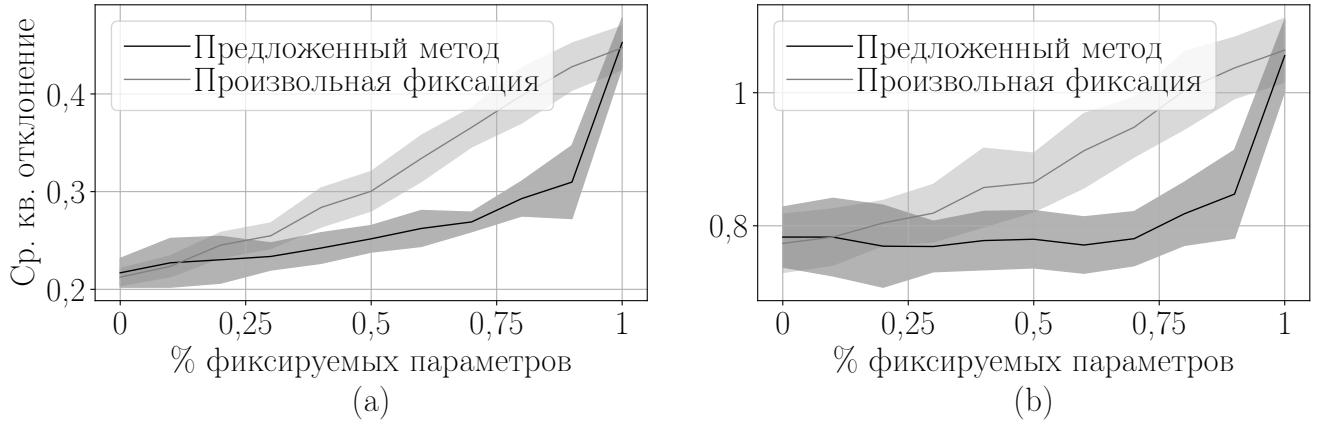


Рис. 24: Зависимость качества модели от числа зафиксированных параметров:
а) на обучающей выборке; б) на тестовой выборке

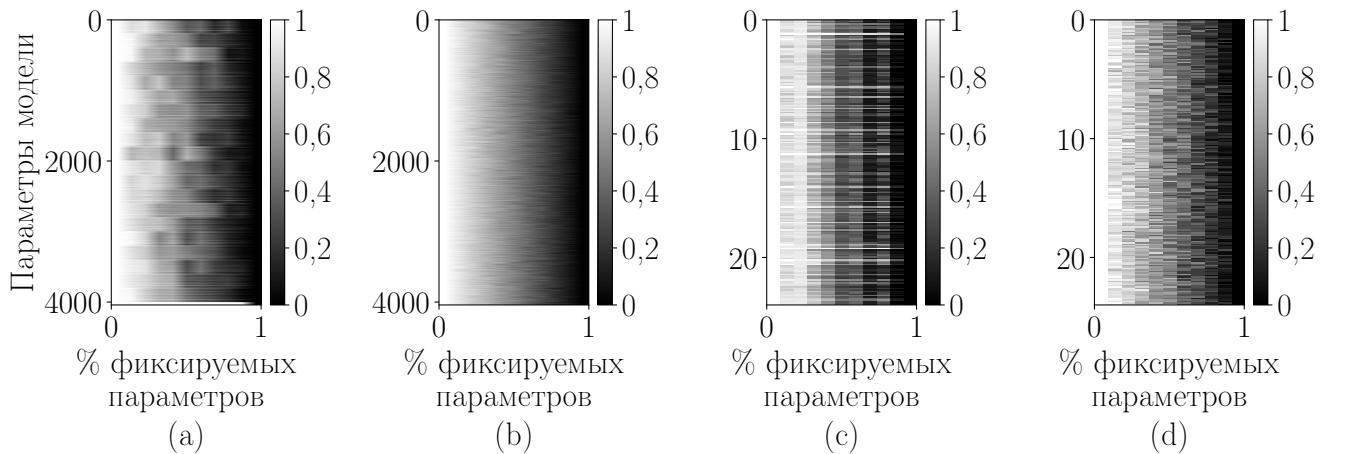


Рис. 25: Визуализация векторов $\hat{\alpha}(k)$ в зависимости от числа фиксируемых параметров: а) все параметры модели упорядочены предложенным методом; б) все параметры модели упорядочены произвольным образом; в) часть параметров модели упорядочена предложенным методом; г) часть параметров модели упорядочена произвольным образом

как $\hat{\alpha}(k)$ является усреднением вектора с компонентами 0 или 1, то предложенный порядок задает некоторый устойчивый порядок на множестве параметров модели. На рис. 25б, 25д видно, что в случае произвольной фиксации параметров компоненты вектора $\hat{\alpha}(k)$ имеют одинаковые значения, следовательно, никакого порядка на множестве параметров нет.

Выборка Boston Housing. Эксперимент проводился на реальных данных. На рис. 26 показаны графики зависимости функции потерь \mathcal{L} от числа фиксируемых параметров. В случае фиксации параметров предложенным методом, функция потерь \mathcal{L} растет так же, как и в случае фиксации параметров произвольным образом. Данный результат следует из того, что нейросеть оказалась избыточно сложной моделью с большим числом параметров. После фиксации

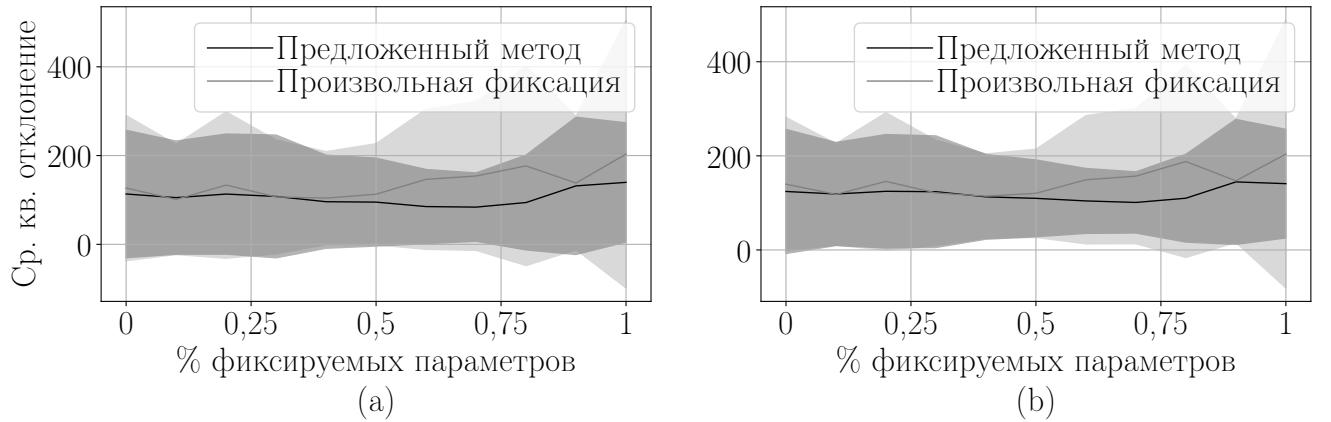


Рис. 26: Зависимость качества модели от числа зафиксированных параметров:
а) на обучающей выборке; б) на тестовой выборке

значимого числа параметров у модели оставалась большое число параметров для дообучения.

На рис. 27 показана зависимость векторов $\hat{\alpha}(k)$ от числа фиксируемых параметров. На рис. 27а, 27с видно, что $\hat{\alpha}(k)$ меняется незначительно от запуска к запуску алгоритма. Следовательно, предложенный порядок задает устойчивый к разным запускам порядок на множестве параметров модели. На рис. 27б, 27д видно, что в случае произвольной фиксации параметров вектор $\hat{\alpha}(k)$ является произвольным и никакого порядка на множестве параметров нет.

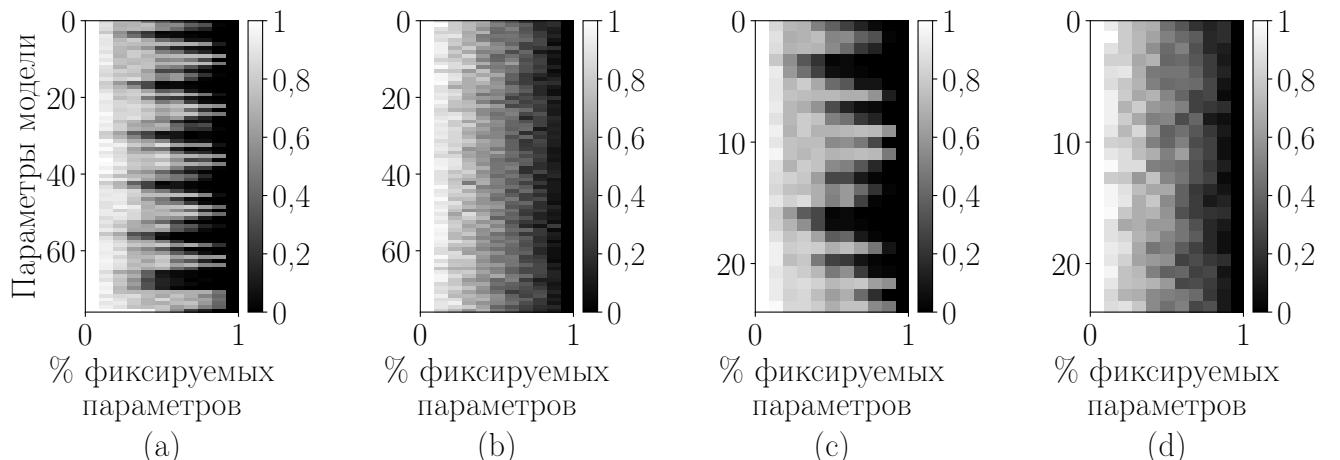


Рис. 27: Визуализация векторов $\hat{\alpha}(k)$ в зависимости от числа фиксируемых параметров: а) все параметры модели упорядочены предложенным методом; б) все параметры модели упорядочены произвольным образом; в) часть параметров модели, упорядочена предложенным методом; г) часть параметров модели упорядочена произвольным образом

Выборка Synthetic 3. Эксперимент проводился на синтетически построенных данных. В качестве модели использовалась линейная модель регрессии.

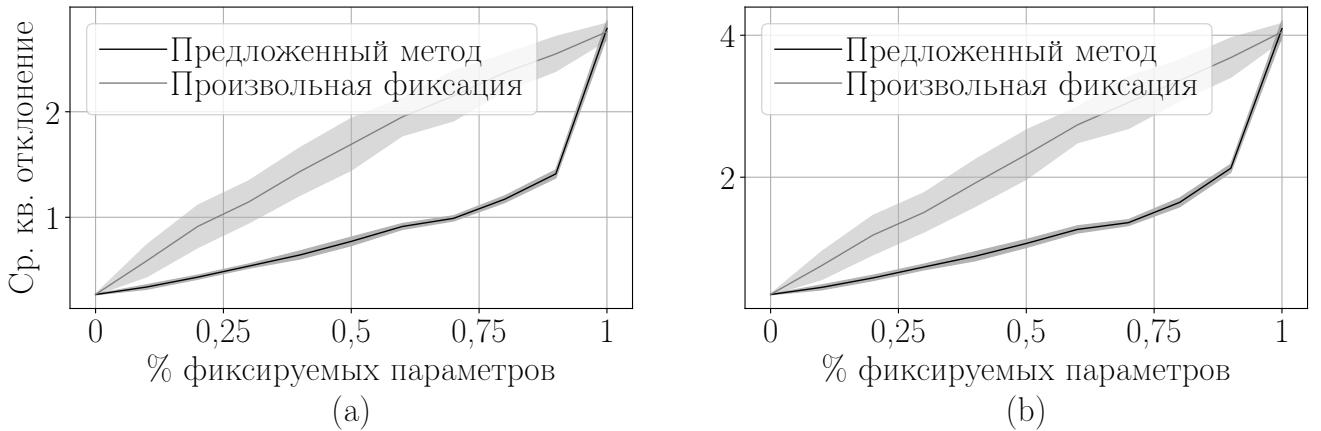


Рис. 28: Зависимость качества модели от числа зафиксированных параметров:
а) на обучающей выборке; б) на тестовой выборке

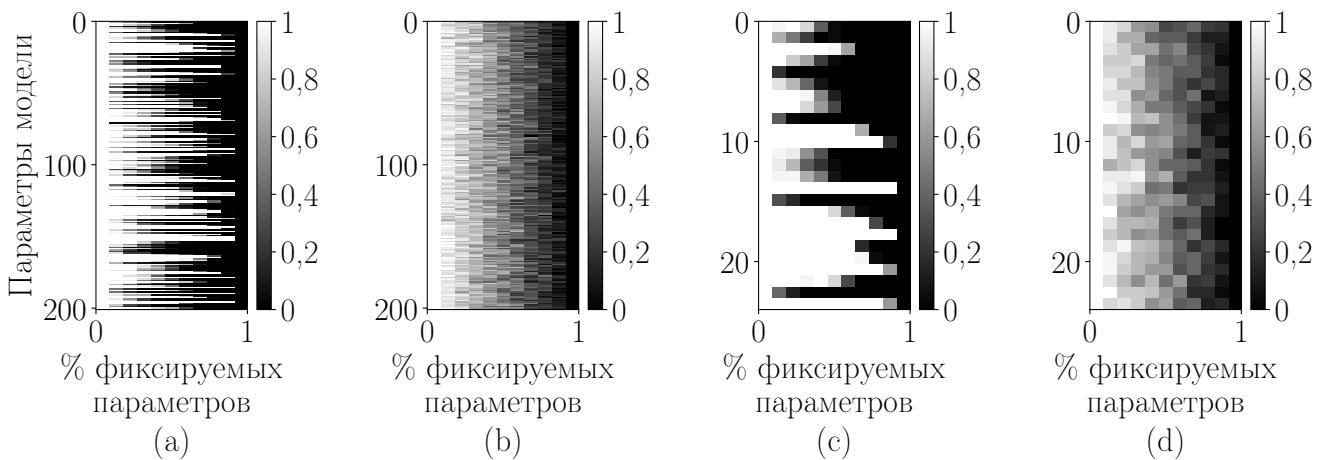


Рис. 29: Визуализация векторов $\hat{\alpha}(k)$ в зависимости от числа фиксируемых параметров: а) все параметры модели упорядочены предложенным методом; б) все параметры модели упорядочены произвольным образом; в) часть параметров модели, упорядочены предложенным методом; г) часть параметров модели упорядочены произвольным образом

На рис. 28 показаны графики зависимости функции потерь \mathcal{L} от числа фиксируемых параметров. В случае фиксации параметров предложенным методом функция потерь \mathcal{L} растет значительно медленней в сравнении со случаем фиксации параметров произвольным образом. Дисперсия функции ошибки также значительно меньше в случае фиксации параметров предложенным методом.

На рис. 29 показано, что вектора $\hat{\alpha}(k)$ не меняются от запуска к запуску. Так как данная модель линейная, то порядок на параметрах модели индуцирует некоторый порядок на множестве признаков.

Выборка MNIST. В эксперименте рассматривался двухслойный перцептрон для классификации изображений. В качестве входных данных рассматривались изображения размера 28×28 , на которых изображены цифры.

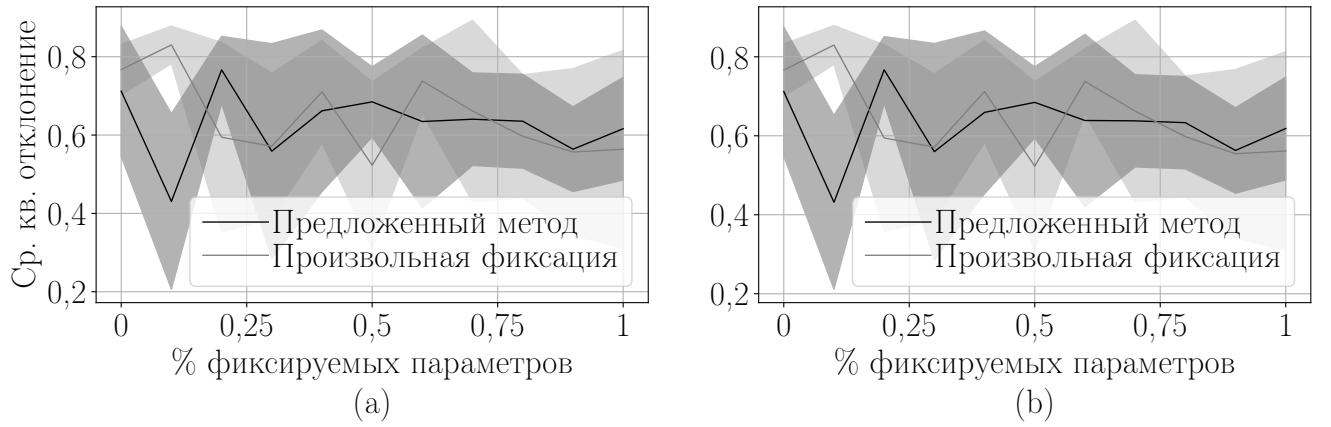


Рис. 30: Зависимость качества модели от числа зафиксированных параметров:
а) на обучающей выборке; б) на тестовой выборке

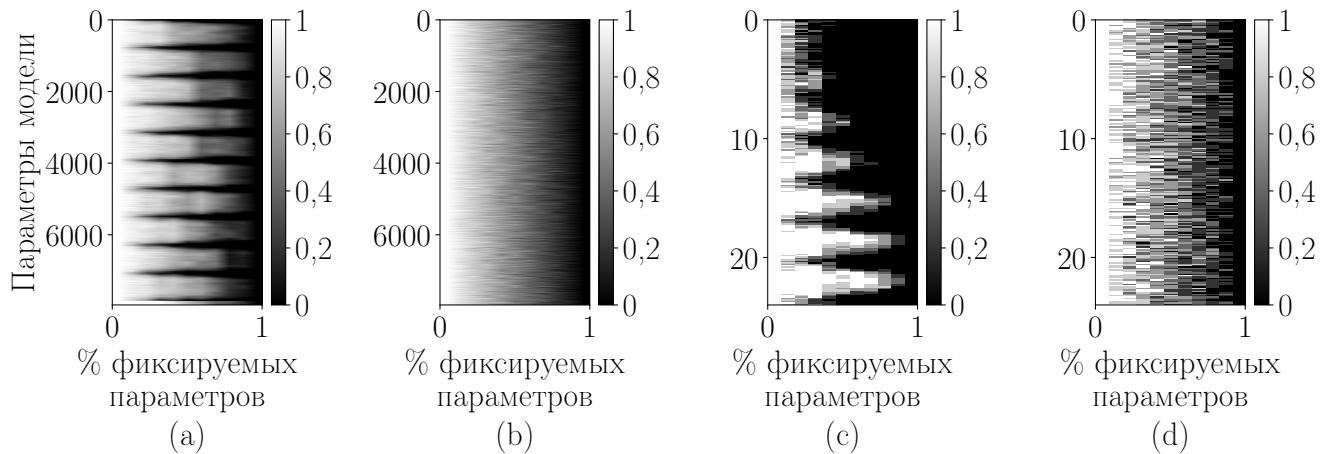


Рис. 31: Визуализация векторов $\hat{\alpha}(k)$ в зависимости от числа фиксируемых параметров: а) все параметры модели упорядочены предложенным методом; б) все параметры модели упорядочены произвольным образом; в) часть параметров модели упорядочена предложенным методом; г) часть параметров модели упорядочена произвольным образом

На рис. 30 показано, что графики функции ошибки похожи в случае фиксации параметров параметров предложенным методом и в случае произвольной фиксации. Данный результат есть следствие того факта, что нейросеть является заведомо переусложненной моделью с большим числом параметров. После фиксации большого числа параметров у нейросети все еще остается значимое число параметров модели для дообучения.

На рис. 31 показано, что в случае модели со значимым числом оптимизационных параметров, предложенный метод упорядочения параметров устойчив от запуску к запуску.

6 Релевантность параметров параметрических моделей

Данная работа посвящена прореживанию структуры сети. Предлагается удалять наименее релевантные параметры модели. Под релевантностью [45] подразумевается то, насколько параметр влияет на функцию ошибки. Малая релевантность указывает на то, что удаление этого параметра не влечет значимого изменения функции ошибки. Метод предлагает построение исходной избыточной сложности нейросети с большим количеством избыточных параметров. Для определения релевантности параметров предлагается оптимизировать параметры и гиперпараметры в единой процедуре. Для удаления параметров предлагаются использовать метод Белсли [44].

6.1 Постановка задачи к назначению релевантности параметрам модели

Задана выборка

$$\mathfrak{D} = \{\mathbf{x}_i, y_i\}, i = 1, \dots, N, \quad (6.1)$$

где $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{1, \dots, Y\}$, Y — число классов. Рассмотрим модель $f(\mathbf{x}, \mathbf{w}) : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \{1, \dots, Y\}$, где $\mathbf{w} \in \mathbb{R}^n$ — пространство параметров модели,

$$f(\mathbf{x}, \mathbf{w}) = \text{softmax}(f_1(f_2(\dots(f_l(\mathbf{x}, \mathbf{w})))), \quad (6.2)$$

где $f_i(\mathbf{x}, \mathbf{w}) = \tanh(\mathbf{w}^\top \mathbf{x})$, l — число слоев нейронной сети, $i \in \{1 \dots l\}$. Параметр w_j модели f называется активным, если $w_j \neq 0$. Множество индексов активных параметров обозначим $\mathcal{A} \subset \mathcal{J} = \{1, \dots, n\}$. Задано пространство параметров модели:

$$\mathbb{W}_{\mathcal{A}} = \{\mathbf{w} \in \mathbb{R}^n | w_j \neq 0, j \in \mathcal{A}\}, \quad (6.3)$$

Для модели f с множеством индексов активных параметров \mathcal{A} и соответствующего ей вектора параметров $\mathbf{w} \in \mathbb{W}_{\mathcal{A}}$ определим логарифмическую функцию правдоподобия выборки:

$$\mathcal{L}_{\mathfrak{D}}(\mathfrak{D}, \mathcal{A}, \mathbf{w}) = \log p(\mathfrak{D} | \mathcal{A}, \mathbf{w}), \quad (6.4)$$

где $p(\mathfrak{D} | \mathcal{A}, \mathbf{w})$ — апостериорная вероятность выборки \mathfrak{D} при заданных \mathbf{w}, \mathcal{A} . Оптимальные значения \mathbf{w}, \mathcal{A} находятся из минимизации $-\mathcal{L}_{\mathcal{A}}(\mathfrak{D}, \mathcal{A})$ — логарифма правдоподобия модели:

$$\mathcal{L}_{\mathcal{A}}(\mathfrak{D}, \mathcal{A}) = \log p(\mathfrak{D} | \mathcal{A}) = \log \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} p(\mathfrak{D} | \mathbf{w}) p(\mathbf{w} | \mathcal{A}) d\mathbf{w}, \quad (6.5)$$

где $p(\mathbf{w} | \mathcal{A})$ — априорная вероятность вектора параметров в пространстве $\mathbb{W}_{\mathcal{J}}$.

Так как вычисление интеграла (6.5) является вычислительно сложной задачей, рассмотрим вариационный подход [86] для решения этой задачи. Пусть задано распределение q :

$$q(\mathbf{w}) \sim \mathcal{N}(\mathbf{m}, \mathbf{A}_{\text{ps}}^{-1}), \quad (6.6)$$

где $\mathbf{m}, \mathbf{A}_{\text{ps}}^{-1}$ — вектор средних и матрица ковариации, аппроксимирующее неизвестное апостериорное распределение $p(\mathbf{w}|\mathfrak{D}, \mathcal{A})$:

$$p(\mathbf{w}|\mathcal{A}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{A}_{\text{pr}}^{-1}), \quad (6.7)$$

где $\boldsymbol{\mu}, \mathbf{A}_{\text{pr}}^{-1}$ — вектор средних и матрица ковариации.

Приблизим интеграл (6.5) методом предложенном в [86]:

$$\begin{aligned} \mathcal{L}_{\mathcal{A}}(\mathfrak{D}, \mathcal{A}) &= \log p(\mathfrak{D}|\mathcal{A}) = \\ &= \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} q(\mathbf{w}) \log \frac{p(\mathfrak{D}, \mathbf{w}|\mathcal{A})}{q(\mathbf{w})} d\mathbf{w} - \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} q(\mathbf{w}) \log \frac{p(\mathbf{w}|\mathfrak{D}, \mathcal{A})}{q(\mathbf{w})} d\mathbf{w} \approx \\ &\approx \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} q(\mathbf{w}) \log \frac{p(\mathfrak{D}, \mathbf{w}|\mathcal{A})}{q(\mathbf{w})} d\mathbf{w} = \\ &= \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} q(\mathbf{w}) \log \frac{p(\mathbf{w}|\mathcal{A})}{q(\mathbf{w})} d\mathbf{w} + \int_{\mathbf{w} \in \mathbb{W}_{\mathcal{J}}} q(\mathbf{w}) \log p(\mathfrak{D}|\mathcal{A}, \mathbf{w}) d\mathbf{w} = \\ &= \mathcal{L}_{\mathbf{w}}(\mathfrak{D}, \mathcal{A}, \mathbf{w}) + \mathcal{L}_E(\mathfrak{D}, \mathcal{A}). \end{aligned} \quad (6.8)$$

Первое слагаемое формулы (6.8) — это сложность модели. Оно определяется расстоянием Кульбака-Лейблера:

$$\mathcal{L}_{\mathbf{w}}(\mathfrak{D}, \mathcal{A}, \mathbf{w}) = -D_{KL}(q(\mathbf{w})||p(\mathbf{w}|\mathcal{A})). \quad (6.9)$$

Второе слагаемое формулы (6.8) является матожиданием правдоподобия выборки $\mathcal{L}_{\mathfrak{D}}(\mathfrak{D}, \mathcal{A}, \mathbf{w})$. В данной работе оно является функцией ошибки:

$$\mathcal{L}_E(\mathfrak{D}, \mathcal{A}) = \mathbb{E}_{\mathbf{w} \sim q} \mathcal{L}_{\mathfrak{D}}(\mathbf{y}, \mathfrak{D}, \mathcal{A}, \mathbf{w}). \quad (6.10)$$

Требуется найти параметры, доставляющие минимум суммарному функционалу потерь $\mathcal{L}_{\mathcal{A}}(\mathfrak{D}, \mathcal{A}, \mathbf{w})$ из (6.8):

$$\hat{\mathbf{w}} = \arg \min_{\mathcal{A} \subset \mathcal{J}, \mathbf{w} \in \mathbb{W}_{\mathcal{A}}} -\mathcal{L}_{\mathcal{A}}(\mathfrak{D}, \mathcal{A}, \mathbf{w}) == \arg \min_{\mathcal{A} \subset \mathcal{J}, \mathbf{w} \in \mathbb{W}_{\mathcal{A}}} D_{KL}(q(\mathbf{w})||p(\mathbf{w}|\mathcal{A})) - \mathcal{L}_{\mathfrak{D}}(\mathfrak{D}, \mathcal{A}, \mathbf{w}). \quad (6.11)$$

Случайное удаление. Метод случайного удаления заключается в том, что случайным образом удаляется некоторый параметр w_{ξ} из множества активных параметров сети. Индекс параметра ξ из равномерного распределения случайная величина, предположительно доставляющая оптимум в (6.11).

$$\xi \sim \mathcal{U}(\mathcal{A}). \quad (6.12)$$

Оптимальное прореживание. Метод оптимального прореживания [45] использует вторую производную целевой функции (6.4) по параметрам для определения нерелевантных параметров. Рассмотрим функцию потерь \mathcal{L} (6.4) разложенную в ряд Тейлора в некоторой окрестности вектора параметров \mathbf{w} :

$$\delta \mathcal{L} = \sum_{j \in \mathcal{A}} g_j \delta w_j + \frac{1}{2} \sum_{i,j \in \mathcal{A}} h_{ij} \delta w_i \delta w_j + O(||\delta \mathbf{w}||^3), \quad (6.13)$$

где δw_j — компоненты вектора $\delta \mathbf{w}$, g_j — компоненты вектора градиента $\nabla \mathcal{L}$, а h_{ij} — компоненты гессиана \mathbf{H} :

$$g_j = \frac{\partial \mathcal{L}}{\partial w_j}, \quad h_{ij} = \frac{\partial^2 \mathcal{L}}{\partial w_i \partial w_j}. \quad (6.14)$$

Задача является вычислительно сложной в силу размерности матрицы \mathbf{H} . Введем следующее предположение [45], о том что удаление нескольких параметров приводит к такому же изменению функции потерь \mathcal{L} , как и суммарное изменение при индивидуальном удалении:

$$\delta \mathcal{L} = \sum_{j \in \mathcal{A}} \delta \mathcal{L}_j, \quad (6.15)$$

где \mathcal{A} — множество активных параметров, $\delta \mathcal{L}_j$ — изменение функции потерь, при удалении одного параметра w_j .

В силу данного предположения будем рассматривать только диагональные элементы матрицы \mathbf{H} . После введенного предположения, выражение (6.13) принимает вид

$$\delta \mathcal{L} = \frac{1}{2} \sum_{j \in \mathcal{A}} h_{jj} \delta w_j^2, \quad (6.16)$$

Получаем следующую задачу оптимизации:

$$\xi = \arg \min_{j \in \mathcal{A}} h_{jj} \frac{w_j^2}{2}, \quad (6.17)$$

где ξ — индекс наименее релевантного, удаляемого параметра, предположительно доставляющая оптимум в (6.11).

Удаление неинформативных параметров с помощью вариационного вывода. Для удаления параметров в работе [160] предлагается удалить параметры, которые имеют максимальное отношение плотности $p(\mathbf{w}|\mathcal{A})$ априорной вероятности в нуле к плотности вероятности априорной вероятности в математическом ожидании параметра.

Для гауссовского распределения с диагональной матрицей ковариации получаем:

$$p_j(\mathbf{w}|\mathcal{A})(x) = \frac{1}{\sqrt{2\sigma_j^2}} \exp\left(-\frac{(x - \mu_j)^2}{2\sigma_j^2}\right). \quad (6.18)$$

Разделив плотность вероятности в нуле к плотности в математическом ожидании

$$\frac{p_j(\mathbf{w}|\mathcal{A})(0)}{p_j(\mathbf{w}|\mathcal{A})(\mu_j)} = \exp\left(-\frac{\mu_j^2}{2\sigma_j^2}\right), \quad (6.19)$$

Получаем следующую задачу оптимизации:

$$\xi = \arg \min_{j \in \mathcal{A}} \left| \frac{\mu_j}{\sigma_j} \right|, \quad (6.20)$$

где ξ — индекс наименее релевантного, удаляемого параметра.

Прореживание сети на основе метода Белсли. Предлагается метод основанный на модификации метода Белсли. Пусть \mathbf{w} — вектор параметров доставляющий минимум функционалу потерь \mathcal{L} на множестве $\mathbb{W}_{\mathcal{A}}$, а \mathbf{A}_{ps} соответствующая ему ковариационная матрица.

Выполним сингулярное разложение матрицы

$$\mathbf{A}_{ps} = \mathbf{U}\Lambda\mathbf{V}^T. \quad (6.21)$$

Индекс обусловленности η_j определим как отношение максимального элемента к j -му элементу матрицы Λ . Для нахождения мультиколлиниарных признаков требуется найти индекс ξ вида:

$$\xi = \arg \max_{j \in \mathcal{A}} \eta_j. \quad (6.22)$$

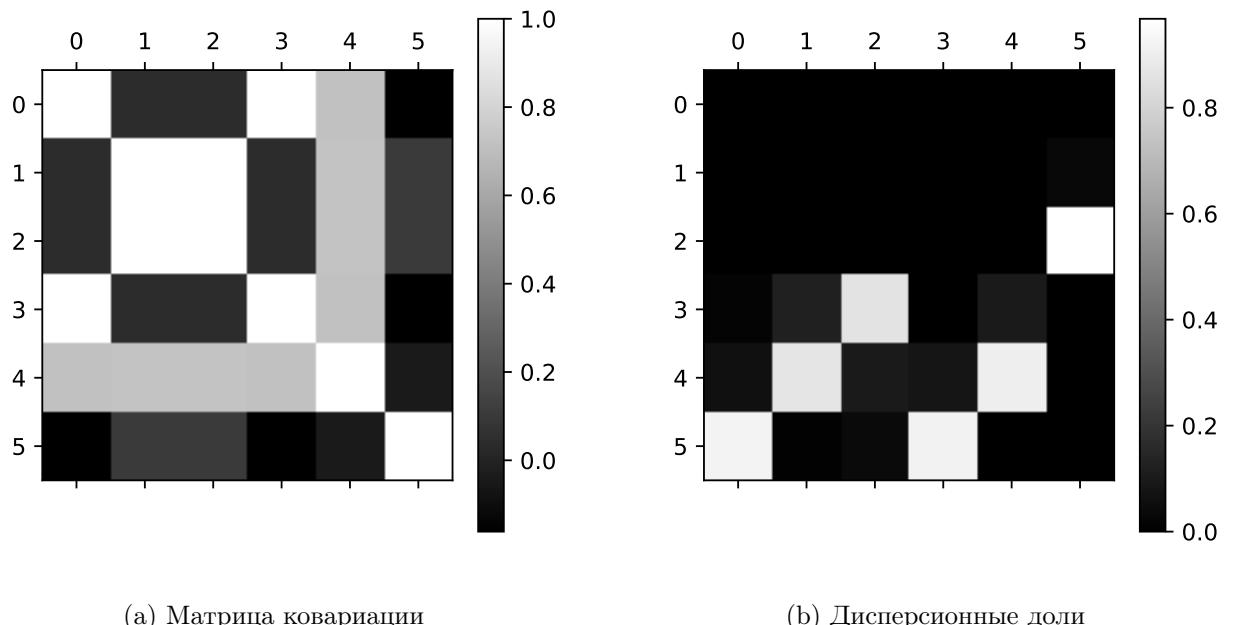


Рис. 32: Иллюстрация метода Белсли

Таблица 6: Иллюстрация метода Белсли

η	q_1	q_2	q_3	q_4	q_5	q_6
1.0	$2 \cdot 10^{-17}$	$4 \cdot 10^{-17}$	$1 \cdot 10^{-16}$	$2 \cdot 10^{-17}$	$6 \cdot 10^{-17}$	$3 \cdot 10^{-4}$
1.5	$5 \cdot 10^{-17}$	$9 \cdot 10^{-17}$	$2 \cdot 10^{-16}$	$5 \cdot 10^{-17}$	$3 \cdot 10^{-20}$	$3 \cdot 10^{-2}$
3.3	$9 \cdot 10^{-18}$	$1 \cdot 10^{-17}$	$2 \cdot 10^{-17}$	$9 \cdot 10^{-18}$	$2 \cdot 10^{-19}$	$9 \cdot 10^{-1}$
$2 \cdot 10^{15}$	$1 \cdot 10^{-2}$	$1 \cdot 10^{-1}$	$8 \cdot 10^{-1}$	$2 \cdot 10^{-3}$	$9 \cdot 10^{-2}$	$1 \cdot 10^{17}$
$8 \cdot 10^{15}$	$6 \cdot 10^{-2}$	$8 \cdot 10^{-1}$	$9 \cdot 10^{-2}$	$8 \cdot 10^{-2}$	$9 \cdot 10^{-1}$	$2 \cdot 10^{17}$
$1 \cdot 10^{16}$	$9 \cdot 10^{-1}$	$1 \cdot 10^{-2}$	$4 \cdot 10^{-2}$	$9 \cdot 10^{-1}$	$1 \cdot 10^{-3}$	$5 \cdot 10^{-21}$

Дисперсионный долевой коэффициент q_{ij} определим как вклад j -го признака в дисперсию i -го элемента вектора параметра \mathbf{w} :

$$q_{ij} = \frac{u_{ij}^2 / \lambda_{jj}}{\sum_{j=1}^n u_{ij}^2 / \lambda_{jj}}. \quad (6.23)$$

Большие значение дисперсионных долей указывают на наличие зависимости между параметрами. Находим долевые коэффициенты, которые вносят максимальный вклад в дисперсию параметра w_ξ :

$$\zeta = \arg \max_{j \in \mathcal{A}} q_{\xi j}. \quad (6.24)$$

Параметр с индексом ζ определим как наименее релевантный параметр нейросети.

Проиллюстрируем принцип работы метода Белсли на примере. Рассмотрим данные порожденные следующим образом:

$$\mathbf{w} = \begin{bmatrix} \sin(x) \\ \cos(x) \\ 2+\cos(x) \\ 2+\sin(x) \\ \cos(x) + \sin(x) \\ x \end{bmatrix} \quad (6.25)$$

с матрицей ковариации на рис. 32.a, где $x \in [0.0, 0.02, \dots, 20.0]$.

В табл. 6 приведены индексы обусловленности и соответствующие им дисперсионные доли, которые также изображены на рис. 32.b. Согласно этим данным, максимальный индекс обусловленности $\eta_6 = 1.2 \cdot 10^{16}$. Ему соответствуют максимальные дисперсионные доли признаков с индексами 1 и 4, которые, как видно из построения выборки, являются линейно зависимые.

6.2 Анализ разных подходов к определению релевантности

Для анализа свойств предложенного алгоритма и сравнения его с существующими был проведен вычислительный эксперимент в котором параметры нейросети удалялись методами, которые были описаны в разделах 3.1—3.3 и методом Белсли.

В качестве данных использовались три выборки. Выборки Wine [4] и Boston Housing [39] — это реальные данные. Синтетические данные сгенерированы таким образом чтобы параметры сети были мультиколлинеарными. Генерация данных состояла из двух этапов. На первом этапе генерировался вектор параметров $\mathbf{W}_{\text{synthetic}}$:

$$\mathbf{w}_{\text{synthetic}} \sim \mathcal{N}(\mathbf{m}_{\text{synthetic}}, \mathbf{A}_{\text{synthetic}}), \quad (6.26)$$

$$\text{где } \mathbf{m}_{\text{synthetic}} = \begin{bmatrix} 1.0 \\ 0.0025 \\ \dots \\ 0.0025 \end{bmatrix}, \quad \mathbf{A}_{\text{synthetic}} = \begin{bmatrix} 1.0 & 10^{-3} & \dots & 10^{-3} & 10^{-3} \\ 10^{-3} & 1.0 & \dots & 0.95 & 0.95 \\ \dots & \dots & \dots & \dots & \dots \\ 10^{-3} & 0.95 & \dots & 0.95 & 1.0 \end{bmatrix}.$$

На втором этапе генерировалась выборка $\mathfrak{D}_{\text{synthetic}}$:

$$\mathfrak{D}_{\text{synthetic}} = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \sim \mathcal{N}(\mathbf{1}, \mathbf{I}), y_i = x_{i0}, i = 1 \dots 10000\}. \quad (6.27)$$

В приведенном выше векторе параметров $\mathbf{w}_{\text{synthetic}}$ для выборки $\mathfrak{D}_{\text{synthetic}}$, наиболее релевантным является первый параметр, а все остальные параметры являются нерелевантными. Матрица ковариации была выбрана таким образом, чтобы все нерелевантные параметры были зависимы и метод Белсли был максимально эффективен.

Таблица 7: Описание выборок

Выборка	Тип задачи	Размер выборки	Число признаков
Wine	классификация	178	13
Boston Housing	регрессия	506	13
Synthetic data	регрессия	10000	100

Для алгоритмов тренировочная и тестовая выборки составили 80% и 20% соответственно. Критерием качества прореживания служит процент параметров нейросети, удаление которого не влечет значимой потери качества прогноза. Также критерием качества служит устойчивость нейросети к зашумленности данных.

Качеством прогноза R_{cl} модели для задачи классификации является точность прогноза модели:

$$R_{\text{cl}} = \frac{\sum_{(\mathbf{x}, y) \in \mathfrak{D}} [f(\mathbf{x}, \mathbf{w}) = y]}{|\mathfrak{D}|}, \quad (6.28)$$

Качеством прогноза R_{rg} модели для задачи регрессии является среднеквадратическое отклонение результата модели от точного:

$$R_{\text{rg}} = \frac{\sum_{(\mathbf{x}, y) \in \mathfrak{D}} (f(\mathbf{x}, \mathbf{w}) - y)^2}{|\mathfrak{D}|}, \quad (6.29)$$

Wine. Рассмотрим нейронную сеть с 13 нейронами на входе, 13 нейронами в скрытом слое и 3 нейронами на выходе.

На рис. 33 показано как меняется точность прогноза R_{cl} при удалении параметров указанными методами. Из графика видно, что метод оптимального прореживания, вариационный метод и метод Белсли позволяют удалить $\approx 80\%$ параметров и качество всех этих методов падает при удалении $\approx 90\%$ параметров нейросети.

На рис. 34 показаны поверхности изменения уровня шума ответов нейросети при изменении процента удаленных параметров и уровня шума входных данных для разных методов прореживания. На графиках показано, что при удалении параметров нейросети методом Белсли шум меньше, чем при удалении

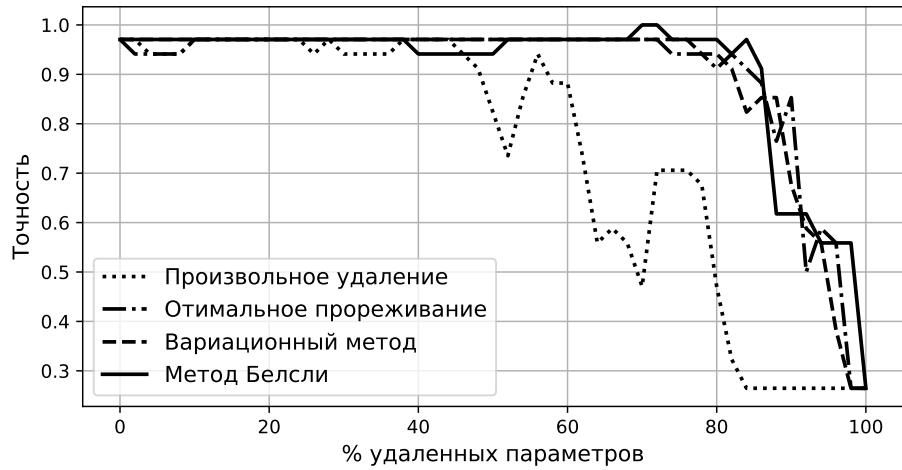


Рис. 33: Качество прогноза при удалении параметров на выборке Wine

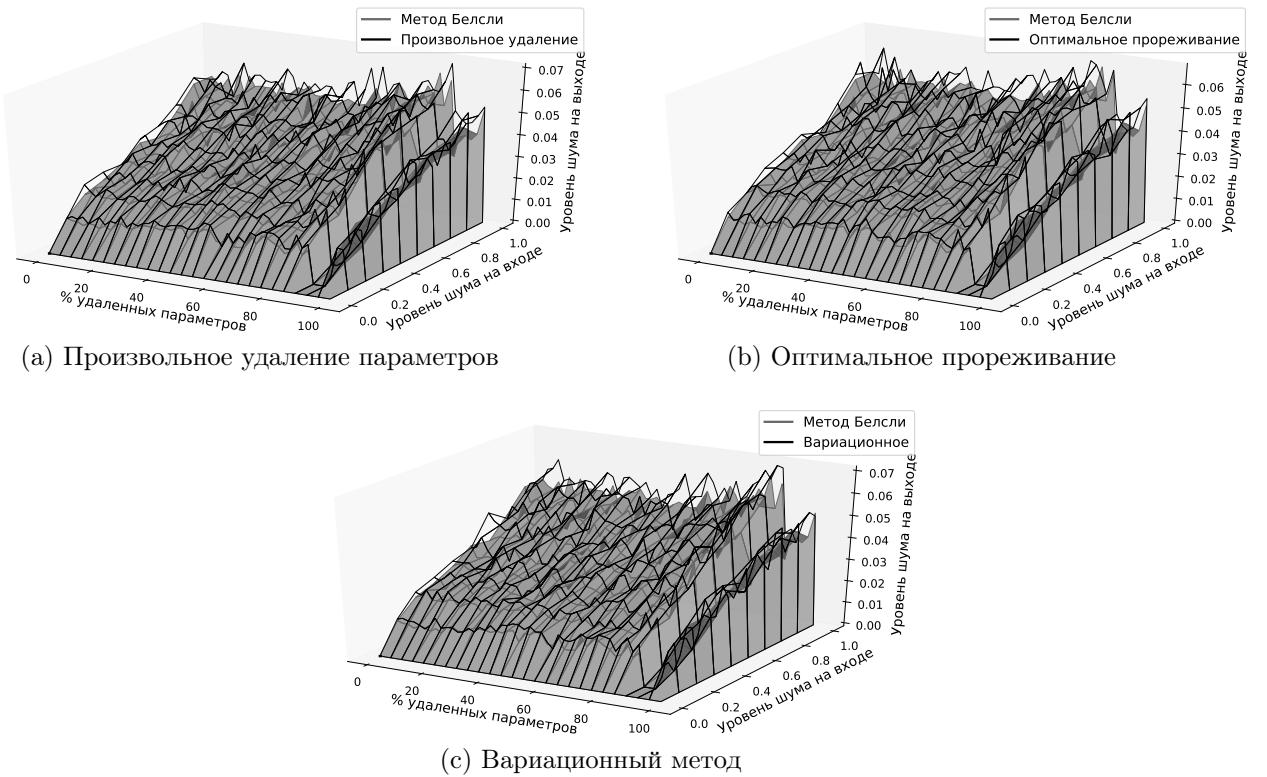


Рис. 34: Влияние шума в начальных данных на шум выхода нейросети на выборке Wine

параметров другими методами, на это указывает то что поверхность которая соответствует методу Белсли ниже других поверхностей.

Boston Housing. Рассмотрим нейронную сеть с 13 нейронами на входе, 39 нейронами в скрытом слое и одним нейроном на выходе.

На рис. 35 показано как меняется среднеквадратическое отклонение прогноза R_{rg} от точного ответа при удалении параметров указанными методами. Гра-

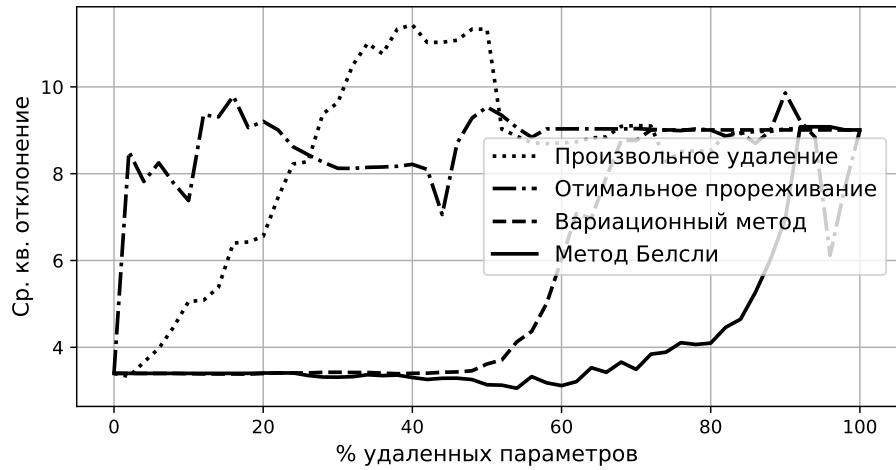


Рис. 35: Качество прогноза при удаление параметров на выборке Boston

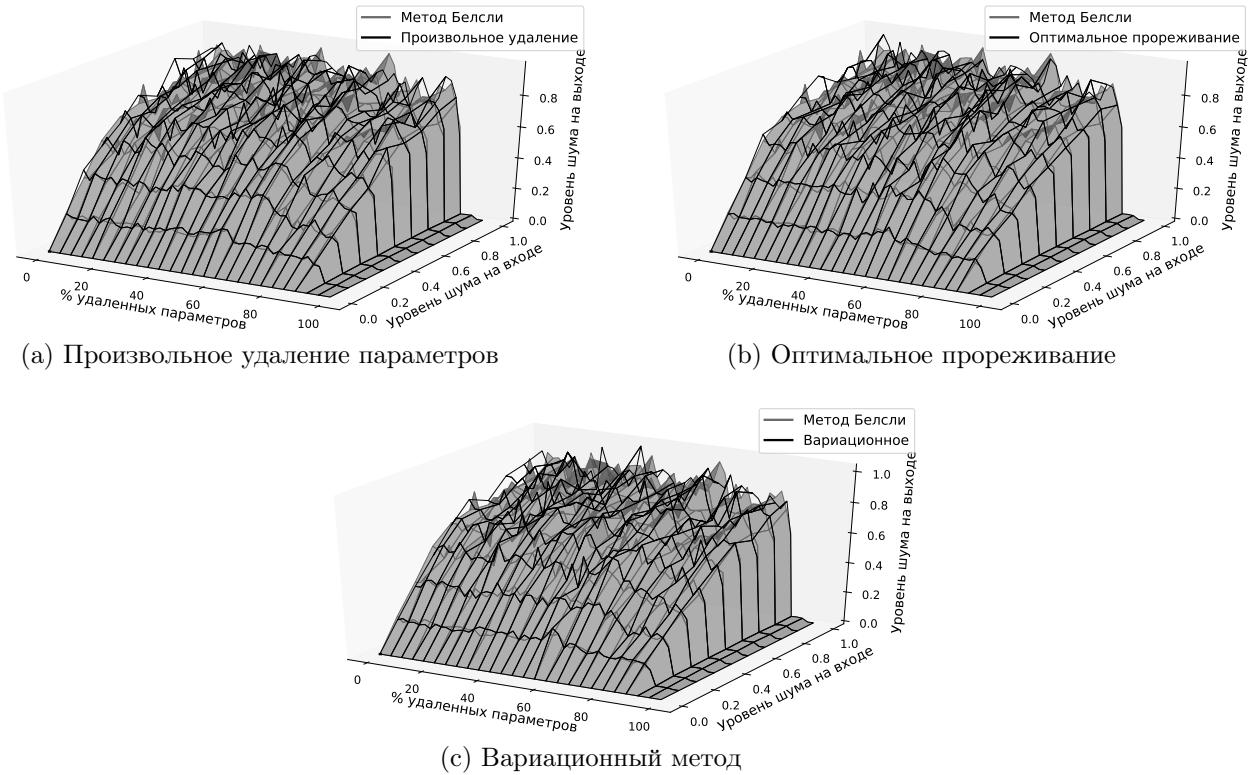


Рис. 36: Влияние шума в начальных данных на шум выхода нейросети на выборке Boston

фик показывает, что метод Белсли является более эффективным, чем другие методы, так-как позволяет удалить больше параметров нейросети без потери качества.

На рис. 36 показаны поверхности изменения уровня шума ответов нейросети при изменении процента удаленных параметров и уровня шума входных данных для разных методов прореживания. График показывает, что уровень шума всех методов одинаковый, так-как поверхности всех методов находятся на одном

уровне.

Синтетические данные. Рассмотрим нейронную сеть с 100 нейронами на входе и одним нейроном на выходе.

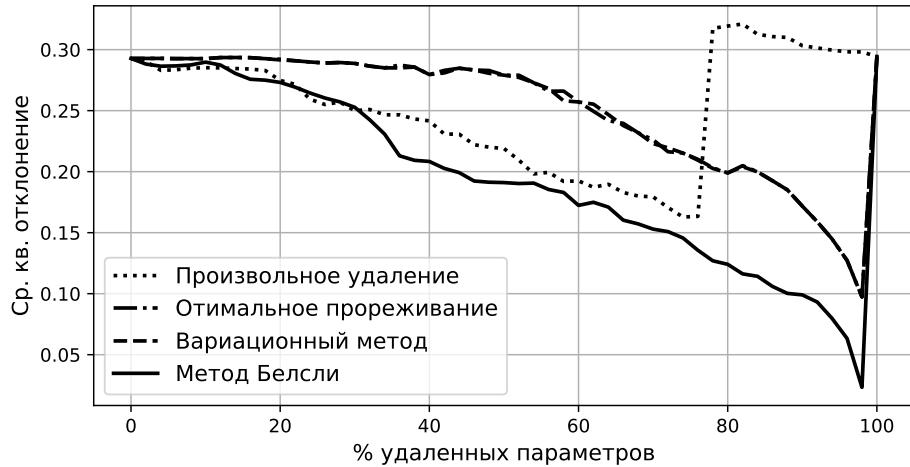


Рис. 37: Качество прогноза при удаление параметров на синтетической выборке

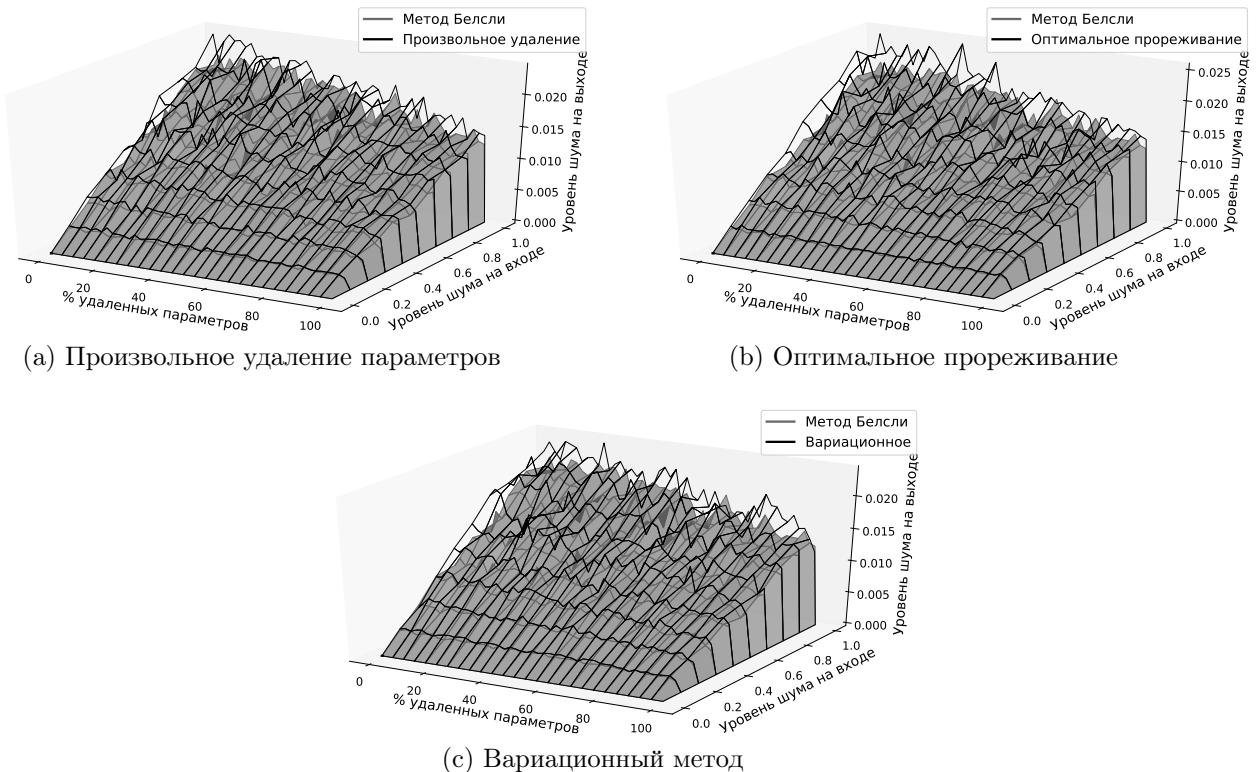


Рис. 38: Влияние шума в начальных данных на шум выхода нейросети на синтетической выборке

На рис. 37 показано как меняется среднеквадратическое отклонение прогноза от R_{rg} точного ответа при удалении параметров указанными методами.

График показывает, что удаление параметров методом Белсли является более эффективным чем другие методы прореживания, так-как качество прогноза нейросети улучшается при удалении шумовых параметров.

На рис. 38 показаны поверхности изменения уровня шума ответов нейросети при изменении процента удаленных параметров и уровня шума входных данных для разных методов прореживания. На графиках показано, что при удалении параметров нейросети методом Белсли шум меньше, чем при удалении параметров другими методами, так-как поверхность которая соответствует методу Белсли ниже других поверхностей.

7 Оптимальный размер выборки для построения линейных моделей

Планирование эксперимента требует оценки минимального размера выборки: количества выполненных измерений набора характеристик, необходимых для построения сформулированных условий. Выбор метода оценки размера выборки зависит от решаемой задачи, которая определяет формулировку статистической гипотезы и статистики для ее проверки. В таблице 8 представлены десять методов оценки размера выборки. Он включает как классический, так и байесовский методы оценки размера выборки.

Классические методы предполагают, что образец соответствует некоторым предварительным условиям, сформулированным ранее. Эти условия сформулированы как статистический критерий (Self et al. 1988, 1992; Shieh 2000; Demidenko 2007). Метод оценки размера выборки, связанный с этим критерием, гарантирует достижение фиксированной статистической мощности $1 - \beta$ со степенью ошибки первого рода, не превышающей установленное значение α . Такой размер выборки называется достаточным.

Однако практическое применение методов оценки размера выборки предполагает, что модель соответствует измеренным данным (Kloek, 1975). Эти модели выбираются в соответствии с постановкой задачи регрессии или классификации. В этой статье представлены обобщенные линейные модели. В статье (Self et al. 1988) описывается метод оценки мощности критерия множителя Лагранжа для коэффициентов обобщенной линейной регрессии, с помощью которого оценивается размер выборки. Недостаток данного метода заключается в том, что, когда альтернативная гипотеза сильно отличается от нулевой гипотезы, оценки максимального правдоподобия для параметров модели и ковариационная матрица, используемые для оценки мощности, не являются асимптотически согласованными в альтернативной гипотезе. Позднее (Self et al. 1992) был предложен подход к оценке мощности и размера выборки, связанной с ней, на основе теста отношения максимального правдоподобия. Этот подход оказался более точным для ряда независимых переменных. Кроме того, в статье (Shieh 2005) был предложен метод оценки мощности для статистики Вальда. В статье (Мотренко и др., 2014) в случае логистической регрессии предлагается использовать метод, использующий кривую ROC-AUC и концепцию сдвига. Классические методы (Self et al. 1988, 1992; Shieh 2000, 2005; Demidenko 2007) имеют ряд ограничений, связанных с практическим применением этих методов. Чтобы оценить размер выборки, необходимо знать дисперсию оценки параметра или, в более общем случае, иметь оценку параметра нецентральности в распределении статистики, используемой, когда альтернативная гипотеза верна. Эти методы не показывают, как получить эти значения. Кроме того, дисперсия оценки и параметр нецентральности не будут получены с определенной дисперсией, влияние которой на результат оценки размера выборки не имеет значения.

Таблица 8: Methods

Метод	Описание	Ссылка
Lagrange multipliers test	Правдоподобие выборки имеет следующий вид: $p(y \mathbf{x}, \mathbf{w}) = \exp(y\theta - b(\theta) + c(y))$. Достаточный размер выборки m^* : $m^* = \frac{\gamma^*}{\gamma_0}$, где γ^* и γ_0 задаются выражениям (7.12) и (7.11).	Self et al. 1988
Likelihood ratio test	Правдоподобие выборки имеет следующий вид: $p(y \mathbf{x}, \mathbf{w}) = \exp\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right)$. Достаточный размер выборки m^* : $m^* = \frac{\gamma^*}{\Delta^*}$, где γ^* и Δ^* задаются выражениями (7.17) и (7.19).	Shieh 2000
Wald statistic	Правдоподобие выборки имеет следующий вид: $p(y \mathbf{x}, \mathbf{w}) = \exp\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right)$. Достаточный размер выборки m^* : $m^* = \frac{\gamma^*}{\delta}$, где γ^* и δ задаются выражением (7.26).	Shieh 2005
Cross-validation	Достаточный размер выборки m^* : $\forall m \geq m^* RS(m) \geq 1 - \varepsilon$, где ε задается экспертом, RS определяется выражением (7.31).	Motrenko et al. 2014
Bootstrap	Достаточный размер выборки m^* : $\forall m \geq m^* \max_i (b_i^m - a_i^m) < l$, где (a_i^m, b_i^m) является квантилью доверительного интервала i -й бутсрап подвыборки размера m	(Qumsiyeh 2013)
Kullback-Leibler	Достаточный размер выборки m^* : $\forall \mathfrak{D}_{\mathcal{B}_1} : \mathfrak{D}_{\mathcal{B}_1} \geq m^* E_{\mathfrak{D}_{\mathcal{B}_2}} D_{KL}(p_1, p_2) \leq \varepsilon$, где $\mathcal{B}_1, \mathcal{B}_2$ удовлетворяет (7.42).	Motrenko et al. 2014
Average posterior variance criterion	Достаточный размер выборки m^* : $\forall m \geq m^* E_{\mathfrak{D}_m} D[\hat{\mathbf{w}} \mathfrak{D}_m] \leq \varepsilon$, где ε достаточно малое значение.	Joseph et al. 1997, 1995
Average coverage criterion	Достаточный размер выборки m^* : $\forall m \geq m^* E_{\mathfrak{D}_m} P\{\mathbf{w} \in A(\mathfrak{D}_m)\} \geq 1 - \alpha$, где α достаточно малое значение.	Joseph et al. 1997, 1995
Average length criterion	Достаточный размер выборки m^* : $\forall m \geq m^* E_{\mathfrak{D}_m} r_m \leq l$, where r_m is described in (7.39)	Joseph et al. 1997, 1995
Utility maximization	Достаточный размер выборки m^* : $m^* = \arg \max_m E_{\mathfrak{D}_m} \int_{\mathbf{w}} u(\mathfrak{D}, \mathbf{w}) p(\mathbf{w} \mathfrak{D}) d\mathbf{w}$, где функция полезности $u(\mathfrak{D}, \mathbf{w})$ задается выражением (7.41).	Lindley 1997

Статистические методы позволяют оценить размер выборки на основе предположений о распределении данных и информации о соответствии между наблюдаемыми значениями и предположениями нулевой гипотезы. Когда размер исследуемой выборки является достаточным или чрезмерным, можно использовать методы, основанные на наблюдении изменения определенной характеристики процедуры построения модели при увеличении размера выборки. В частности, наблюдая за соотношением качества прогнозирования с контрольной выборкой и обучающей выборкой (Мотренко и др., 2014), мы будем определять достаточный размер выборки, который соответствует началу переобучения. В статье (Qumsiyeh 2013) для оценки достаточного размера выборки используется метод ресемплирования выборки используя процедуру бутстррап. Превышение текущего размера выборки проверяется на основе анализа доверительных интервалов оцениваемого параметра. Ширина доверительного интервала с различными значениями объема выборки оценивается с помощью метода бутстррапа. Для этого выборки меньшего размера отбираются заданное количество раз и вычисляется доверительный интервал ошибки при оценке параметра модели. Размер выборки считается достаточным, если ширина доверительного интервала не превышает заранее установленного значения.

Перечисленные выше ограничения статистических методов оценки размера выборки подробно рассматриваются в байесовской процедуре (Lindley 1997; Rubin et al. 1998; Wang et al. 2002), где оценка размера выборки определяется на основе максимизации ожидаемого значение некоторой функции качества (Lindley 1997). Функция качества может включать в себя явные функции распределения параметров и штрафы за увеличение размера выборки. Альтернативой подходам (Wang et al. 2002), основанным на функции качества, является выборка размера выборки путем установления ограничений на определенный критерий качества оценки параметров модели. Примеры таких критериев следующие: критерий средней апостериорной дисперсии (AVPC), критерий средней длины (ALC), критерий среднего покрытия (ACC). Для каждого перечисленного критерия оценка размера выборки определяется как минимальное значение размера выборки, для которого ожидаемое значение выбранного критерия не превышает какого-либо фиксированного порога. В статье (Motrenko et al. 2014) предлагается считать размер выборки достаточным, если расстояние Кульбака-Лейблера между распределениями, оцененными на основе подвыборок такого размера, достаточно мало. Такой подход не требует дальнейшего обобщения в случае нескольких переменных. Кроме того, оценка может производиться как при наличии предположений о распределении данных, так и при их отсутствии. Недостаток этого подхода заключается в том, что количественная оценка может быть получена только при чрезмерно большом размере выборки.

7.1 Постановка задачи определения оптимального размера выборки

Задана выборка размера m :

$$\mathfrak{D}_m = \{\mathbf{x}_i, y_i\}_{i=1}^m, \quad (7.1)$$

где $\mathbf{x}_i \in \mathbb{R}^n$, $y_i \in \mathbb{Y}$. Вектор признаков $\mathbf{x} = [\mathbf{u}, \mathbf{v}]$ соединяет $\mathbf{u}_i \in \mathbb{R}^k$ and $\mathbf{v}_i \in \mathbb{R}^{n-k}$. Выборка \mathfrak{D}_m случайным образом делится на обучающую и тестовую части:

$$\mathfrak{D}_{\mathcal{T}_m} = \{\mathbf{x}_i, y_i\}_{i \in \mathcal{T}_m}, \quad \mathfrak{D}_{\mathcal{L}_m} = \{\mathbf{x}_i, y_i\}_{i \in \mathcal{L}_m}, \quad \mathcal{T}_m \sqcup \mathcal{L}_m = \{1, \dots, m\}. \quad (7.2)$$

Введем параметрическое семейство функций для аппроксимации неизвестного распределения $p(y|\mathbf{x}, \mathfrak{D}_{\mathcal{L}_m})$:

$$\mathfrak{F} = \left\{ f(y, \mathbf{x}, \mathbf{w}) \mid \mathbf{w} \in \mathbb{W}, \int_{y \in \mathbb{Y}, \mathbf{x} \in \mathbb{R}^n} f(y, \mathbf{x}, \mathbf{w}) dy d\mathbf{x} = 1 \right\}. \quad (7.3)$$

Для модели f с вектором параметров \mathbf{w} определим функцию правдоподобия и логарифм функции правдоподобия выборки \mathfrak{D} :

$$L(\mathfrak{D}, \mathbf{w}) = \prod f(y, \mathbf{x}, \mathbf{w}), \quad l(\mathfrak{D}, \mathbf{w}) = \sum \log f(y, \mathbf{x}, \mathbf{w}), \quad (7.4)$$

где $f(y, \mathbf{x}, \mathbf{w})$ является оценкой правдоподобия выборки $\mathfrak{D}_{\mathcal{L}}$ с заданным вектором параметров \mathbf{w} . Используйте принцип максимального правдоподобия для оценки параметров \mathbf{w}

$$\hat{\mathbf{w}} = \arg \max_{\mathbf{w} \in \mathbb{W}} L(\mathfrak{D}_{\mathcal{L}}, \mathbf{w}). \quad (7.5)$$

Информационная матрица Фишера имеет вид:

$$\mathbf{I}(\mathfrak{D}, \mathbf{w}) = -\nabla \nabla^T l(\mathfrak{D}, \mathbf{w}), \quad \mathbf{V} = \mathbf{I}^{-1}(\mathfrak{D}, \mathbf{m}), \quad (7.6)$$

статистические методы и байесовские методы используют информационную матрицу Фишера для оценки размера выборки.

7.2 Обзор методов для определения оптимального размера выборки на основе статистических тестов

Основным преимуществом методов, основанных на статистике, является их способность оценивать достаточный размер выборки при недостаточном наборе выборки. Они позволяют прогнозировать необходимое количество образцов на ранней стадии эксперимента.

Пусть вероятность имеет следующий вид:

$$p(y|\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v) = \exp(y\theta - b(\theta) + c(y)), \quad (7.7)$$

где θ является параметром распределения, который вычисляются с помощью функции связи $\theta = \theta(\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v)$.

Тестируемая гипотеза

$$H_0 : \mathbf{m}_u = \mathbf{m}_u^0, \quad H_1 : \mathbf{m}_u \neq \mathbf{m}_u^0. \quad (7.8)$$

Пусть статистики $S_{m,u}(\mathbf{w}_u, \mathbf{w}_v)$ и $S_{m,v}(\mathbf{w}_u, \mathbf{w}_v)$ являются производными логарифма правдоподобия выборки \mathfrak{D}_m в точках \mathbf{w}_u и \mathbf{w}_v . Рассмотрим $\mathbf{s}_m = S_{m,u}(\mathbf{m}_u^0, \hat{\mathbf{w}}_v^0)$, где $\hat{\mathbf{w}}_v^0$ получается из уравнения

$$S_{m,v}(\mathbf{m}_u^0, \mathbf{w}_v) = 0. \quad (7.9)$$

Статистика Лагранджа равняется

$$LM = \mathbf{s}_m^\top \mathbf{Q}_m^{-1} \mathbf{s}_m. \quad (7.10)$$

где \mathbf{Q}_m ковариационная матрица вектора \mathbf{s}_m .

В случае истинности гипотезы H_0 статистика LM асимптотически имеет распределения $\chi^2(k)$. В работе (Self et al. 1988) показано, что при альтернативной гипотезе H_1 статистика LM асимптотически имеет распределения $\chi^2(k, \gamma)$, где γ является параметром не центральности

$$\gamma = \boldsymbol{\xi}_m^\top \boldsymbol{\Sigma}_m^{-1} \boldsymbol{\xi}_m = m \boldsymbol{\xi}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\xi} = m\gamma^0, \quad (7.11)$$

где $\boldsymbol{\xi}_m$ и $\boldsymbol{\Sigma}_m$ матрицы математического ожидания и ковариации \mathbf{s}_m . Обозначим $\boldsymbol{\xi}_1 = \boldsymbol{\xi}$, $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}$.

Альтернативный метод получения γ включает условия на уровне значимости α и вероятность ошибки II рода β :

$$\gamma^* : \chi_{k,1-\alpha}^2 = \chi_{k,\beta}^2(\gamma). \quad (7.12)$$

Используя (7.11) и (7.12) получаем

$$m^* = \frac{\gamma^*}{\gamma^0}. \quad (7.13)$$

Это достаточный минимальный размер выборки, чтобы различить вектор \mathbf{m}_u от \mathbf{m}_u^0 .

Пусть правдоподобие выборки задается выражением

$$p(y|\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v) = \exp\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right), \quad (7.14)$$

где θ является параметром распределения, который вычисляются с помощью функции связи $\theta = \theta(\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v)$.

Проверяемая гипотеза:

$$H_0 : \mathbf{m}_u = \mathbf{m}_u^0, \quad H_1 : \mathbf{m}_u \neq \mathbf{m}_u^0. \quad (7.15)$$

Введем логарифм статистики отношения правдоподобий:

$$LR = 2\left(l(\mathfrak{D}, \hat{\mathbf{w}}) - l(\mathfrak{D}, \hat{\mathbf{w}}^0)\right), \quad (7.16)$$

где $\hat{\mathbf{w}} = [\hat{\mathbf{w}}_u, \hat{\mathbf{w}}_v]$ является вектором, который максимизирует правдоподобие (7.14), $\hat{\mathbf{w}}^0 = [\mathbf{m}_u^0, \hat{\mathbf{w}}_v^0]$ является вектором, который максимизирует правдоподобие (7.14) с фиксируемым подвектором параметров \mathbf{m}_u^0 .

В случае истинности гипотезы H_0 статистика LR асимптотически имеет распределения $\chi^2(k)$. В работе (Shieh 2000) показано, что при альтернативной гипотезе H_1 статистика LR асимптотически имеет распределения $\chi^2(k, \gamma)$, где γ является параметром не центральности

$$\gamma = m\Delta^*, \quad \Delta^* = E [2a^{-1}(\phi) \{ (\theta - \theta^*) \nabla b(\theta) - b(\theta) + b(\theta^*) \}], \quad (7.17)$$

где параметры θ и θ^* рассчитываются используя параметры $\mathbf{w} = [\mathbf{w}_u, \mathbf{w}_v]$ и $\mathbf{w}^* = [\mathbf{w}_u^0, \mathbf{w}_v^*]$. Параметры \mathbf{w}_v^* вычисляются на основе решения уравнения:

$$\lim_{m \rightarrow \infty} m^{-1} E \left(\frac{\partial l(\mathfrak{D}, [\mathbf{m}_u^0, \mathbf{w}_v])}{\partial \mathbf{w}_v} \right) = 0. \quad (7.18)$$

Тогда с учетом α и β достаточный размер выборки m^* вычисляется следующим образом:

$$m^* = \frac{\gamma^*}{\Delta^*}, \quad \gamma^* : \chi_{k,1-\alpha}^2 = \chi_{k,\beta}^2(\gamma), \quad (7.19)$$

где $\chi_{k,1-\alpha}^2$, $\chi_{k,\beta}^2(\gamma^*)$ квантили распределений χ_k^2 and $\chi_k^2(\gamma^*)$.

Пусть правдоподобие выборки задается следующим образом:

$$p(y|\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v) = \exp \left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right), \quad (7.20)$$

где θ является параметром распределения, который вычисляются с помощью функции связи $\theta = \theta(\mathbf{u}, \mathbf{v}, \mathbf{w}_u, \mathbf{w}_v)$.

Тестируемая гипотеза:

$$H_0 : \mathbf{m}_u = \mathbf{m}_u^0, \quad H_1 : \mathbf{m}_u \neq \mathbf{m}_u^0. \quad (7.21)$$

Тест Вальда для гипотезы:

$$W = (\hat{\mathbf{w}}_u - \mathbf{m}_u^0)^T \hat{\mathbf{V}}_u^{-1} (\hat{\mathbf{w}}_u - \mathbf{m}_u^0), \quad (7.22)$$

где $\hat{\mathbf{w}} = [\hat{\mathbf{w}}_u, \hat{\mathbf{w}}_v]$ вектор параметров, который максимизирует правдоподобие выборки (7.20), где матрица $\hat{\mathbf{V}}_u$ задается в выражении (7.6).

В случае истинности гипотезы H_0 статистика Вальда W асимптотически имеет распределение χ^2 . В работе (Shieh 2005) показано, что в случае истинности альтернативной гипотезы H_1 статистика Вальда W асимптотически имеет распределение $\chi^2(k, \gamma)$ с параметром не центральности γ :

$$\gamma = m\delta, \quad \delta = (\hat{\mathbf{w}}_u - \mathbf{m}_u^0)^T \Sigma_u^{-1} (\hat{\mathbf{w}}_u - \mathbf{m}_u^0), \quad \Sigma_u = m\hat{\mathbf{V}}_u. \quad (7.23)$$

Использую заданный уровень значимости α и заданную ошибку второго рода β определим оптимальный размер выборки:

$$m^* = \frac{\gamma^*}{\delta}, \quad \gamma^* : \chi_{k,1-\alpha^*}^2 = \chi_{k,\beta}^2(\gamma), \quad (7.24)$$

где $\chi_{k,1-\alpha^*}^2$, $\chi_{k,\beta}^2(\gamma^*)$ квантили распределения, а параметр α^* это поправка на уровни значимости:

$$\alpha^* = P(\boldsymbol{\xi}^T \Sigma^{*-1} \boldsymbol{\xi} > \chi_{k,1-\alpha}^2), \quad \Sigma^* = \mathbf{I}^{-1}(\mathfrak{D}, \mathbf{w}^*), \quad (7.25)$$

где $\mathbf{w}^* = [\mathbf{m}_u^0, \mathbf{w}_v^*]$ является решением уравнения:

$$\lim_{m \rightarrow \infty} m^{-1} E \left(\frac{\partial l(\mathfrak{D}, [\mathbf{m}_u^0, \mathbf{w}_v])}{\partial \mathbf{w}_v} \right) = 0. \quad (7.26)$$

7.3 Эвристические методы определения достаточного размера выборки

В методе, основанном на эвристике, используются популярные статистические эвристики, такие как бутстреп, перекрестная проверка и задание функции полезности. Введем набор индексов \mathcal{A} для параметров логистической регрессии \mathbf{w} . Тестируемая гипотеза:

$$H_0 : j \notin \mathcal{A} (\mathbf{w}_j = 0), \quad H_1 : j \in \mathcal{A}^* (\mathbf{w}_j \neq 0), \quad (7.27)$$

где \mathbf{w}_j является j -м элементом вектора \mathbf{w} . Установим параметр отступа c_0 для задачи логистической регрессии:

$$H_0 : 1 - c_0 = p_0, \quad H_1 : 1 - c_0 = p_1, \quad (7.28)$$

где c_0 оптимальное решение, когда исключен j -й элемент вектора. Используя статистику

$$Z = \frac{\hat{p} - p_0}{\sqrt{p_0(1 - p_0)}} \sqrt{m}, \quad \hat{p} = \frac{1}{m} \sum_{i=1}^m y_i. \quad (7.29)$$

В случае истинности нулевой гипотезы H_0 статистика Z асимптотически имеет распределение $\mathcal{N}(0, 1)$. В случае истинности альтернативной гипотезы H_1 статистика Z асимптотически имеет распределение $\mathcal{N}\left(p_1 - p_0, \sqrt{\frac{p_1(1-p_1)}{p_0(1-p_0)}}\right)$.

Достаточный объем выборки задается следующим образом:

$$m^* = \frac{p_0(1 - p_0) \left(Z_{1-\alpha/2} + Z_{1-\beta} \sqrt{\frac{p_1(1-p_1)}{p_0(1-p_0)}} \right)^2}{(p_1 - p_0)^2}, \quad (7.30)$$

где $Z_{1-\alpha/2}$ и $Z_{1-\beta}$ являются квантилями распределения $\mathcal{N}(0, 1)$.

Мы не будем рассматривать этот метод далее, поскольку его можно использовать только для задачи логистической регрессии.

Определите критерий переобучения как

$$RS(m) = \ln \frac{L(\mathfrak{D}_{\mathcal{L}(m)}, \hat{\mathbf{w}})}{L(\mathfrak{D}_{\mathcal{T}(m)}, \hat{\mathbf{w}})}, \quad \frac{|\mathcal{T}(m)|}{|\mathcal{L}(m)|} = \text{const} \leq 0.5. \quad (7.31)$$

Заметим, что

$$\lim_{m \rightarrow \infty} RS(m) \rightarrow 0. \quad (7.32)$$

Достаточный размер выборки m^* определяется согласно условию:

$$m^* : \forall m \geq m^* \mathbb{E}_{\mathfrak{D}_m} RS(m) \leq \varepsilon, \quad (7.33)$$

где ε некоторый параметр задан экспертизой.

Этот метод предполагает, что длины доверительных интервалов квантиля не превышают некоторого фиксированного значения l . Для некоторого размера выборки m вычисляются квантильные доверительные интервалы $(a_1^m, b_1^m), (a_2^m, b_2^m)$, с уровнем значимости α с использованием начальной загрузки для каждого параметра модели. Достаточный размер выборки задается следующим образом:

$$m^* : \forall m \geq m^* \max_i (b_i^m - a_i^m) < l. \quad (7.34)$$

Обратите внимание, что этот метод является покоординатным. Следовательно, для повышения точности прогноза требуется значительное увеличение размера выборки.

7.4 Байесовский подход к определению оптимального размера выборки

Байесовские методы оценки размера выборки основаны на ограничении некоторых характеристик модели. Для анализа эффективности определяется функция размера выборки. Увеличение этой функции интерпретируется как снижение эффективности модели. Размер выборки m^* выбирается таким, чтобы исследуемая функция была меньше некоторого порогового значения ε .

Average posterior variance criterion. Размер выборки m^* определяется условием:

$$\forall m \geq m^* \mathbb{E}_{\mathfrak{D}_m} D[\hat{\mathbf{w}} | \mathfrak{D}_m] \leq l. \quad (7.35)$$

где l некоторый заданный экспертизно параметр, который количественно определяет неопределенность оценки параметра.

Average coverage criterion. Обозначим через $A(\mathfrak{D}) \subset \mathbb{R}^n$ некоторый набор параметров модели \mathbf{w} :

$$A(\mathfrak{D}) = \{\mathbf{w} : \|\mathbf{w} - \hat{\mathbf{w}}\| \leq l\}, \quad (7.36)$$

где l — некоторый фиксированный радиус шара. Размер выборки m^* определяется критерием среднего покрытия:

$$\forall m \geq m^* \mathbb{E}_{\mathfrak{D}_m} P\{\mathbf{w} \in A(\mathfrak{D}_m)\} \geq 1 - \alpha, \quad (7.37)$$

где α некоторый параметр заданный экспертизно.

Average length criterion. Определим функцию $A(\mathfrak{D})$ следующим образом:

$$P(A(\mathfrak{D})) = 1 - \alpha. \quad (7.38)$$

Оценки критерия средней длины m^* as in (7.37):

$$\forall m \geq m^* \mathbb{E}_{\mathfrak{D}_m} r_m \leq l, \quad (7.39)$$

где r_m является радиусом шара $A(\mathfrak{D}_m)$.

Методы следующего класса максимизируют ожидание некоторой функции полезности $u(\mathfrak{D}, \mathbf{w})$ по размеру выборки:

$$m^* = \arg \max_m \mathbb{E}_{\mathfrak{D}_m} \int_{\mathbf{w}} u(\mathfrak{D}_m, \mathbf{w}) p(\mathbf{w} | \mathfrak{D}_m) d\mathbf{w}, \quad (7.40)$$

где функция полезности $u(\mathfrak{D}, \mathbf{w})$ задается в виде:

$$u(\mathfrak{D}_m, \mathbf{w}) = l(\mathfrak{D}_m, \mathbf{w}) - cm, \quad (7.41)$$

где c функция штрафов для каждого элемента в наборе выборки.

Будем говорить, что индексы $\mathcal{B}_1, \mathcal{B}_2 \subset \{1, \dots, m\}$ по соседству, если

$$|\mathcal{B}_1 \Delta \mathcal{B}_2| = 1. \quad (7.42)$$

Таким образом, \mathcal{B}_2 можно преобразовать в \mathcal{B}_1 путем удаления, замены или добавления одного элемента. В работе (Motrenko et al. 2014) показано, что если размер набора выборок $\mathfrak{D}_{\mathcal{B}_1}$ достаточно велик, чем параметры модели $\hat{\mathbf{w}}_1$, оптимизированные с помощью $\mathfrak{D}_{\mathcal{B}_1}$, должны находиться в окрестности параметров модели $\hat{\mathbf{w}}_2$, которые оптимизированы с помощью $\mathfrak{D}_{\mathcal{B}_2}$.

Используйте дивергенцию Кульбака-Лейблера в качестве функции близости между распределениями параметров модели, оптимизированных с помощью $\mathfrak{D}_{\mathcal{B}_1}$ и $\mathfrak{D}_{\mathcal{B}_2}$:

$$D_{KL}(p_1, p_2) = \int_{\mathbf{w} \in \mathbb{W}} p_1(\mathbf{w}) \log \frac{p_1(\mathbf{w})}{p_2(\mathbf{w})} d\mathbf{w}, \quad (7.43)$$

где p_1 and p_2 апостериорные вероятности вектора параметров \mathbf{w} рассчитаны на подвыборках $\mathfrak{D}_{\mathcal{B}_1}$ и $\mathfrak{D}_{\mathcal{B}_2}$ соответственно. Также предполагается, что $\mathfrak{D}_{\mathcal{B}_1}$ и $\mathfrak{D}_{\mathcal{B}_2}$ находятся по соседству. Достаточный размер выборки m^* оценивается следующим образом:

$$\forall \mathfrak{D}_{\mathcal{B}_1} : |\mathfrak{D}_{\mathcal{B}_1}| \geq m^* \mathbb{E}_{\mathfrak{D}_{\mathcal{B}_2}} D_{KL}(p_1, p_2) \leq \varepsilon. \quad (7.44)$$

7.5 Вычислительный эксперимент по анализу разных подходов к определению оптимального размера выборки

Таблица 9: Описание выборок для эксперимента

Выборка	Задача	Число признаков	Размер выборки
Boston Housing	regression	14	506
Diabets	regression	20	576
Forest Fires	regression	13	517
Servo	regression	4	167
NBA	classification	12	2235

Перечисленные выше методы оценки размера выборки реализованы в пакете Python. Этот пакет можно использовать как для прогнозирования достаточного размера выборки на ранней стадии эксперимента, так и для ретроспективного анализа достаточности набора выборки. Этот пакет также включает некоторые вспомогательные функции для исследования размера выборки и визуализации результатов, как показано на рисунках. Исходный код для службы оценки размера выборки доступен по адресу <https://github.com/andriygav/SampleSizeLib>. Код эксперимента и наборы данных, использованные в статье, доступны по адресу <https://github.com/ttgadaev/SampleSizeEstimation>.

Этот эксперимент был проведен для анализа свойств методов оценки размера выборки. Эксперимент состоит из трех частей. Во время первой части получаются оценки размера для всех наборов выборок при фиксированных идентичных параметрах методов. Во второй части исследуется зависимость достаточного размера выборки от имеющегося размера выборки. В третьей части исследуется поведение методов в зависимости от изменения параметров методов. В качестве данных использовались пять наборов образцов, описанных в таблице 9. Девять методов в строках таблицы 10 показывают оценки размера выборки для соответствующих наборов данных.

Таблица 10: Эксперимент по оценке размера выборки для различных наборов выборок

Методы и наборы данных	Boston Housing	Diabetes	Forest Fires	Servo	NBA
Lagrange Multipliers Test	18	25	44	38	218
Likelihood Ratio Test	17	25	43	18	110
Wald Test	66	51	46	76	200
Cross Validation	178	441	172	120	–
Bootstrap	113	117	86	60	405
APVC	98	167	351	20	–
ACC	228	441	346	65	–
ALC	98	267	516	25	–
Utility Function	148	172	206	105	925

Эта часть вычислительного эксперимента показывает, как разные методы работают с разными наборами данных. В эксперименте используются следующие наборы данных: Boston Housing (Harrison et al. 1978), Diabetes, Forest Fires, Servo (Quinlan 1992), NBA. Результат представлен в таблице 10. Символ “–” в таблице означает, что данных для прогноза недостаточно.

Each method was provided with the whole sample at the start was performed. The parameters of each method for all samples are registered and described in the Table 11. Since the Lagrange, Likelihood Ration and Wald tests are asymptotic equivalent the parameters of these methods were set identically. The parameters of the Average Coverage and Average Length methods were set identically as well.

Параметры каждого метода для всех выборок описаны в таблице 11. Поскольку критерии Лагранжа, отношения правдоподобия и Вальда асимптотически эквивалентны, параметры этих методов задавались одинаково. Параметры методов «Average Coverage» и «Average Length» также были заданы идентично.

The computational experiment was conducted to analyse the described methods. The sample set is the Boston Housing Dataset. Having a full sample set, fix some sample size m and generate series of bootstrap subsamples of size m from the initial sample set. For different values of m compute m^* , average them and calculate standard deviation.

Вычислительный эксперимент проводился для анализа описанных методов.

Таблица 11: Перечень параметров для разных методов оценки объема выборки

Method	GLM parameters	l	ε	α	β
Lagrange Multipliers Test	\mathbf{w}_u^0	—	0.2	0.05	0.2
Likelihood Ratio Test	\mathbf{w}_u^0	—	0.2	0.05	0.2
Wald Test	\mathbf{w}_u^0	—	0.2	0.05	0.2
Cross Validation	—	—	0.05	—	—
Bootstrap	—	0.5	—	0.05	—
APVC	—	0.5	—	—	—
ACC	—	0.25	—	0.05	—
ALC	—	0.5	—	0.05	—
Utility function	—	—	0.005	—	—

Имея полный набор выборок, фиксируется некоторый размер выборки t и методом бутсрап генерируется серия подвыборок размером t . Для разных значений t вычисляется t^* .

На рисунке 39 показана зависимость статических значений каждого метода для каждого набора данных с фиксированным размером выборки t . Пороговые значения для каждого метода устанавливаются экспертизно, что позволяет контролировать различные функции набора данных. Рисунок 39 демонстрирует адекватность различных определений достаточности размера выборки. Все представленные функции монотонны и все они асимптотически стремятся к константе. На рисунке 40 показаны результаты методов на выборках разного размера. Он показывает, как методы различаются по дисперсии вычисленно-го t^* и поведению в случае небольшого набора выборок. Методы сходятся, и результат становится независимым от доступного размера выборки t .

Малое значение дисперсия интерпретируется как вычислительная стабильность рассмотренных методов. Некоторые методы не могут дать оценку достаточного размера выборки, если у них нет соответствующего размера выборки. Это означает, что они не эффективны с точки зрения прогноза, но могут быть использованы для ретроспективы и анализа уже проведенного эксперимента.

В эксперименте также рассмотрена зависимость между оценкой размера выборки с помощью рассмотренных методов и доступного объема данных. Константа, достигаемая диаграммой зависимости t^* от t , является прогнозом метода оптимального размера выборки. Если эта константа меньше t там, где это показано на диаграмме, то метод прогнозирует оптимальный размер выборки до его получения. Таким свойством обладают только методы Лагранжа, Вальда и метод отношения правдоподобия.

Исследуется изменение оценки размера выборки в зависимости от изменения определенных гиперпараметров для байесовских методов, а также методов,

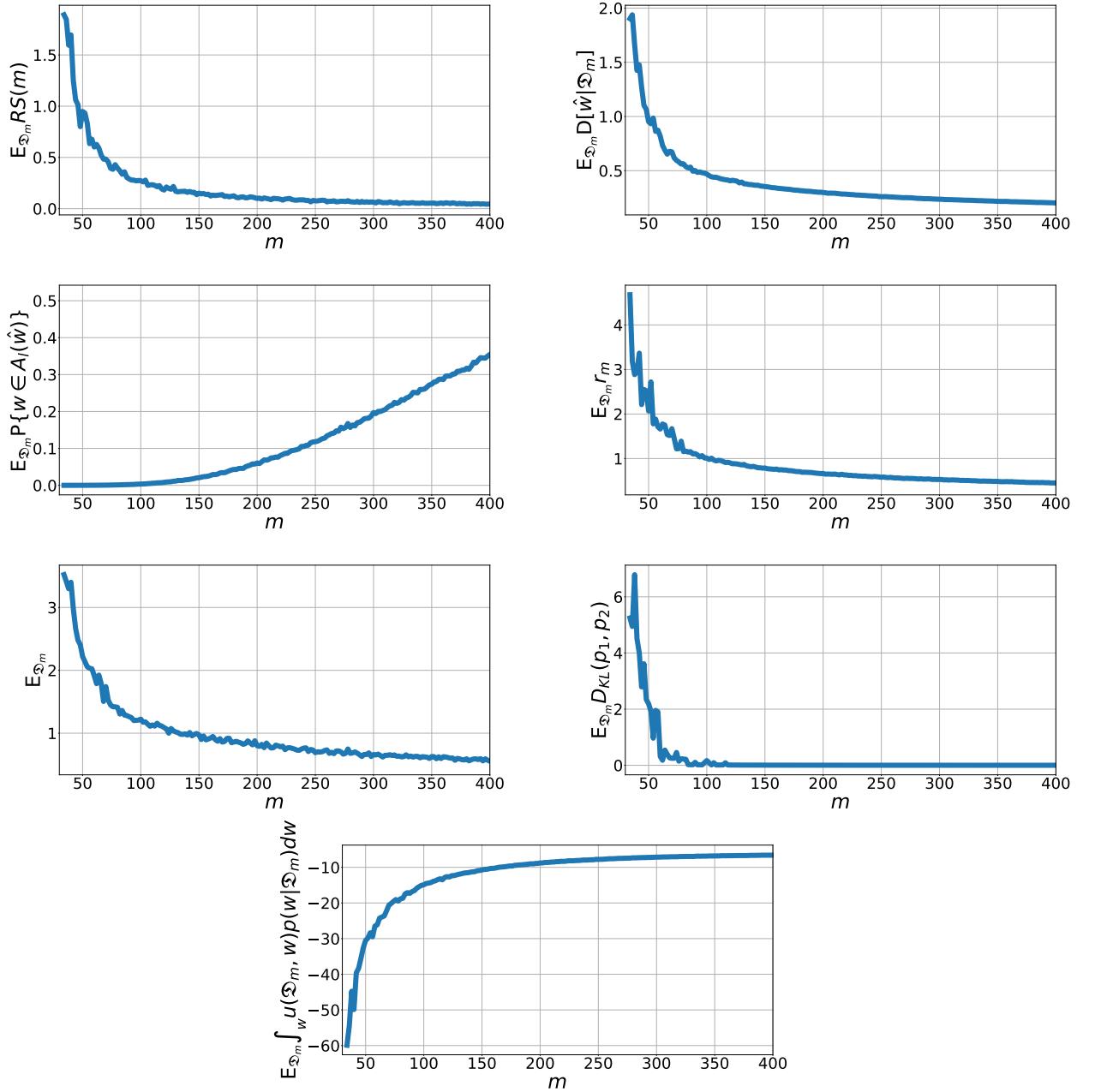


Рис. 39: Поведение основных скалярных функций методов в зависимости от доступного объема выборки

основанных на перекрестной проверке и бутстрэпе. Чтобы проанализировать поведение методов, рассмотрена выборка Boston Housing, остальные выборки имеют такую же тенденцию. Байесовские методы, а также методы, основанные на перекрестной проверке и начальной загрузке, работают на основе определенного решающего правила для определенной выборочной функции. На рисунке 39 показана зависимость этих функций от размера подвыборки. Как показано на рисунке 39, эти функции монотонно убывают или увеличиваются. Тип поведения функции зависит от метода. Изменяя ограничения, установленные экспертом, можно изменить размер выборки, который будет соответствовать этим ограничениям.

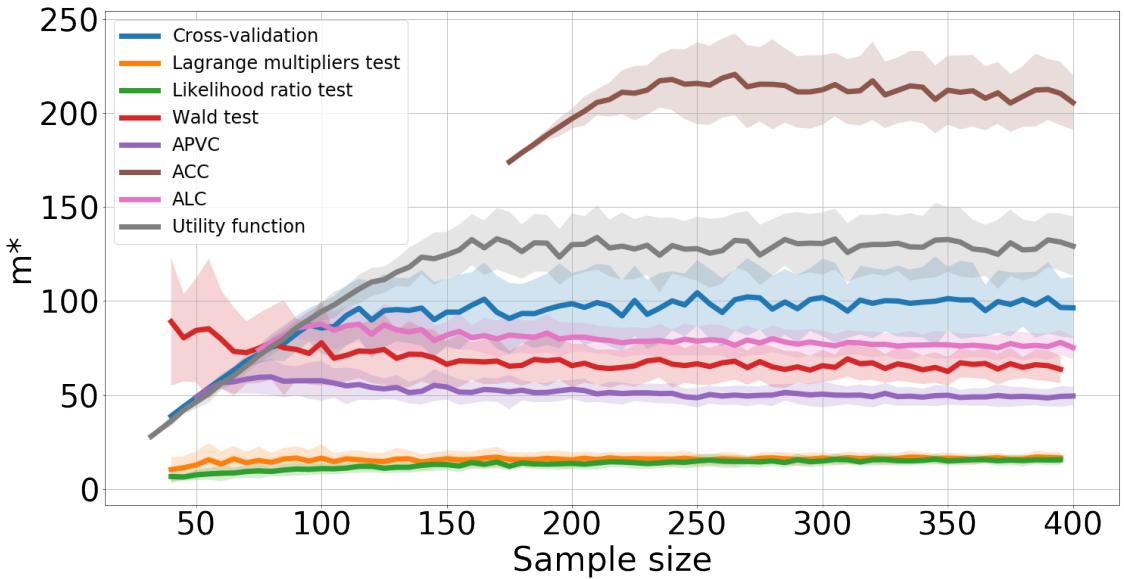


Рис. 40: Поведение методов в зависимости от доступного размера выборки

8 Аппроксимация кривых второго порядка при помощи обучения с экспертом

Построение интерпретируемых моделей в машинном обучении (Ribeiro et al. 2016) является одной из ключевых задач в машинном обучении. Современные решения проблемы классификации изображений на основе сетей глубокого обучения ResNet, VGG, Intercept (Kaiming et al., 2016) являются плохо интерпретируемыми моделями. В работах (Han et al. 2020; Akhtar et al. 2018) показано, что сети глубокого обучения чувствительны даже к небольшому шуму в данных, что связано с их неинтерпретируемостью.

В этом разделе мы предлагаем метод *обучения с экспертом*. Этот метод предполагает использование предметных знаний экспертов для улучшения качества аппроксимации, а также для получения интерпретируемых моделей машинного обучения. Предметные знания экспертов об образце будут называться *экспертная информация*. Предполагается, что использование экспертной информации позволяет аппроксимировать выборку простыми интерпретируемыми моделями, такими как линейные модели. Методы машинного обучения, которые учитывают экспертные знания при построении моделей, называются *экспертным обучением*.

В статье решается задача аппроксимации кривых второго порядка на контурном изображении. Кривые второго порядка выбираются для анализа, так как они легко описываются линейными моделями. В этом случае эти фигуры необходимо восстанавливать в таких прикладных задачах, как задача распознавания радужной оболочки глаза (Matveev 2010; Matveev et al. 2014; Bowyer et al. 2010), задача описания трека частицы в адронный коллайдер (Salamani et al. 2018). Экспертная информация о кривой второго порядка позволяет отображать точки на плоскости в новое описание объекта, где каждая кривая аппрок-

симируется одной линейной моделью. Модель, которая аппроксимирует одну кривую, называется *локальной моделью*. Для аппроксимации всего контурного изображения необходимо аппроксимировать несколько кривых второго порядка с помощью нескольких локальных моделей. В этой статье вводятся следующие ограничения на изображения: а) изображение состоит только из кривых второго порядка; б) изображение аппроксимируется небольшим количеством кривых второго порядка; в) количество и тип кривых на изображении известны.

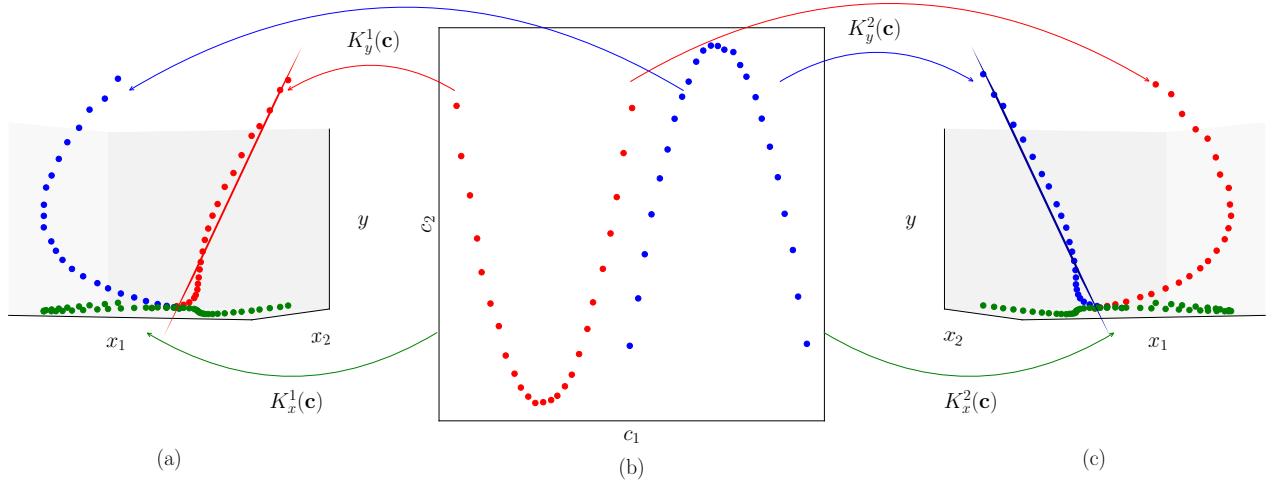


Рис. 41: Пример: а) экспертная информация первого эксперта; б) исходные данные; в) экспертная информация второго эксперта

На рисунке 41 показан пример кривых второго порядка, а также экспертная информация о кривых. На рисунке 41.а показана экспертная информация первого эксперта. Используя эту информацию, первая кривая аппроксимируется линейной моделью, а вторая кривая - шумом. На рисунке 41.б показана экспертная информация второго эксперта. Используя эту информацию, вторая кривая аппроксимируется линейной моделью, а первая кривая представляет собой шум.

При аппроксимации нескольких кривых на одном контурном изображении строится многомодель. Примером нескольких моделей является случайный лес (Chen et al. 2012), бустинг деревьев (Chen et al. 2016), смесь экспертов (Yuksel et al. 2012). В данной работе смесь экспертов рассматривается как мульти модель. Смесь экспертов - это мульти модель, которая линейно взвешивает локальные модели, которые аппроксимируют часть выборки. Значения весовых коэффициентов зависят от объекта, для которого делается прогноз. Для решения проблемы смеси экспертов используется вариационный EM-алгоритм (Dempster et al. 1997; Bishop 2010; Peng et al. 1996). Смесь экспертов имеет множество применений в ряде приложений. В работе (Estabrooks et al. 2001) решается задача классификации текстов. В работах (Cheung et al. 1995; Weigend et al. 2000; Cao 2003; Mossavat et al. 2010; Sminchisescu C et al. 2007; Tuerk 2001; Yumlu et al. 2003), используется смесь экспертов для прогнозировать временные ряды для распознавания речи, повседневной деятельности человека и прогнозирования

стоимости ценных бумаг. В работе (Ebrahimpour et al. 2009) смесь экспертов рассмотрена для решения задачи распознавания рукописных чисел на изображениях.

В качестве примера рассматривается задача аппроксимации изображения радужной оболочки глаза. На рисунке 42а показан пример изображения, которое необходимо аппроксимировать. Рассматриваем обработанное изображение, которое дано в виде схемы, пример такого изображения показан на рисунке 42б. На рисунке 42б показаны две модели окружностей, которые аппроксимируют радужную оболочку глаза. Окружности - простой пример кривой второго порядка.

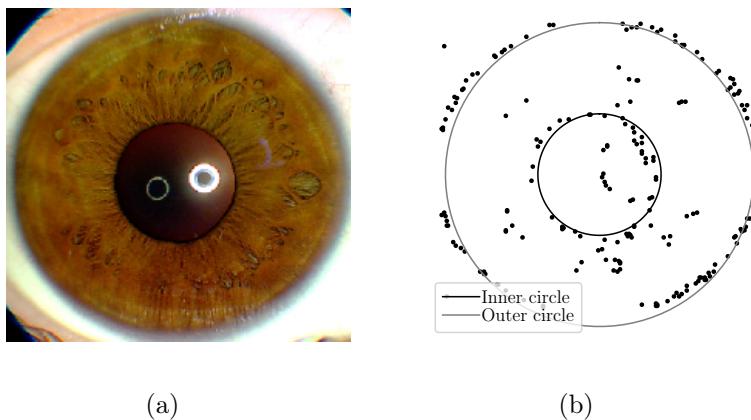


Рис. 42: Пример изображения радужной оболочки глаза и ее контурное изображение: а) изображение радужной оболочки глаза; б) контурное изображение радужной оболочки и аппроксимация заданного изображения окружностей

Для задачи аппроксимации радужной оболочки глаза используется следующая экспертная информация: радужная оболочка глаза аппроксимируется двумя концентрическими окружностями. Экспертная информация используется для построения описания характеристик точек на плоскости, а также для построения функции оптимизации. Часть функции ошибок для оптимизации, использующая экспертную информацию, называется регуляризатором. Таким образом, информация о том, что изображение окружностей задается описанием признака, и информация о том, что концентрические окружности задаются с помощью специального регуляризатора.

В вычислительном эксперименте качество аппроксимации контурного изображения анализируется в зависимости от заданной экспертной информации и от уровня шума в синтетически сгенерированных данных. Анализ качества аппроксимации диафрагмы проводится в зависимости от количества экспертной информации, которая использовалась при построении модели. Обратите внимание, что каждое приближенное изображение представляет собой отдельный набор точек, которые необходимо приблизить.

8.1 Постановка задачи поиска параметров кривых второго порядка

Рассматривается бинарное изображение:

$$\mathbf{M} \in \{0, 1\}^{m_1 \times m_2}, \quad (8.1)$$

где 1 соответствует черной точке изображения, а 0 соответствует белой точке фона.

Из изображения \mathbf{M} строится выборка \mathbf{C} , элементами которой являются координаты (x_i, y_i) черных точек:

$$\mathbf{C} \in \mathbb{R}^{N \times 2}. \quad (8.2)$$

Эксперт предполагает, что изображение состоит из кривой второго порядка Ω . Пусть для набора точек $\mathbf{C} \in \mathbb{R}^{N \times 2}$, образующих кривую Ω , дана экспертная информация о фигуре $E(\Omega)$. Множество $E(\Omega)$ состоит из формы Ω , ожидаемой экспертом, и множества ее допустимых преобразований.

На основе экспертного описания введем отображения в новую задачу аппроксимации:

$$K_x(E(\Omega)) : \mathbb{R}^2 \rightarrow \mathbb{R}^n, \quad K_y(E(\Omega)) : \mathbb{R}^2 \rightarrow \mathbb{R}, \quad (8.3)$$

где K_x отображение объектов с признаковым описанием объектов, n число признаков, а K_y отображение в пространство целевых переменных для объекта. Применение отображений K_x, K_y для всех элементов выборки \mathbf{C} :

$$K_x(E(\Omega), \mathbf{c}) = \mathbf{x}, \quad K_y(E(\Omega), \mathbf{c}) = y, \quad (8.4)$$

где $\mathbf{c} = (x_i, y_i)$ точки из выборки \mathbf{C} .

Применив отображения (8.4) к точкам \mathbf{C} получаем выборку

$$\mathfrak{D} = \{(\mathbf{x}, y) \mid \forall \mathbf{c} \in \mathbf{C} \quad \mathbf{x} = K_x(\mathbf{c}), \quad y = K_y(\mathbf{c})\}. \quad (8.5)$$

Получаем, что исходная задача аппроксимации кривой Ω сводится к аппроксимации выборки \mathfrak{D} . В этой статье предполагается, что выборка \mathfrak{D} аппроксируется линейной моделью:

$$g(\mathbf{x}, \mathbf{w}) = \mathbf{x}^\top \mathbf{w}, \quad (8.6)$$

где \mathbf{w} вектор параметров, который необходимо найти.

Чтобы найти оптимальный вектор параметров $\hat{\mathbf{w}}$, необходимо решить следующую оптимизационную задачу:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w} \in \mathbb{R}^n} \sum_{(\mathbf{x}, y) \in \mathfrak{D}} \|g(\mathbf{x}, \mathbf{w}) - y\|_2^2. \quad (8.7)$$

Таким образом, задача аппроксимации исходной кривой Ω сводится к решению задачи линейной регрессии, т.е. нахождению компонентов вектора $\hat{\mathbf{w}}$.

В случае, когда на изображении K кривые второго порядка $\Omega_1, \dots, \Omega_K$, для каждой из которых есть экспертная информация $E_k = E(\Omega_k)$, $k \in \{1, \dots, K\}$ ставится задача построения мультимодели, называемой смесью K экспертов.

Определение 8.1. Мульти модель f называется смесью K экспертов

$$f = \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) g_k(\mathbf{w}_k), \quad \pi_k(\mathbf{x}, \mathbf{V}) : \mathbb{R}^{n \times |\mathbf{V}|} \rightarrow [0, 1], \quad \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) = 1, \quad (8.8)$$

где g_k является локальной моделью — экспертом, а \mathbf{x} является признаком описанием объекта, π_k — шлюзовая функция, вектор \mathbf{w}_k является параметрами локальной модели, а матрица \mathbf{V} является параметрами шлюзовой функции.

Для каждой кривой второго порядка даны отображения (8.3). Для удобства введем следующие обозначения: $K_x^k(\mathbf{c}) = K_x(\Omega_k, \mathbf{c})$ и $K_y^k(\mathbf{c}) = K_y(\Omega_k, \mathbf{c})$. Затем с помощью локальных линейных моделей строится универсальная мульти модель, описывающая кривые $\Omega_1, \dots, \Omega_K$ на изображении \mathbf{M} :

$$f = \sum_{\mathbf{c} \in \mathbf{C}} \sum_{k=1}^K \pi_k(\mathbf{c}, \mathbf{V}) g_k(K_x^k(\mathbf{c}), \mathbf{w}_k), \quad (8.9)$$

где π_k задает шлюзовую функцию. Рассматривается случай, когда $\mathbf{x} = K_x^1(\mathbf{c}) = \dots = K_x^K(\mathbf{c})$, то есть выражение (8.9) переписывается в следующей простой форме:

$$f = \sum_{\mathbf{c} \in \mathbf{C}} \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) g_k(\mathbf{x}, \mathbf{w}_k), \quad (8.10)$$

где шлюзовая функция π_k имеет следующий вид:

$$\pi_k(\mathbf{x}, \mathbf{V}) : \mathbb{R}^{n \times |\mathbf{V}|} \rightarrow [0, 1], \quad \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) = 1, \quad (8.11)$$

где \mathbf{V} — параметры шлюзовой функции, а g_k — локальная модель.

Рассматривается следующий вид функций:

$$\boldsymbol{\pi}(\mathbf{x}, \mathbf{V}) = \text{softmax}(\mathbf{V}_1^\top \boldsymbol{\sigma}(\mathbf{V}_2^\top \mathbf{x})), \quad (8.12)$$

где $\mathbf{V} = \{\mathbf{V}_1, \mathbf{V}_2\}$ параметры шлюзовой функции, $\mathbf{V}_1 \in \mathbb{R}^{p \times k}$, $\mathbf{V}_2 \in \mathbb{R}^{n \times p}$.

Чтобы найти оптимальные параметры мульти модели, необходимо решить следующую оптимизационную задачу:

$$\mathcal{L} = \sum_{(\mathbf{x}, y) \in \mathfrak{D}} \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V})(y - \mathbf{w}_k^\top \mathbf{x})^2 + R(\mathbf{V}, \mathbf{W}, E(\Omega)) \rightarrow \min_{\mathbf{V}, \mathbf{W}}, \quad (8.13)$$

где $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_k]$ — параметры локальных моделей, $R(\mathbf{V}, \mathbf{W}, E(\Omega))$ регуляризационный параметр на основе экспертной информации.

Единое пространство для кривых второго порядка. Произвольная кривая второго порядка, главная ось которой не параллельна оси ординат, задается следующим выражением:

$$x^2 = B'xy + C'y^2 + D'x + E'y + F', \quad (8.14)$$

где на коэффициенты B', C' действуют ограничения, зависящие от типа кривой. Выражение (8.4) принимает следующий вид:

$$K_x(\mathbf{c}_i) = [x_i y_i, y_i^2, x_i, y_i, 1], \quad K_y(\mathbf{c}_i) = x_i^2, \quad (8.15)$$

откуда получается задача линейной регрессии для восстановления параметров B', C', D', E', F' по выбранной выборке.

Окружность. Как частный случай кривой второго порядка, рассматривается окружность. Пусть (x_0, y_0) - это центр окружности, которую нужно найти на бинарном изображении \mathbf{M} , а r - его радиус. Элементы $(x_i, y_i) \in \mathbf{C}$ представляют собой геометрическое место точек, которое аппроксимируется уравнением окружности:

$$(x_i - x_0)^2 + (y_i - y_0)^2 = r^2. \quad (8.16)$$

Раскладывая скобки, получаем:

$$(2x_0) \cdot x_i + (2y_0) \cdot y_i + (r^2 - x_0^2 - y_0^2) \cdot 1 = x_i^2 + y_i^2. \quad (8.17)$$

Тогда выражение (8.4) принимает следующий вид:

$$K_x(\mathbf{c}_i) = [x_i, y_i, 1] = \mathbf{x}, \quad K_y(\mathbf{c}_i) = x_i^2 + y_i^2 = y. \quad (8.18)$$

Получаем задачу линейной регрессии (8.5). Компоненты вектора $\mathbf{w} = [w_0, w_1, w_2]^T$ связывают признаковое описание \mathbf{x} и целевую переменную y . Параметры окружности, через параметры линейной модели:

$$x_0 = \frac{w_0}{2}, \quad y_0 = \frac{w_1}{2}, \quad r = \sqrt{w_3 + x_0^2 + y_0^2}. \quad (8.19)$$

8.2 Композиция кривых второго порядка на изображении

Для построения композиции из фигур воспользуемся выражением (8.13), которое принимает следующий вид:

$$\mathcal{L} = \sum_{\mathbf{c} \in \mathbf{C}} \sum_{k=1}^K \pi_k(\mathbf{c}, \mathbf{V}) (K_y^k(\mathbf{c}) - \mathbf{w}_k^T K_x^k(\mathbf{c}))^2 + R(\mathbf{V}, \mathbf{W}, E(\Omega)) \rightarrow \min_{\mathbf{V}, \mathbf{W}}, \quad (8.20)$$

где K_x^k, K_y^k экспертное представление k -го эксперта. Предполагая, что все кривые на изображении описываются одним признаковым описанием $\mathbf{x} = K_x^1(\mathbf{c}) = \dots = K_x^K(\mathbf{c}), x = K_y^1(\mathbf{c}) = \dots = K_y^K(\mathbf{c})$, получаем следующую задачу оптимизации:

$$\mathcal{L} = \sum_{(\mathbf{x}, y) \in \mathfrak{D}} \sum_{k=1}^K \pi_k(\mathbf{x}, \mathbf{V}) (y - \mathbf{w}_k^T \mathbf{x})^2 + R(\mathbf{V}, \mathbf{W}, E(\Omega)) \rightarrow \min_{\mathbf{V}, \mathbf{W}}, \quad (8.21)$$

В качестве регуляризатора R рассматриваются дополнительные ограничения на векторы параметров модели. Для решения задачи оптимизации (8.21) предлагается использовать ЕМ-алгоритм.

8.3 Анализ смеси экспертов для аппроксимации кривых второго порядка на изображении

Проведен вычислительный эксперимент по анализу качества моделей кривых второго порядка на изображении. Эксперимент разделен на несколько частей. Первая часть - это эксперимент с несколькими окружностями на изображении. Во второй части анализируется сходимость метода в зависимости от уровня шума в данных и от указанной экспертной информации. В третьей части проводится эксперимент по аппроксимации радужной оболочки глаза.

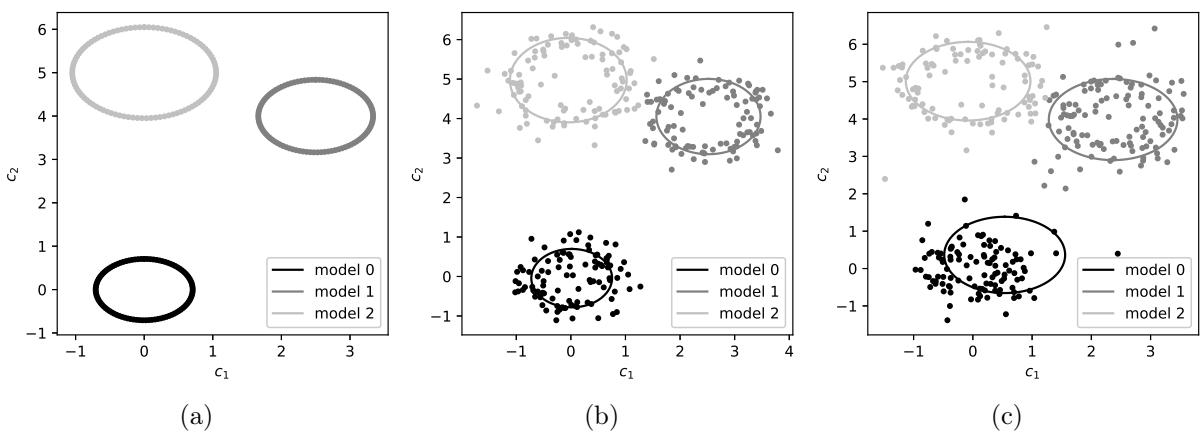


Рис. 43: Мультимодель в зависимости от различных предварительных предположений и уровня шума. Слева направо: окружности без шума; шум в радиусе круга; шум в радиусе круга, а также произвольные точки по всему изображению.

В этой части эксперимента показан пример обучения мультимодели для аппроксимации нескольких фигур второго порядка одновременно. В качестве данных используется синтетическая выборка, который получается путем создания трех произвольных непересекающихся окружностей, а также добавления шума к этим окружностям. Шум был добавлен к радиусу круга для каждой точки, а также случайные точки были добавлены к выборке.

На рисунке 43 показан результат построения ансамбля локально аппроксимирующих моделей, которые аппроксимируют образец. Каждая локальная модель аппроксимирует одну окружность, а при добавлении еще большего шума качество аппроксимации падает. На рисунке 44 показан график зависимости радиуса окружностей r и их центров (x_0, y_0) от номера итерации.

В этой части эксперимента анализируется качество аппроксимации S в зависимости от уровня шума β в данных и по параметру априорных распределений γ . Пример получается следующим образом: сначала случайным образом выбираются два вектора параметров: $\mathbf{w}_1^{\text{true}}$ и $\mathbf{w}_2^{\text{true}}$ коэффициенты двух парабол. Векторы $\mathbf{w}_1^{\text{true}}$ и $\mathbf{w}_2^{\text{true}}$ используются для создания точек x_i и y_i с добавлением нормального шума $\varepsilon \sim \mathcal{N}(0, \beta)$. При обучении мультимодели, априорное распределение параметров считается $\mathbf{w}_1 \sim \mathcal{N}(\mathbf{w}_1^{\text{true}}, \gamma \mathbf{I})$, $\mathbf{w}_2 \sim \mathcal{N}(\mathbf{w}_2^{\text{true}}, \gamma \mathbf{I})$.

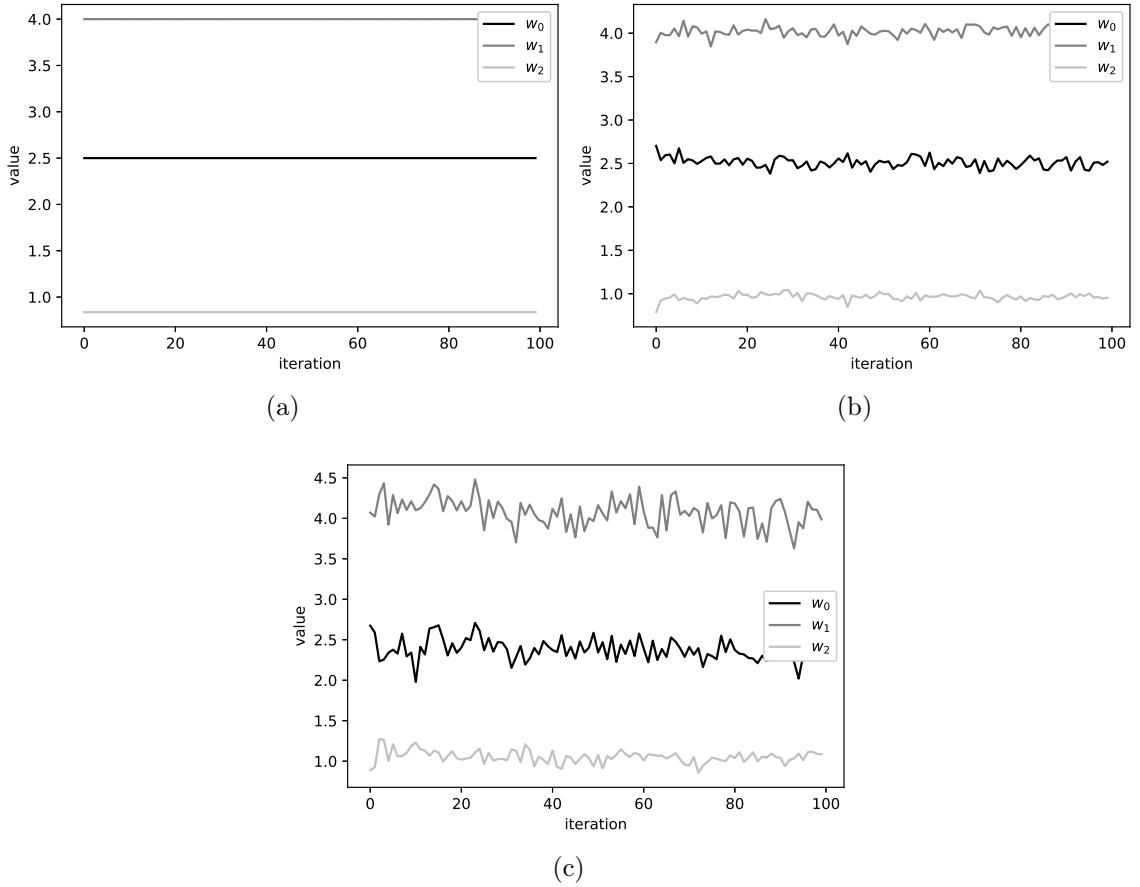


Рис. 44: Зависимость параметров r , x_0 и y_0 от номера итерации для различных априорных распределений. Слева направо: окружности без шума; шум в радиусе круга; шум в радиусе круга, а также произвольные точки по всему изображению.

Рассмотрим критерий качества:

$$S = \|\mathbf{w}_1^{\text{pred}} - \mathbf{w}_1^{\text{true}}\|_2^2 + \|\mathbf{w}_2^{\text{pred}} - \mathbf{w}_2^{\text{true}}\|_2^2, \quad (8.22)$$

где $\mathbf{w}_1^{\text{pred}}$ аппроксимация вектора параметров первой локальной модели, а $\mathbf{w}_2^{\text{pred}}$ аппроксимация вектора параметров второй локальной модели.

На рисунке 46 показана зависимость критерия качества S от уровня шума β и параметра априорного распределения γ . График показывает, что при низком уровне шума β качество приближения не зависит от параметра γ , а с увеличением шума β качество приближения S снижается.

На рисунке 46 показан пример того, как алгоритм работает с разными параметрами β и γ . Видно, что в отсутствие шума β обе локальные модели аппроксимируют образец. С увеличением уровня шума качество аппроксимации снижается: при $\beta = 0,2$ при увеличении γ первая локальная модель от параболы переходит в эллипс; при $\beta = 0,4$ при увеличении γ первая локальная модель из параболы переходит в эллипс, а вторая модель из параболы переходит в гиперболу.

Анализ качества аппроксимации проводится для задачи аппроксимации ра-

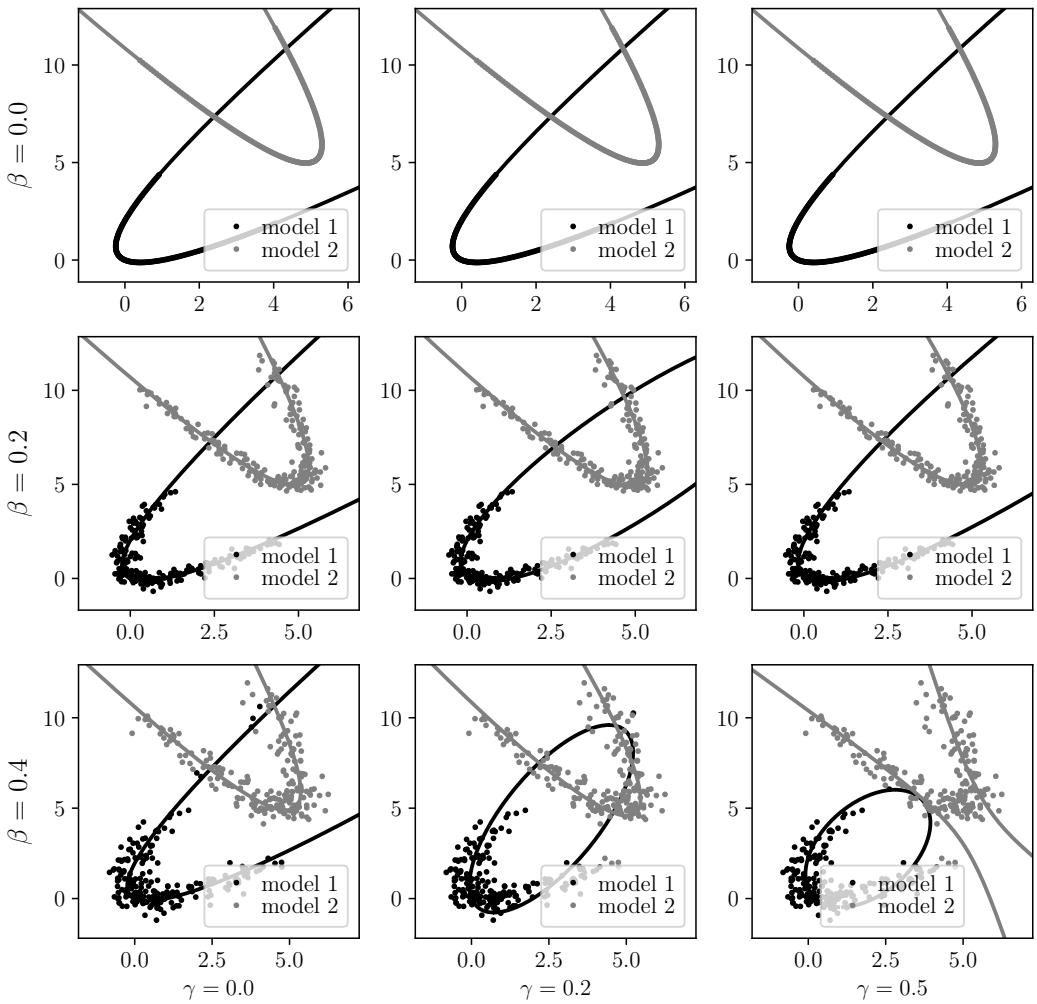


Рис. 45: Результат аппроксимации для данных с разными уровнями шума β и дисперсией априорного распределения γ

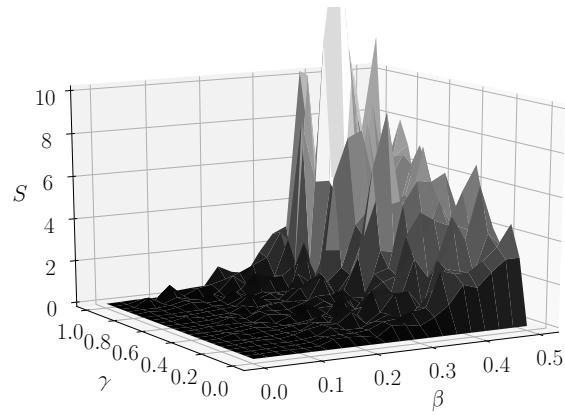


Рис. 46: Зависимость моделей от уровня шума β в данных, а также от дисперсии априорного распределения γ

дужной оболочки глаза на изображении. Радужная оболочка глаза состоит из двух концентрических окружностей, поэтому рассматривается мульти модель, состоящая из двух экспертов: каждый эксперт аппроксимирует одно из обсто-

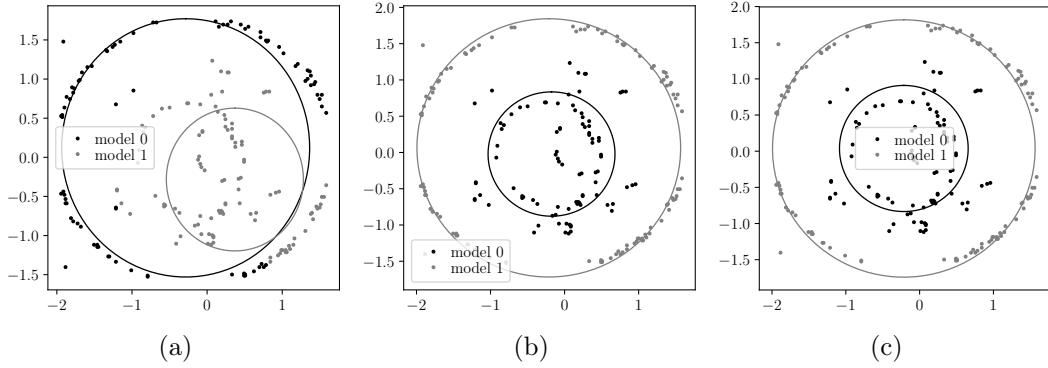


Рис. 47: Визуализация приближения радужной оболочки: а) если указан регуляризатор R_0 ; б) если указан регуляризатор R_1 ; в) если указан регуляризатор R_2

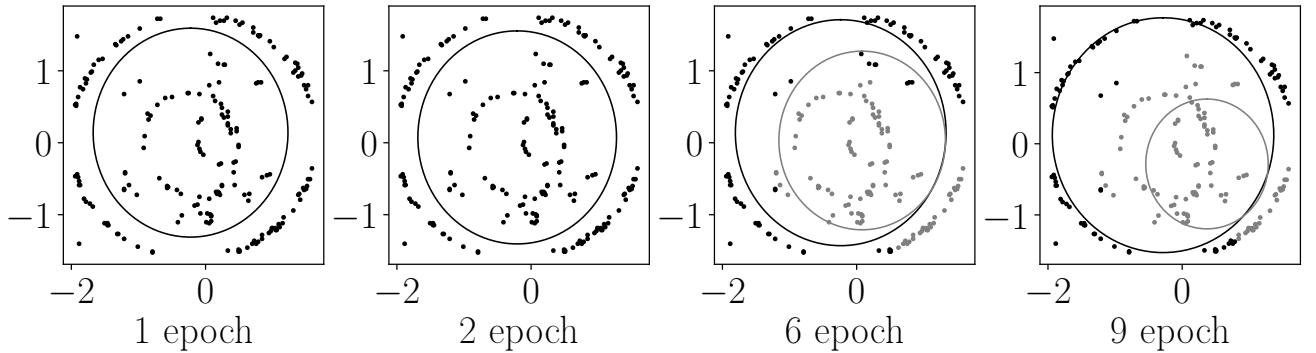


Рис. 48: Визуализация процесса сходимости параметров мульти модели в случае регуляризатора R_0

ятельств. В вычислительном эксперименте сравнивается качество аппроксимации окружностей при задании разных регуляризаторов R_0, R_1, R_2 . Регуляризатор $R_0(\mathbf{V}, \mathbf{W}, E(\Omega)) = 0$, то есть регуляризатора нет. Регуляризатор:

$$R_1(\mathbf{V}, \mathbf{W}, E(\Omega)) = -\sum_{k=1}^K \mathbf{w}_k^\top \mathbf{w}_k, \quad (8.23)$$

который удаляет оклонулевые параметры локальных моделей. Регуляризатор

$$R_2(\mathbf{V}, \mathbf{W}, E(\Omega)) = -\sum_{k=1}^K \mathbf{w}_k^\top \mathbf{w}_k + \sum_{k=1}^K \sum_{k'=1}^K \sum_{j=1}^2 (w_k^j - w_{k'}^j)^2, \quad (8.24)$$

что способствует совпадению центров окружностей и близким к нулю параметрам модели. На рисунке 47 показан результат алгоритма аппроксимации радужной оболочки глаза после 10 итераций. Видно, что при отсутствии регуляризатора одна из окружностей находится некорректно. Если задан регуляризатор R_1 , модель аппроксимирует обе окружности с хорошим качеством, но окружности не концентрические. В случае задания регуляризатора R_2 мы получим концентрические окружности на изображении.

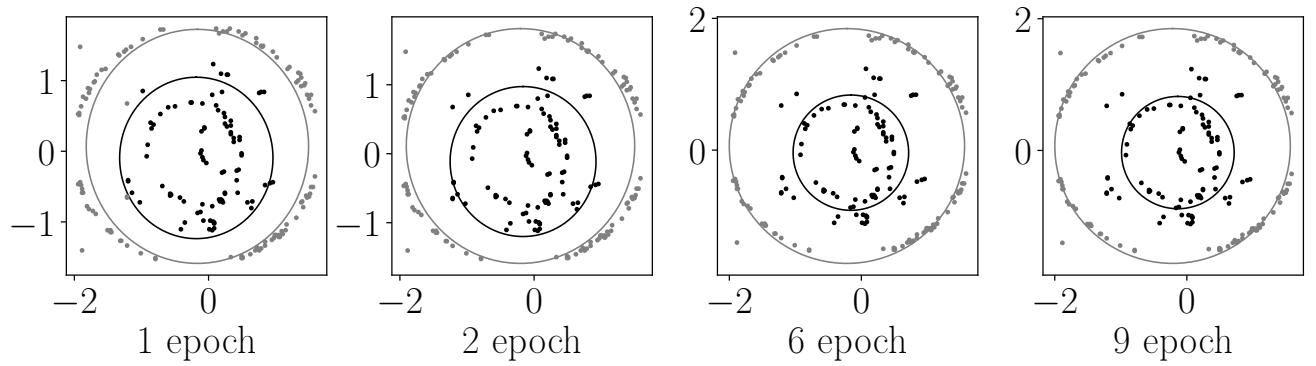


Рис. 49: Визуализация процесса сходимости параметров мульти модели в случае регуляризатора R_1

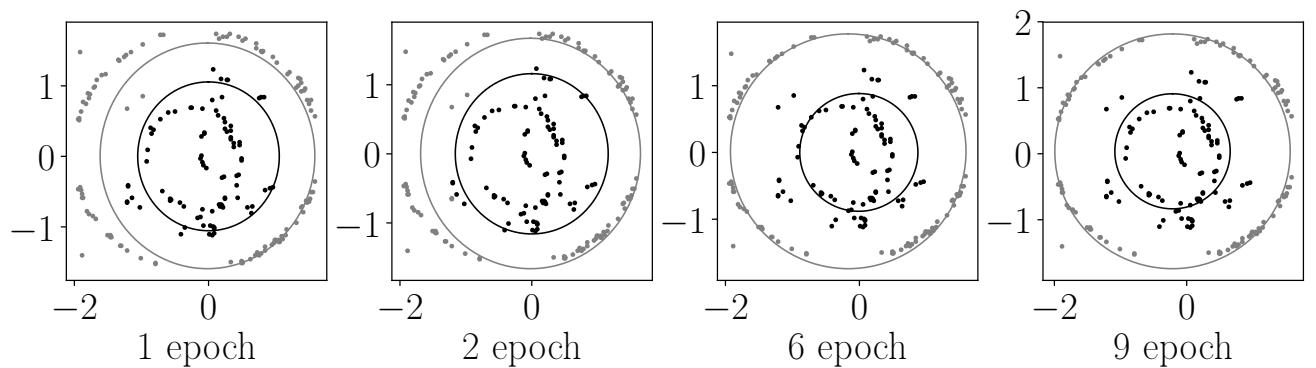


Рис. 50: Визуализация процесса сходимости параметров мульти модели в случае регуляризатора R_2

На рисунках 48-50 показан процесс сходимости мультимоделей в случае задания разных регуляризаторов R_0, R_1, R_2 . Видно, что модели с типом регуляризатора R_1 и R_2 аппроксимируют обе окружности, а мультимодель с регуляризатором R_0 аппроксимирует только большую окружность.

9 Локальные модели в задачах кластеризации временных рядов

Анализ физической активности человека производится при помощи мобильных телефонов, разумных часов [51, 52]. Эти устройства используют акселерометр, гироскоп и магнитометр. Цель данной работы заключается в разметке и распознавании человеческой активности [53, 54, 55], а также поиска начала каждого действия [56]. Примерами одного сегмента действия служит шаг, шаг бега, приседание, прыжок и др. Исследуются последовательности, которые состоят не менее чем из двух подряд идущих сегментов, которые соответствуют одному и тому же типу человеческой активности.

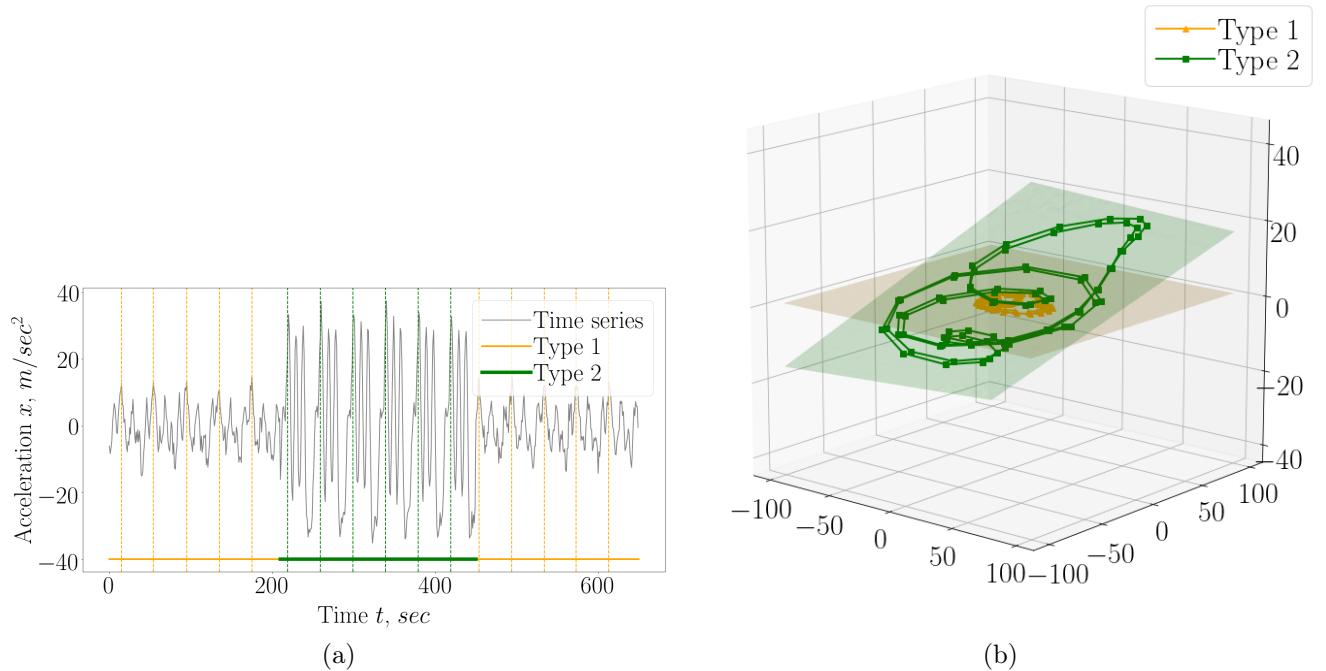


Рис. 51: Временной ряд, с разметкой на кластеры: а) временной ряд с аксессорской разметкой на кластеры и выделением начала квазипериодического сегмента; б) проекция фазовых траекторий на первые две главные компоненты

Временные ряды — это объекты сложной структуры. При их классификации значимую роль играет модель построения признакового пространства. В данной работе объектом анализа и кластеризации является точка на оси времени. Решается задача кластеризации точек временного ряда. При *кластеризации* каждой точке временного ряда ставится в соответствие метка из конечного множества меток. Каждая метка соответствует одному характерному физическому действию. *Сегмент* это часть временного ряда, которая соответствует одному характерному физическому действию, например: шаг двумя ногами при ходьбе, или шаг двумя ногами при беге. Последовательность сегментов, которые соответствуют одному физическому действию образуют *цепочку* действий. Предполагается, что цепочка действий образует квазипериодическую последовательность значений временного ряда. Последовательность точек $\{b_t\}_{t=1}^N$ назовем *квазипериодической* с периодом T , если для всех t найдется Δ , такое

что:

$$b_t \approx b_{t+\Delta}, \quad |\Delta| \ll T. \quad (9.1)$$

Пример кластеризации и разбиения ряда на сегменты показан на рис. 51а. Данный ряд разбит на два характерных физических действия, которые обозначаются Type 1 и Type 2. Также данный ряд содержит в себе две квазипериодические цепочки действий.

Решение задачи кластеризации состоит из двух этапов. Во-первых, для получения признакового описания временного ряда предлагается алгоритм локальной аппроксимации временного ряда при помощи метода главных компонент [61]. Под *локальной* аппроксимацией временного ряда подразумевается, что для признакового описания его точки используется не весь ряд, а только некоторая окрестность данной точки. В качестве признакового описания точки временного ряда рассматриваются две главные компоненты *сегмента фазовой траектории* в окрестности данной точки. На рис. 51б показаны две первые главные компоненты *фазовых траекторий*, а также проекция фазовых траекторий на эти компоненты. Они соответствуют разным физическим действиям, которые обозначаются Type 1 и Type 2, внутри одного временного ряда. Как видно плоскости, которые порождены данными главными компонентами не совпадают. Это говорит о том, что наблюдаются различные действия. Во-вторых, вводится функция расстояния в построенном пространстве признакового описания. Данная функция является расстояниям между двумя базисами некоторых подпространств внутри всего фазового пространства временного ряда. На рис. 51б данная функция является некоторым расстояниям между двумя плоскостями. Получив расстояния между точками временного ряда, выполним кластеризацию данных точек. Задача сегментации внутри каждого кластера решается при помощи метода, который рассмотрен в [56].

Для решения задачи кластеризации точек временного ряда вводятся предположения. Предполагается, что периоды различных сегментов различаются незначительно, причем известны минимальный и максимальный периоды сегмента и число различных сегментов внутри временного ряда. Также предполагается, что тип активности во времени не меняется часто, а также что фазовые траектории разных сегментов являются различными.

Проверка и анализ метода кластеризации проводится на синтетической и реальной выборках. Синтетическая выборка построенная при помощи суммы нескольких первых членов ряда Фурье со случайными коэффициентами. Эксперимент по сегментации временного ряда проводился на простых синусоидальных сигналах с произвольной амплитудой и частотой. Реальные данные получены при помощи мобильного акселерометра, который снимал показания во время некоторой физической активности человека.

В [51] рассматривается метод построения признакового описания на основе экспертно заданных порождающих функций. В [57] рассматривается метод построения признаков на основе гипотезы порождения данных. В [58] рассматривается комбинированное признаковое описание на основе данных методов. В

[59] рассматривается проблема построение признакового пространства и предлагается критерий избыточности выбранных признаков.

Работа [56] является ближайшей работой по данной теме. Она заключается в поиске начала сегмента внутри квазипериодического сигнала, который состоит, только из одной цепочки действий. Этот метод основан на исследовании фазового пространства, а именно поиска устойчивой гиперплоскости, которая делит фазовое пространство на две равные части. В качестве начала сегмента выбираются точки, которые находятся близко к данной гиперплоскости. В [56] предлагается выполнить проекцию фазового пространства на первые две главные компоненты, после чего провести устойчивую прямую, выделив начала каждого сегмента. Данный метод имеет недостаток в том, что позволяет находить начало только для временного ряда, который состоит из квазипериодического сигнала единственного типа.

Также близкой является работа [55]. Данная работа заключается в поиске периодической структуры внутри ряда при помощи модели LSTM с модифицированным механизмом Attention. Предполагается, что механизм Attention будет давать максимальное значение score в точках, которые удалены от данной на целое количество периодов.

9.1 Постановка задачи кластеризации точек временного ряда

Задан временной ряд

$$\mathbf{x} \in \mathbb{R}^N, \quad (9.2)$$

где N число точек временного ряда. Он состоит из последовательности сегментов:

$$\mathbf{x} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_M], \quad (9.3)$$

где \mathbf{v}_i некоторый сегмент из множества сегментов \mathbf{V} , которые встречаются в данном ряде. Причем для всех i либо $[\mathbf{v}_{i-1}, \mathbf{v}_i]$ либо $[\mathbf{v}_i, \mathbf{v}_{i+1}]$ является цепочкой действий. Пусть множество \mathbf{V} удовлетворяет следующим свойствам:

$$|\mathbf{V}| = K, \quad \mathbf{v} \in \mathbf{V} \quad |\mathbf{v}| \leq T, \quad (9.4)$$

где $|\mathbf{V}|$ число различных действий в множестве сегментов \mathbf{V} , $|\mathbf{v}|$ длина сегмента, а K и T это число различных действий во временном ряде и длина максимального сегмента соответственно.

Рассматривается отображение

$$a : t \rightarrow \mathbb{Y} = \{1, \dots, K\}, \quad (9.5)$$

где $t \in \{1, \dots, N\}$ некоторый момент времени, на котором задан временной ряд. Требуется, чтобы отображение a удовлетворяло следующим свойствам:

$$\begin{cases} a(t_1) = a(t_2), & \text{если в моменты } t_1, t_2 \text{ совершаются один тип действий} \\ a(t_1) \neq a(t_2), & \text{если в моменты } t_1, t_2 \text{ совершаются разные типы действий} \end{cases} \quad (9.6)$$

Пусть задана некоторая асессорская разметка временного ряда:

$$\mathbf{y} \in \{1, \dots, K\}^N. \quad (9.7)$$

Тогда ошибка алгоритма a на временном ряде \mathbf{x} представляется в следующем виде:

$$S = \frac{1}{N} \sum_{t=1}^N [y_t = a(t)], \quad (9.8)$$

где t — момент времени, y_t асессорская разметка t -го момента времени для заданного временного ряда.

9.2 Кластеризация точек в фазовом пространстве

Рассмотрим фазовую траекторию временного ряда \mathbf{x} :

$$\mathbf{H} = \{\mathbf{h}_t | \mathbf{h}_t = [x_{t-T}, x_{t-T+1}, \dots, x_t], T \leq t \leq N\}, \quad (9.9)$$

где \mathbf{h}_t — точка фазовой траектории.

Информация об длине максимального сегмента T внутри временного ряда позволяет разбить фазовую траекторию на сегменты из $2T$ векторов:

$$\mathbf{S} = \{\mathbf{s}_t | \mathbf{s}_t = [\mathbf{h}_{t-T}, \mathbf{h}_{t-T+1}, \dots, \mathbf{h}_{t+T-1}], T \leq t \leq N - T\}, \quad (9.10)$$

где \mathbf{s}_t — это сегмент фазовой траектории. Данные сегменты имеют всю локальную информацию об временном ряде, так как содержит всю информацию на периоде до момента времени t и информацию о периоде после момента времени t .

В качестве признакового описания точки временного ряда t рассматривают главные компоненты \mathbf{W}_t для T -мерных сегментов \mathbf{s}_t . Сегмент \mathbf{s}_t проектируется на подпространство размерности два при помощи метода главных компонент $\mathbf{z}_t = \mathbf{W}_t \mathbf{s}_t$. Получаем:

$$\mathbf{W} = \{\mathbf{W}_t | \mathbf{W}_t = [\lambda_t^1 \mathbf{w}_t^1, \lambda_t^2 \mathbf{w}_t^2]\}, \quad \Lambda = \{\boldsymbol{\lambda}_t | \boldsymbol{\lambda}_t = [\lambda_t^1, \lambda_t^2]\}, \quad (9.11)$$

где $[\mathbf{w}_t^1, \mathbf{w}_t^2]$ и $[\lambda_t^1, \lambda_t^2]$ это базисные векторы и соответствующие им собственные для сегмента фазовой траектории \mathbf{s}_t .

Для кластеризации точек временного ряда рассмотрим функцию расстояния между элементами $\mathbf{W}_{t_1}, \mathbf{W}_{t_2}$:

$$\rho(\mathbf{W}_1, \mathbf{W}_2) = \max \left(\max_{\mathbf{e}_2 \in \mathbf{W}_2} d_1(\mathbf{e}_2), \max_{\mathbf{e}_1 \in \mathbf{W}_1} d_2(\mathbf{e}_1) \right), \quad (9.12)$$

где \mathbf{e}_i это базисный вектор пространства \mathbf{W}_i , а $d_i(\mathbf{e})$ является расстоянием от вектора \mathbf{e} до пространства \mathbf{W}_i .

В случае, когда все подпространства \mathbf{W}_t имеют размерность два, расстояние $\rho(\mathbf{W}_1, \mathbf{W}_2)$ имеет следующую интерпретацию:

$$\rho(\mathbf{W}_1, \mathbf{W}_2) = \max_{\{\mathbf{a}, \mathbf{b}, \mathbf{c}\} \subset \mathbf{W}_1 \cup \mathbf{W}_2} V(\mathbf{a}, \mathbf{b}, \mathbf{c}), \quad (9.13)$$

где $\mathbf{W}_1 \cup \mathbf{W}_2$ это объединение базисных векторов первого и второго пространства, $V(\mathbf{a}, \mathbf{b}, \mathbf{c})$ — объем параллелепипеда построенного на векторах $\mathbf{a}, \mathbf{b}, \mathbf{c}$, которые являются столбцами матрицы $\mathbf{W}_1 \cup \mathbf{W}_2$.

Рассмотрим расстояние между собственными числами:

$$\rho(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = \sqrt{(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)^T (\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2)}. \quad (9.14)$$

Используя выражения (9.13-9.14) введем расстояние между двумя точками t_1, t_2 временного ряда, а также рассмотрим матрицу попарных расстояний \mathbf{M} между точками данного ряда:

$$\rho(t_1, t_2) = \rho(\mathbf{W}_1, \mathbf{W}_2) + \rho(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2), \quad \mathbf{M} = \mathbb{R}^{N \times N}, \quad (9.15)$$

где матрица \mathbf{M} является матрицей попарных расстояний между всеми парами точек t временного ряда \mathbf{x} . Используя матрицу попарных расстояний \mathbf{M} выполним кластеризацию моментов времени t временного ряда (9.5):

9.3 Анализ фазовых траекторий в задаче кластеризации точек временного ряда

Для анализа свойств предложенного алгоритма кластеризации был проведен вычислительный эксперимент в котором кластеризация точек временного ряда проводилась используя матрицы попарных расстояний (9.15).

В качестве данных использовались две выборки временных рядов, которые описаны в таблице 12. Выборка Physical Motion это реальные временные ряды полученные при помощи мобильного акселерометра. Синтетические временные ряды были построены при помощи нескольких первых слагаемых ряда Фурье со случайными коэффициентами из стандартного нормального распределения. Генерация данных состояла из двух этапов. На первом этапе генерировались короткие сегменты \mathbf{v} для построения множества \mathbf{V} . Вторым этапом генерации выборки \mathbf{x} является следующим случайнм процессом:

$$\mathbf{x} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_M] + \boldsymbol{\epsilon}, \quad \begin{cases} \mathbf{v}_1 \sim \mathcal{U}(\mathbf{V}), \\ \mathbf{v}_i = \mathbf{v}_{i-1}, & \text{с вероятностью } \frac{3}{4}, \\ \mathbf{v}_i \sim \mathcal{U}(\mathbf{V}), & \text{с вероятностью } \frac{1}{4} \end{cases} \quad (9.16)$$

где $\mathcal{U}(\mathbf{V})$ — равномерное распределение на объектах из \mathbf{V} , а $\boldsymbol{\epsilon}$ является шумом из нормального распределения.

Синтетические данные. На рис. 52 приведен пример синтетических временных рядов. На рис. 52а показан пример ряда в котором число различных сегментов $K = 2$, а длина каждого сегмента $T = 20$. На рис. 52б показан пример ряда в котором число различных сегментов $K = 3$, а длина каждого сегмента $T = 20$.

Рис. 53 иллюстрирует матрицы попарных расстояний \mathbf{M} между всеми парами точек t временного ряда, которые построены при помощи (9.15). Используя

Таблица 12: Описание временных рядов в эксперименте кластеризации точек временного ряда

Ряд, \mathbf{x}	Длина ряда, N	Число сегментов, K	Длина сегмента, T
Physical Motion 1	900	2	40
Physical Motion 2	900	2	40
Synthetic 1	2000	2	20
Synthetic 2	2000	3	20

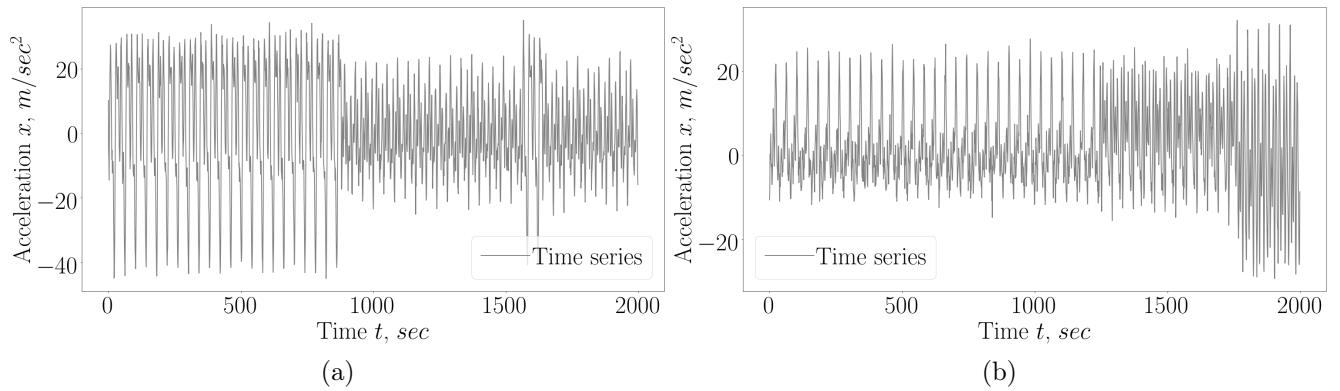


Рис. 52: Пример синтетически построенных временных рядов: а) для временно-го ряда Synthetic 1; б) для временного ряда Synthetic 2

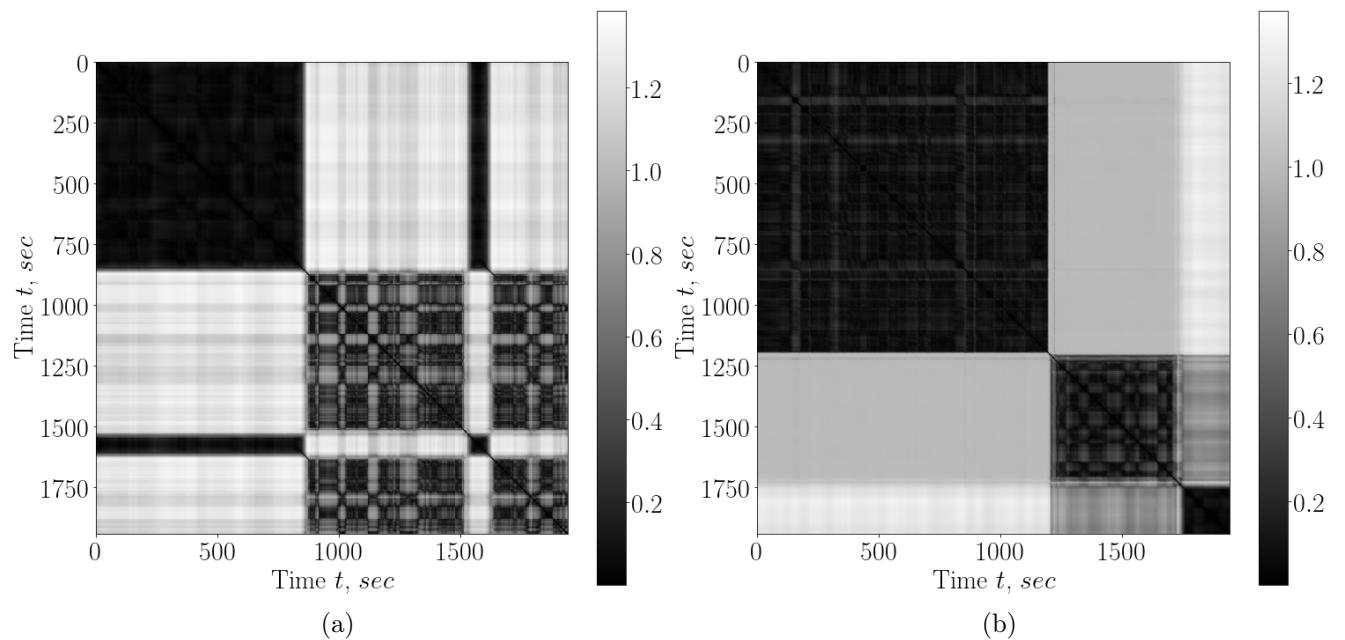


Рис. 53: Матрица попарных расстояний \mathbf{M} между точками временного ряда: а) для временного ряда Synthetic 1; б) для временного ряда Synthetic 2

матрицу попарных расстояний и метод Multidimensional Scaling [60] визуализи-руем точки временного ряда на плоскости. На рис. 54 показана визуализация

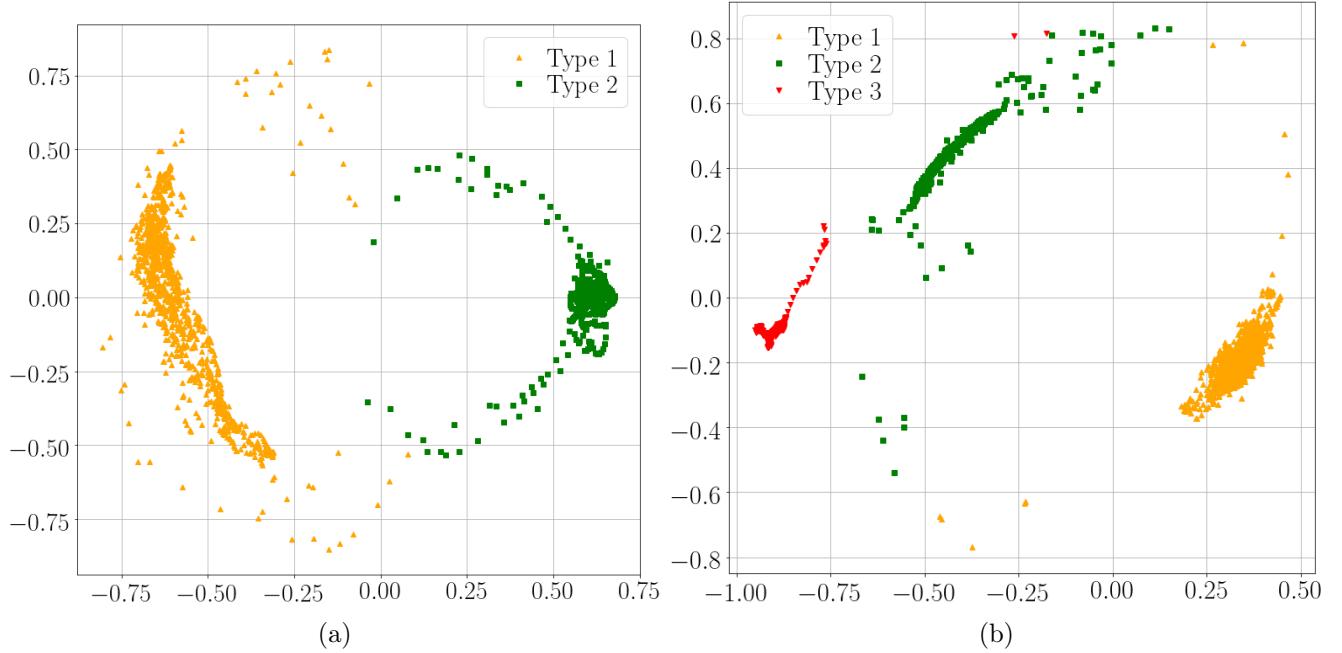


Рис. 54: Проекция точек временного ряда на плоскость при помощи матрицы попарных расстояний \mathbf{M} : а) для временного ряда Synthetic 1; б) для временного ряда Synthetic 2

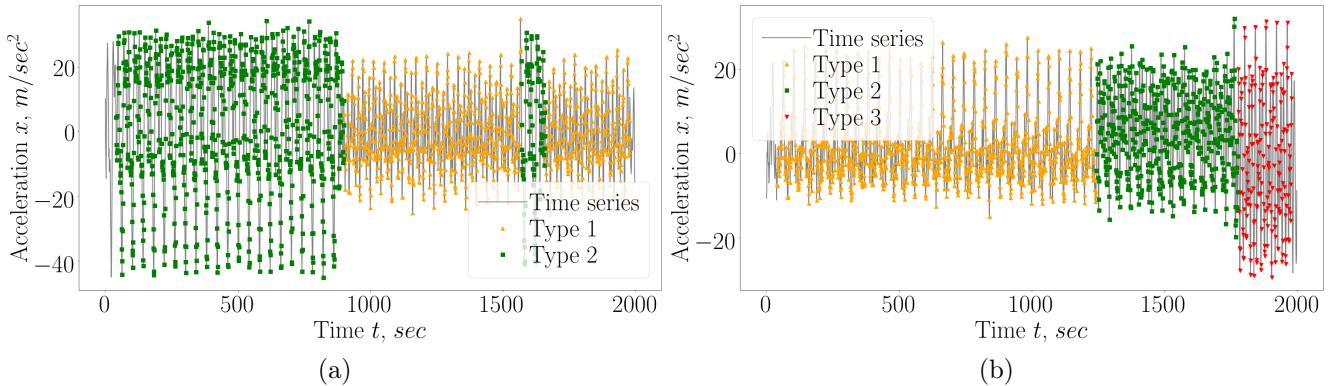


Рис. 55: Кластеризация точек временного ряда: а) для временного ряда Synthetic 1; б) для временного ряда Synthetic 2

точек на плоскости и выполнена их кластеризация при помощи метода иерархической кластеризации. Иллюстрация кластеров точек временного ряда продемонстрирована на рис. 55.

Реальные данные. На рис. 56 приведен пример реальных временных рядов полученных при помощи взятия одной из координат мобильного акселерометра.

Рис. 57 иллюстрирует матрицы попарных расстояний \mathbf{M} между всеми парами точек t временного ряда, которые построены при помощи (9.15). Используя матрицу попарных расстояний и метод Multidimensional Scaling [60] визуализируем точки временного ряда на плоскости. На рис. 58 показана визуализация

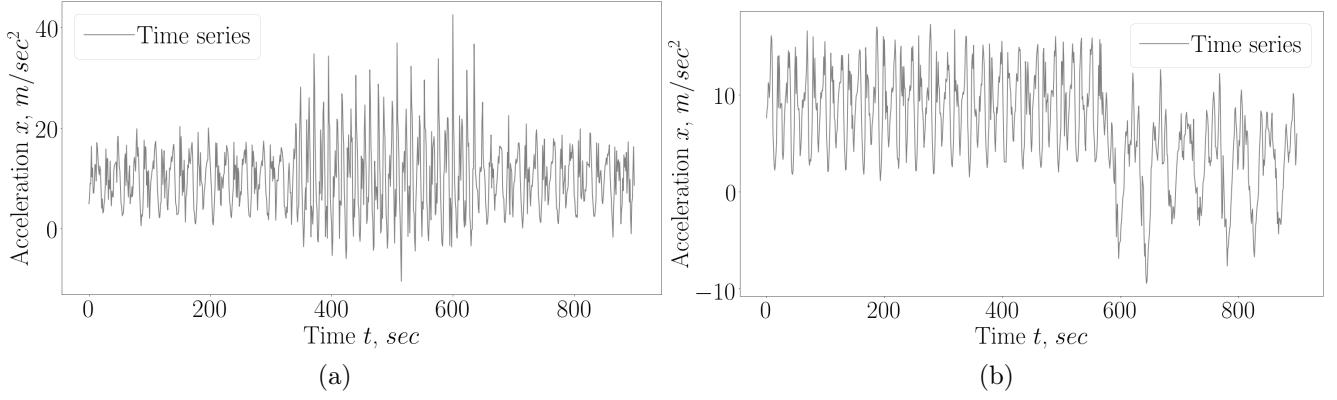


Рис. 56: Пример синтетически построенных временных рядов: а) для временно-го ряда Physical Motion 1; б) для временного ряда Physical Motion 2

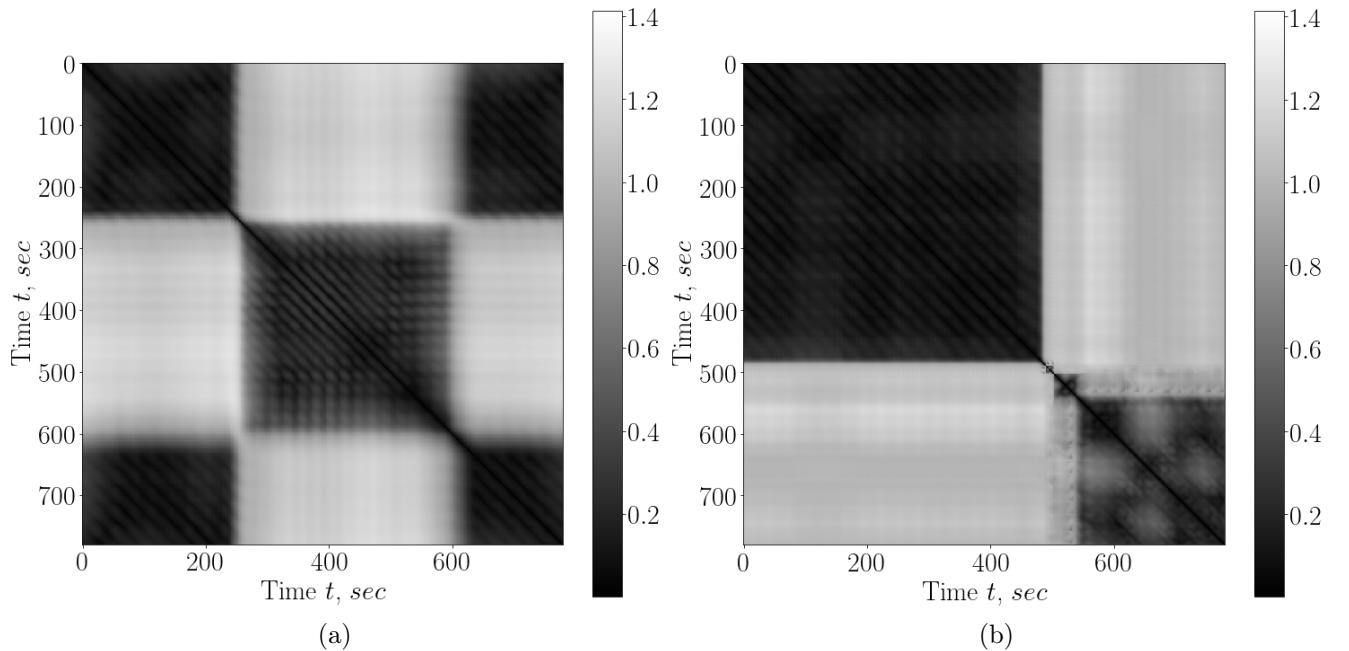


Рис. 57: Матрица попарных расстояний M между точками временного ряда: а) для временного ряда Physical Motion 1; б) для временного ряда Physical Motion 2

точек на плоскости и выполнена их кластеризация при помощи метода иерархической кластеризации. Иллюстрация кластеров точек временного ряда продемонстрирована на рис. 59.

Сегментация временных рядов проводится на синтетических и реальных данных. Для данного эксперимента в качестве синтетического ряда рассматривается ряд построенный из двух синусов с произвольной частотой и амплитудой. Описание временных рядов, которые используются в данном эксперименте представлены в таблице 13.

Сегментация проводится при помощи метода, который представлен в работе [56]. Данный метод применяется для каждого действия внутри временного ряда

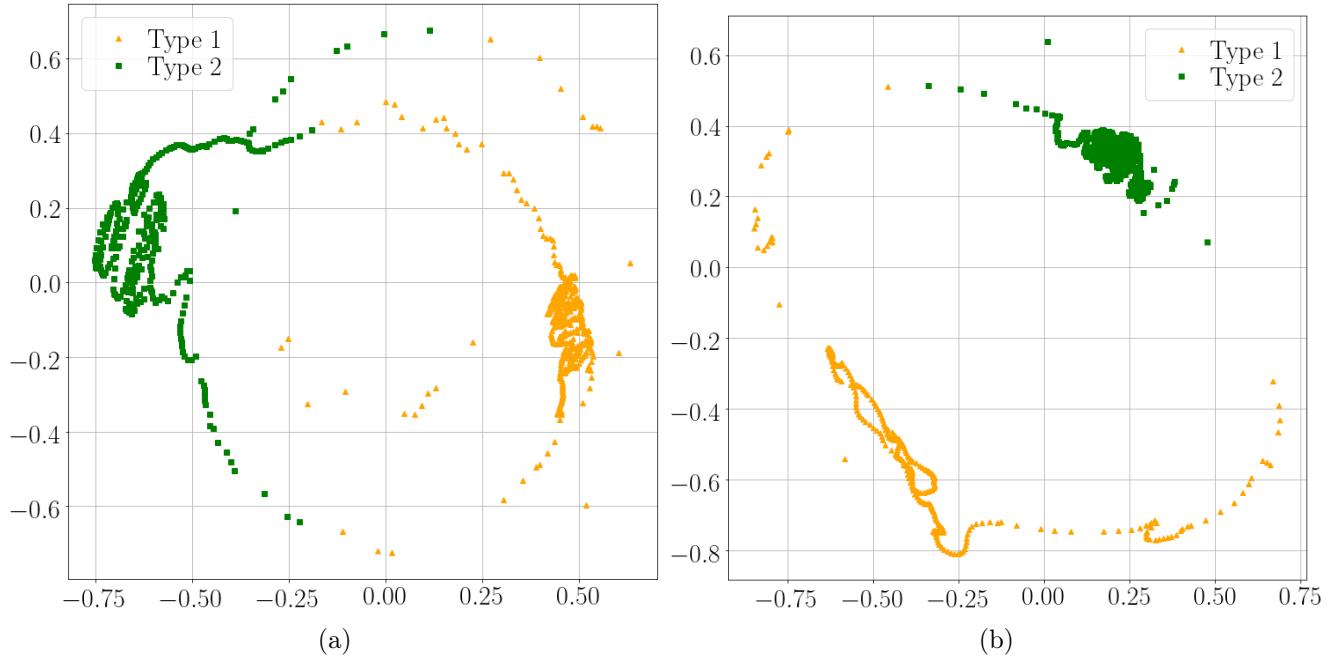


Рис. 58: Проекция точек временного на плоскость при помощи матрицы попарных расстояний \mathbf{M} : а) для временного ряда Physical Motion 1; б) для временного ряда Physical Motion 2

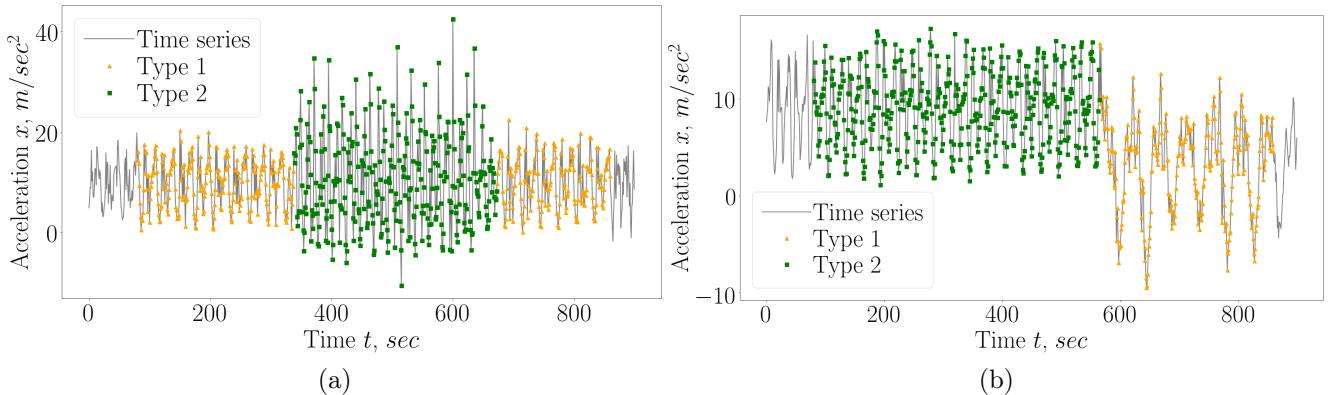


Рис. 59: Кластеризация точек временного ряда: а) для временного ряда Physical Motion 1; б) для временного ряда Physical Motion 2

по отдельности.

Таблица 13: Описание временных рядов в эксперименте сегментации временных рядов

Ряд, \mathbf{x}	Длина ряда, N	Число сегментов, K	Длина сегмента, T
Simple 1	1000	2	100
Physical Motion 2	900	2	40

Синтетические данные. На рис. 60 показан результат работы сегментации для временного ряда Simple 1. Данный алгоритм хорошо выделил начала сегментов. Также на рис. 60 показаны проекции фазовых пространств для обеих кластеров на их первые две главные компоненты.

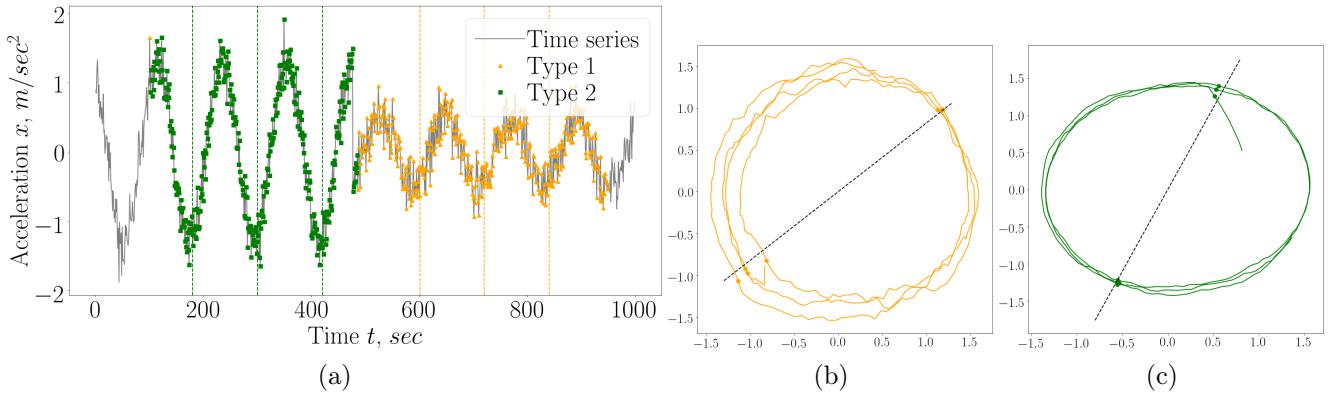


Рис. 60: Сегментация точек временного ряда Simple 1: а) сегментация временного ряда; б) проекция фазового пространства на первые две главные компоненты для первого кластера; в) проекция фазового пространства на первые две главные компоненты для второго кластера

Реальные данные. На рис. 61 показан результат работы сегментации для временного ряда Physical Motion 2. Данный алгоритм хорошо выделил начала сегментов для Type 1 и плохо для Type 2. Также на рис. 61 показаны проекции фазовых пространств для обеих кластеров на их первые две главные компоненты. Видно, что в случае проекции фазового пространства для части ряда, который относится к Type 2 получаем, что фазовая траектория имеет само-пересечение внутри одного сегмента, что влечет нахождения ложного начала сегмента.

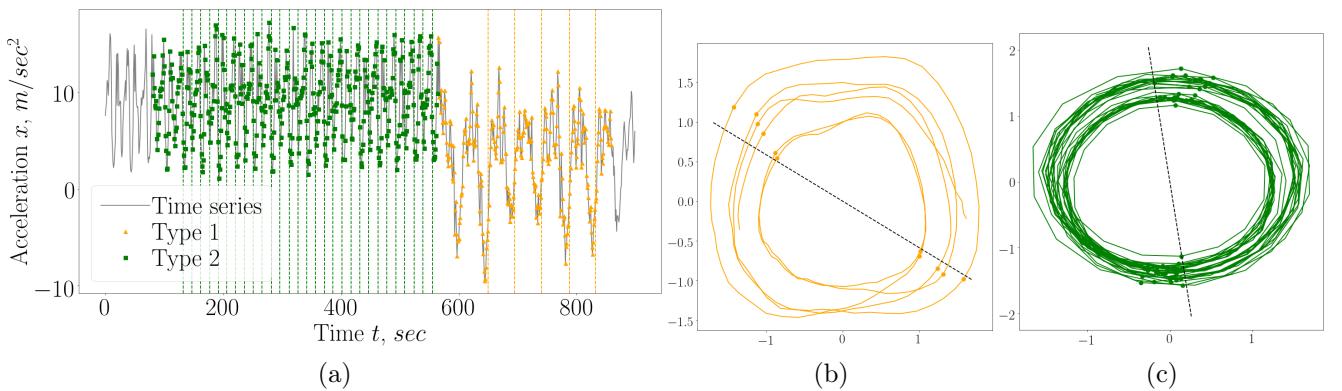


Рис. 61: Сегментация точек временного ряда Physical Motion 2: а) сегментация временного ряда; б) проекция фазового пространства на первые две главные компоненты для первого кластера; в) проекция фазового пространства на первые две главные компоненты для второго кластера

10 Заключение

В рамках данного реферата приведен обзор существующих подходов для снижения сложности моделей глубокого обучения. Снижение сложности моделей глубокого обучения производится с целью улучшения интерпретируемости моделей глубокого обучения.

Рассмотрена проблема задания порядка на множестве параметров сложных аппроксимирующих моделей. Исследован метод задания порядка на основе анализа стохастических свойств градиента функции ошибки \mathcal{L} по параметрам модели. Для задания порядка использовалась ковариационная матрица градиентов параметров \mathbf{C}_{η_0} , которая рассчитывается итеративно, в течение t_0 итераций градиентного метода параллельно оптимизации. Число итераций t_0 выбиралось заранее экспериментально. Отдельно стоит заметить, что данный метод позволяет упорядочивать параметры в процессе оптимизации параметров модели. Также рассмотрены методы оптимального прореживания, метод основанный на вариационном подходе, а также метод основанный на методе Белсли для удаления зависимых параметров модели. Все данные методы позволяет задать полный порядок на множестве параметров моделей глубокого обучения.

Полный порядок на множестве параметров позволяет выбирать архитектуры нейросетевых моделей ученика. Выбранные архитектуры рассматриваются в качестве модели ученика в методах дистилляции.

Список литературы

- [1] *Alex Krizhevsky and Vinod Nair and Geoffrey Hinton* CIFAR-10 (Canadian Institute for Advanced Research) // <http://www.cs.toronto.edu/~kriz/cifar.html>
- [2] *Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.* Imagenet: A large-scale hierarchical image database // IEEE conference on computer vision and pattern recognition, 2009. P. 248–255.
- [3] *Huang, Zehao and Wang, Naiyan* Like What You Like: Knowledge Distill via Neuron Selectivity Transfer // arXiv e-prints, 2017.
- [4] *S. Aeberhard* Wine Data Set, 1991.
- [5] *Kui Ren and Tianhang Zheng and Zhan Qin and Xue Liu* Adversarial Attacks and Defenses in Deep Learning // Engineering, 2020. P. 346–360.
- [6] *Alex Krizhevsky, Ilya Sutskever, Geoffrey Hinton* ImageNet Classification with Deep Convolutional Neural Networks // NIPS, 2012.
- [7] *Karen Simonyan and Andrew Zisserman* Very Deep Convolutional Networks for Large-Scale Image Recognition // NIPS, 2014.
- [8] *Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I.* Attention Is All You Need // In Advances in Neural Information Processing Systems. 2017. V. 5. P. 6000–6010.
- [9] *Devlin J., Chang M., Lee K., Toutanova K.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // arXiv preprinted, 2018.
- [10] *Tom B. Brown et al* GPT3: Language Models are Few-Shot Learners // arXiv preprinted, 2020.
- [11] *Linting Xue and Noah Constant and Adam Roberts and Mihir Kale and Rami Al-Rfou and Aditya Siddhant and Aditya Barua and Colin Raffel.* mT5: A massively multilingual pre-trained text-to-text transformer // arXiv preprinted, 2021.
- [12] *Yang, Ziqing and Cui, Yiming and Chen, Zhipeng and Che, Wanxiang and Liu, Ting and Wang, Shijin and Hu, Guoping* TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2020. P. 9–16.

- [13] *He K., Zhang X., Ren S., Sun J.* Deep Residual Learning for Image Recognition // Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016. P. 770–778.
- [14] *Бахтееев О.Ю., Стрижсов В.В.* Выбор моделей глубокого обучения субоптимальной сложности // АиТ. 2018. № 8. С. 129–147.
- [15] *Hinton G., Vinyals O., Dean J.* Distilling the Knowledge in a Neural Network // NIPS Deep Learning and Representation Learning Workshop. 2015.
- [16] *LeCun Y., Cortes C., Burges C.* The MNIST dataset of handwritten digits, 1998. <http://yann.lecun.com/exdb/mnist/index.html>.
- [17] *Vapnik V., Izmailov R.* Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.
- [18] *Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V.* Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.
- [19] *Madala H., Ivakhnenko A.* Inductive Learning Algorithms for Complex Systems Modeling. Boca Raton: CRC Press Inc., 1994.
- [20] *Xiao H., Rasul K., Vollgraf R.* Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms // arXiv preprint arXiv:1708.07747. 2017.
- [21] *Wilson T., Kozareva Z., Nakov P., Rosenthal S., Stoyanov V., Ritter A.* SemEval-2013 Task 2: Sentiment Analysis in Twitter // Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, 2013. P. 312–320.
- [22] *LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W., Jackel L.* Backpropagation Applied to Handwritten Zip Code Recognition // Neural Computation. 1989. V. 1. No 4. P. 541–551.
- [23] *Hochreiter S., Schmidhuber J.* Long short-term memory // Neural Computation. 1997. V. 9. No 8. P. 1735–1780.
- [24] *Kingma D., Ba J.* Adam: A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980. 2014.
- [25] *Graves A.* Practical Variational Inference for Neural Networks // Advances in Neural Information Processing Systems, 2011. Vol. 24. P. 2348–2356.

- [26] *Vapnik V., Izmailov R.* Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.
- [27] *Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V.* Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.
- [28] *Sutskever I., Vinyals O., Le Q.* Sequence to Sequence Learning with Neural Networks // Advances in Neural Information Processing Systems, 2014. Vol. 2. P. 3104–3112.
- [29] *Li C., Chen C., Carlson D., Carin L.* Preconditioned Stochastic Gradient Langevin Dynamics for Deep Neural Networks // Thirtieth AAAI Conference on Artificial Intelligence. — Phoenix, USA, 2016. P. 1788–1794.
- [30] *Tibshirani R.* Regression shrinkage and selection via the Lasso // Journal of the Royal Statistical Society, 1996. Vol. 58. P. 267–288.
- [31] *Zou H., Hastie T.* Regularization and variable selection via the Elastic Net // Journal of the Royal Statistical Society, 2005. Vol. 67. P. 301–320.
- [32] *Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R.* Dropout: A Simple Way to Prevent Neural Networks from Overfitting // Journal of Machine Learning Research, 2014. Vol. 15. P. 1929–1958.
- [33] *Molchanov D., Ashukha A., Vetrov D.* Variational Dropout Sparsifies Deep Neural Networks // 34th International Conference on Machine Learning. — Sydney, Australia, 2017. Vol. 70. P. 2498–2507.
- [34] *LeCun Y., Denker J., Solla S.* Optimal Brain Damage // Advances in Neural Information Processing Systems, 1989. Vol. 2. P. 598–605.
- [35] *Грабовой А. Б., Бахтееев О. Ю., Стрижсов В. В.* Определение релевантности параметров нейросети // Информатика и ее применения, 2019. Т. 13. Вып. 2. С. 62–70.
- [36] *Грабовой А. Б., Бахтееев О. Ю., Стрижсов В. В.* Введение отношения порядка на множестве параметров аппроксимирующих моделей // Информатика и ее применения, 2019. Т. 14. Вып. 2. С. 58–65.
- [37] *Mandt S., Hoffman M., Blei D.* Stochastic Gradient Descent as Approximate Bayesian Inference // Journal Of Machine Learning Research, 2017. Vol. 18. P. 1–35.

- [38] *Kingma D., Ba L.* Adam: A Method for Stochastic Optimization // 3rd International Conference on Learning Representations. — San Diego, USA, 2015.
- [39] *Harrison D., Rubinfeld D.* Hedonic prices and the demand for clean air // Journal of Environmental Economics and Management, 1991. Vol. 5. P. 81–102.
- [40] *LeCun Y., Cortes C., Burges C.* The MNIST dataset of handwritten digits, 1998. <http://yann.lecun.com/exdb/mnist/index.html>
- [41] *MacLaurin D., Duvenaud D., Adams R.* Gradient-based Hyperparameter Optimization Through Reversible Learning // Proceedings of the 32th International Conference on Machine Learning, 2015. Vol. 37. P. 2113–2122.
- [42] *Luketina J., Berglund M., Raiko T., Greff K.* Scalable Gradient-based Tuning of Continuous Regularization Hyperparameters // Proceedings of the 33th International Conference on Machine Learning, 2016. Vol. 48. P. 2952–2960.
- [43] *Bishop C.* Pattern Recognition and Machine Learning, 2006. Pp. 396.
- [44] *Neychev R., Katrutsa A., Strijov V.* Robust selection of multicollinear features in forecasting // Factory Laboratory, 2016. Vol. 82. P. 68–74.
- [45] *LeCun Y., Denker J., Solla S.* Optimal Brain Damage // Advances in Neural Information Processing Systems, 1989. P. 598–605.
- [46] *Molchanov D., Ashukha A., Vetrov D.* Variational Dropout Sparsifies Deep Neural Networks // Proceedings of the 34th International Conference on Machine Learning, 2017. Vol. 70. P. 2498–2507.
- [47] *Neal A., Radford M.* Bayesian Learning for Neural Networks, 1995.
- [48] *Sutskever I., Vinyals O., Le Q.* Sequence to Sequence Learning with Neural Networks, 2014. Vol. 2. P. 3104–3112.
- [49] *Graves A.* Practical Variational Inference for Neural Networks, 2011. P. 2348–2356.
- [50] *Louizos C., Ullrich K., Welling M.* Bayesian Compression for Deep Learning, 2017. P. 3288–3298.
- [51] *J. R. Kwapisz, G. M. Weiss, S. A. Moore* Activity Recognition using Cell Phone Accelerometers // Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data, 2010. Vol. 12. P. 74–82.
- [52] *W. Wang, H. Liu, L. Yu, F. Sun* Activity Recognition using Cell Phone Accelerometers // Joint Conference on Neural Networks, 2014. P. 1185–1190.

- [53] *A. D. Ignatov, V. V. Strijov* Human activity recognition using quasiperiodic time series collected from a single tri-axial accelerometer. // Multimedial Tools and Applications, 2015.
- [54] *A. Olivares, J. Ramirez, J. M. Gorris, G. Olivares, M. Damas* Detection of (in)activity periods in human body motion using inertial sensors: A comparative study. // Sensors, 12(5):5791–5814, 2012.
- [55] *Y. G. Cinar and H. Mirisae* Period-aware content attention RNNs for time series forecasting with missing values // Neurocomputing, 2018. Vol. 312. P. 177–186.
- [56] *A. P. Motrenko, V. V. Strijov* Extracting fundamental periods to segment biomedical signals // Journal of Biomedical and Health Informatics, 2015, 20(6). P. 1466 - 1476.
- [57] *Y. P. Lukashin* Adaptive methods for short-term forecasting // Finansy and Statistik, 2003.
- [58] *И. П. Ивкин, М. П. Кузнецов* Алгоритм классификации временных рядов акселерометра по комбинированному признаковому описанию. // Машинное обучение и анализ данных, 2015.
- [59] *V. V. Strijov, A. M. Katrutsa* Stresses procedures for features selection algorithms. // Schemometrics and Intelligent Laboratory System, 2015.
- [60] *I. Borg, P. J. F. Groenen* Modern Multidimensional Scaling. — New York: Springer, 2005. 540 p.
- [61] *Д. Л. Данилова, А. А. Жигловский* Главные компоненты временных рядов: метод "Гусеница". — Санкт-Петербургский университет, 1997.
- [62] *Tianqi C., Carlos G.* XGBoost: A Scalable Tree Boosting System // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.
- [63] *Xi C., Hemant I.* Random Forests for Genomic Data Analysis // Genomics. 2012. Issues. 99. No. 6. P. 323–329.
- [64] *Esen Y.S., Wilson J., Gader P.D.* Twenty Years of Mixture of Experts // IEEE Transactions on Neural Networks and Learning Systems. 2012. Issues. 23. No. 8. P. 1177–1193.
- [65] *Rasmussen C.E., Ghahramani Z.* Infinite Mixtures of Gaussian Process Experts // Advances in Neural Information Processing Systems 14. 2002. P. 881–888.

- [66] *Shazeer N., Mirhoseini A., Maziarz K.* Outrageously large neural networks: the sparsely-gated mixture-of-experts layer // International Conference on Learning Representations. 2017.
- [67] *Jordan M. I.* Hierarchical mixtures of experts and the EM algorithm // Neural Comput. 1994. Vol. 6, No. 2. P. 181–214.
- [68] *Jordan M. I., Jacobs R. A.* Hierarchies of adaptive experts // In Advances in Neural Information Processing Systems. 1991. P. 985–992.
- [69] *Lima C., Coelho A., Zuben F. J.* Hybridizing mixtures of experts with support vector machines: Investigation into nonlinear dynamic systems identification // Inf. Sci. 2007. Vol. 177. No. 10. P. 2049–2074.
- [70] *Cao L.* Support vector machines experts for time series forecasting // Neurocomputing. 2003. Vol. 51. P. 321–339.
- [71] *Yumlu M. S., Gurgen F. S., Okay N.* Financial time series prediction using mixture of experts // In Proc. 18th Int. Symp. Comput. Inf. Sci. 2003. P. 553–560.
- [72] *Cheung Y. M., Leung W. M., Xu L.* Application of mixture of experts model to financial time series forecasting // On Proc. Int. Conf. Neural Netw. Signal Process. 1995. P. 1–4.
- [73] *Weigend A. S., Shi S.* Predicting daily probability distributions of S&P500 returns // J. Forecast. 2000. Vol. 19. No. 4. P. 375–392.
- [74] *Ebrahimpour R., Moradian M. R., Esmkhani A., Jafarlou F. M.* Recognition of Persian handwritten digits using characterization loci and mixture of experts // J. Digital Content Technol. Appl. 2009. Vol. 3. No. 3. P. 42–46.
- [75] *Estabrooks A., Japkowicz N.* A mixture-of-experts framework for text classification //In Proc. Workshop Comput. Natural Lang. Learn., Assoc. Comput. Linguist. 2001. P. 1–8.
- [76] *Mossavat S., Amft O., Petkov Vries B., Kleijn W.* A Bayesian hierarchical mixture of experts approach to estimate speech quality // In Proc. 2nd Int. Workshop Qual. Multimedia Exper. 2010. P. 200–205.
- [77] *Peng F., Jacobs R. A., Tanner M. A.* Bayesian inference in mixtures-of-experts and hierarchical mixtures-of-experts models with an application to speech recognition // J. Amer. Stat. Assoc. 1996. Vol. 91. No. 435. P. 953–960.
- [78] *Tuerk A.* The state based mixture of experts HMM with applications to the recognition of spontaneous speech. Ph.D. thesis. Cambridge: Univ. Cambridge, 2001.

- [79] Sminchisescu C., Kanaujia A., Metaxas D. Discriminative density propagation for visual tracking // IEEE Trans. Pattern Anal. Mach. Intell. 2007. Vol. 29. No. 11. P. 2030–2044.
- [80] Bowyer K., Hollingsworth K., Flynn P. A Survey of Iris Biometrics Research: 2008–2010.
- [81] Matveev I. Detection of iris in image by interrelated maxima of brightness gradient projections // Appl.Comput. Math. 2010. Vol. 9. No. 2. P. 252–257.
- [82] Matveev I., Simonenko I.. Detecting precise iris boundaries by circular shortest path method // Pattern Recognition and Image Analysis. 2014. Vol. 24. P. 304–309.
- [83] Dempster A. P., Laird N. M., Rubin D. B. Maximum Likelihood from Incomplete Data via the EM Algorithm // Journal of the Royal Statistical Society. Series B (Methodological). 1977. Vol. 39. No. 1 P. 1–38.
- [84] Bishop C. Pattern Recognition and Machine Learning. Berlin: Springer, 2006. P. 758.
- [85] Akhtar N, Mian A (2018) Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey. IEEE Access 6:14410–14430
- [86] Bishop C. (2010) Pattern Recognition and Machine Learning. Springer, Berlin
- [87] Bowyer K, Hollingsworth K, Flynn P (2010) A Survey of Iris Biometrics Research: 2008-2010. Handbook of iris recognition 15–54
- [88] Cao L (2003) Support vector machines experts for time series forecasting. Neurocomputing 51:321–339
- [89] Chen T, Guestrin C (2016) XGBoost: A Scalable Tree Boosting System. KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 785–794
- [90] Chen Xi, Ishwaran H (2012) Random Forests for Genomic Data Analysis. Genomics 6:323–329
- [91] Cheung Y, Leung W, Xu L (1995) Application of mixture of experts model to financial time series forecasting. In Proc. Int. Conf. Neural Netw. Signal Process. 1–4
- [92] Dempster A, Laird N, Rubin D (1997) Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society. Series B (Methodological) 39:1–38

- [93] Estabrooks A, Japkowicz N (2001) A mixture-of-experts framework for text classification. In Proc. Workshop Comput. Natural Lang. Learn., Assoc. Comput. Linguist. 1–8
- [94] Ebrahimpour R, Moradian R, Esmkhani A, Jafarlou F (2009) Recognition of Persian handwritten digits using characterization loci and mixture of experts. *J. Digital Content Technol. Appl.* 42–46
- [95] Han X, Yao M, Debayan D, Hui L, Ji-Liang T, Anil J (2020) Adversarial Attacks and Defenses in Images, Graphs and Text: A Review. *International Journal of Automation and Computing* 17:151–178
- [96] Kaiming H, Xiangyu Z, Shaoqing R, Jian S (2016) Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 770–778
- [97] Matveev I (2010) Detection of iris in image by interrelated maxima of brightness gradient projections. *Appl. Comput. Math* 9:252–257
- [98] Matveev I, Simonenko I (2014) Detecting precise iris boundaries by circular shortest path method. *Pattern Recognition and Image Analysis* 24:304–309
- [99] Mossavat S, Amft O, Vries B, Petkov P, Kleijn W (2010) A Bayesian hierarchical mixture of experts approach to estimate speech quality. In Proc. 2nd Int. Workshop Qual. Multimedia Exper. 200–205
- [100] Peng F, Jacobs R, Tanner M (1996) Bayesian inference in mixtures-of-experts and hierarchical mixtures-of-experts models with an application to speech recognition. *J. Amer. Stat. Assoc.* 91:953–960
- [101] Ribeiro M, Singh S, Guestrin C (2016) Why Should I Trust You?": Explaining the Predictions of Any Classifier. *KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 1135–1144
- [102] Salamani D, Gadatsch S, Golling T, Stewart G, Ghosh A, Rousseau D, Hasib A, Schaarschmidt J (2018) Deep Generative Models for Fast Shower Simulation in ATLAS. *2018 IEEE 14th International Conference on e-Science (e-Science)* <https://doi.org/10.1109/eScience.2018.00091>
- [103] Sminchisescu C, Kanaujia A, Metaxas D (2007) Discriminative density propagation for visual tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* 29:2030–2044
- [104] Tuerk A (2001) The state based mixture of experts HMM with applications to the recognition of spontaneous speech. Ph.D. thesis, University of Cambridge.

- [105] Weigend A, Shi S (2000) Predicting daily probability distributions of S&P500 returns. *J. Forecast* 19:375–392
- [106] Yuksel S, Wilson J, Gader P (2012) Twenty Years of Mixture of Experts. *IEEE Transactions on Neural Networks and Learning Systems* 8:1177–1193
- [107] Yumlu M, Gurgen F, Okay N (2003) Financial time series prediction using mixture of experts. In Proc. 18th Int. Symp. Comput. Inf. Sci. 553–560
- [108] Demidenko E (2007) Sample size determination for logistic regression revisited. *Statist. Med.* 26:3385–3397.
- [109] Harrison D, Rubinfeld D (1978) Hedonic prices and the demand for clean air. *Economics and Management* 5:81–102.
- [110] Joseph L, Berger R, Be'lisle P (1995) Bayesian and mixed bayesian likelihood criteria for sample size determination. *Statistician* 16:769–781.
- [111] Joseph L, Wolfson D, Berger R (1997) Sample size calculations for binomial proportions via highest posterior density intervals. *Statistical Medicine* 44:143–154.
- [112] Kloek T (1975). Note on a large-sample result in specification analysis. *Econometrica* 43:933–936.
- [113] Lindley D (1997) The choice of sample size. *The Statistician* 46:129–138.
- [114] Motrenko A, Strijov V, Weber G (2014) Sample size determination for logistic regression. *Journal of Computational and Applied Mathematics* 255:743–752.
- [115] Quinlan J (1992) Learning with continuous classes. *Proc. 5th Australian Joint Conference on AI* 343–348.
- [116] Qumsiyeh M (2013) Using the bootstrap for estimation the sample size in statistical experiments. *Journal of modern applied statistical methods* 8:305–321.
- [117] Rubin D, Stern H (1998) Sample size determination using posterior predictive distributions. *Sankhya: The Indian Journal of Statistics Special Issue on Bayesian Analysis* 60:161–175.
- [118] Self S, Mauritsen R (1988) Power/sample size calculations for generalized linear models. *Biometrics* 44:79–86.
- [119] Self S, Mauritsen R, Ohara J (1992) Power calculations for likelihood ratio tests in generalized linear models. *Biometrics* 48:31–39.

- [120] Shieh G (2000) On power and sample size calculations for likelihood ratio tests in generalized linear models. *Biometrics* 56:1192–1196.
- [121] Shieh G (2005) On power and sample size calculations for Wald tests in generalized linear models. *Journal of Statistical Planning and Inference* 128:43–59.
- [122] Wang F, Gelfand A (2002) A Simulation-based Approach to Bayesian Sample Size Determination for Performance under a Given Model and for Separating Models. *Statistical Science* 17:193–208.
- [123] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. Attention Is All You Need // In Advances in Neural Information Processing Systems. 2017. V. 5. P. 6000–6010.
- [124] Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // arXiv preprint arXiv:1810.04805. 2018.
- [125] He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016. P. 770–778.
- [126] Бахтееев О.Ю., Стрижов В.В. Выбор моделей глубокого обучения субоптимальной сложности // АиТ. 2018. № 8. С. 129–147.
- [127] Hinton G., Vinyals O., Dean J. Distilling the Knowledge in a Neural Network // NIPS Deep Learning and Representation Learning Workshop. 2015.
- [128] LeCun Y., Cortes C., Burges C. The MNIST dataset of handwritten digits, 1998. <http://yann.lecun.com/exdb/mnist/index.html>.
- [129] Vapnik V., Izmailov R. Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.
- [130] Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V. Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.
- [131] Madala H., Ivakhnenko A. Inductive Learning Algorithms for Complex Systems Modeling. Boca Raton: CRC Press Inc., 1994.
- [132] Xiao H., Rasul K., Vollgraf R. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms // arXiv preprint arXiv:1708.07747. 2017.

- [133] Wilson T., Kozareva Z., Nakov P., Rosenthal S., Stoyanov V., Ritter A. SemEval-2013 Task 2: Sentiment Analysis in Twitter // Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, 2013. P. 312–320.
- [134] LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W., Jackel L. Backpropagation Applied to Handwritten Zip Code Recognition // Neural Computation. 1989. V. 1. No 4. P. 541–551.
- [135] Hochreiter S., Schmidhuber J. Long short-term memory // Neural Computation. 1997. V. 9. No 8. P. 1735–1780.
- [136] Kingma D, Ba J. Adam: A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980. 2014.
- [137] Alex Krizhevsky and Vinod Nair and Geoffrey Hinton CIFAR-10 (Canadian Institute for Advanced Research) // <http://www.cs.toronto.edu/~kriz/cifar.html>
- [138] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L. Imagenet: A large-scale hierarchical image database // IEEE conference on computer vision and pattern recognition, 2009. P. 248–255.
- [139] Huang, Zehao and Wang, Naiyan Like What You Like: Knowledge Distill via Neuron Selectivity Transfer // arXiv e-prints, 2017.
- [140] Kui Ren and Tianhang Zheng and Zhan Qin and Xue Liu Adversarial Attacks and Defenses in Deep Learning // Engineering, 2020. P. 346–360.
- [141] Alex Krizhevsky, Ilya Sutskever, Geoffrey Hinton ImageNet Classification with Deep Convolutional Neural Networks // NIPS, 2012.
- [142] Karen Simonyan and Andrew Zisserman Very Deep Convolutional Networks for Large-Scale Image Recognition // NIPS, 2014.
- [143] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. Attention Is All You Need // In Advances in Neural Information Processing Systems. 2017. V. 5. P. 6000–6010.
- [144] Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // arXiv preprinted, 2018.
- [145] Tom B. Brown et al GPT3: Language Models are Few-Shot Learners // arXiv preprinted, 2020.

- [146] *Linting Xue and Noah Constant and Adam Roberts and Mihir Kale and Rami Al-Rfou and Aditya Siddhant and Aditya Barua and Colin Raffel.* mT5: A massively multilingual pre-trained text-to-text transformer // arXiv preprinted, 2021.
- [147] *Yang, Ziqing and Cui, Yiming and Chen, Zhipeng and Che, Wanxiang and Liu, Ting and Wang, Shijin and Hu, Guoping* TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2020. P. 9–16.
- [148] *He K., Zhang X., Ren S., Sun J.* Deep Residual Learning for Image Recognition // Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016. P. 770–778.
- [149] *Бахтееев О.Ю., Стрижсов В.В.* Выбор моделей глубокого обучения субоптимальной сложности // АиТ. 2018. № 8. С. 129–147.
- [150] *Hinton G., Vinyals O., Dean J.* Distilling the Knowledge in a Neural Network // NIPS Deep Learning and Representation Learning Workshop. 2015.
- [151] *LeCun Y., Cortes C., Burges C.* The MNIST dataset of handwritten digits, 1998. <http://yann.lecun.com/exdb/mnist/index.html>.
- [152] *Vapnik V., Izmailov R.* Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.
- [153] *Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V.* Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.
- [154] *Madala H., Ivakhnenko A.* Inductive Learning Algorithms for Complex Systems Modeling. Boca Raton: CRC Press Inc., 1994.
- [155] *Xiao H., Rasul K., Vollgraf R.* Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms // arXiv preprint arXiv:1708.07747. 2017.
- [156] *Wilson T., Kozareva Z., Nakov P., Rosenthal S., Stoyanov V., Ritter A.* SemEval-2013 Task 2: Sentiment Analysis in Twitter // Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, 2013. P. 312–320.
- [157] *LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W., Jackel L.* Backpropagation Applied to Handwritten Zip Code Recognition // Neural Computation. 1989. V. 1. No 4. P. 541–551.

- [158] *Hochreiter S., Schmidhuber J.* Long short-term memory // Neural Computation. 1997. V. 9. No 8. P. 1735–1780.
- [159] *Kingma D, Ba J.* Adam: A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980. 2014.
- [160] *Graves A.* Practical Variational Inference for Neural Networks // Advances in Neural Information Processing Systems, 2011. Vol. 24. P. 2348–2356.
- [161] *Grabovoy A. V., Bakhteev O. Y., Strijov V. V.* Estimation of relevance for neural network parameters // Informatics and Applications, 2019. Vol.13 No 2. P. 62–70.
- [162] *Vapnik V., Izmailov R.* Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.
- [163] *Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V.* Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.