Оптимизация гиперпараметров моделей глубокого обучения градиентными методами

О. Ю. Бахтеев[1], В. В. Стрижов[2]

Аннотация: Решается задача оптимизации гиперпараметров модели глубокого обучения. Для оптимизации гиперпараметров модели предлагаются алгоритмы, основанные на градиентном спуске. Так как сложность рассматриваемых алгоритмах сопоставима со сложностью оптимизации параметров модели, предлагается проводить оптимизацию параметров и гиперпараметров в единой процедуре. Для выбора адекватных значений гиперпараметров вводятся вероятностные предположения о распределении параметров. В качестве оптимизируемой функции выступает байесовское правдоподобие модели и кросс-валидация. Для получения оценки правдоподобия используются вариационные методы. Проводится вычислительный эксперимент на нескольких выборках.

Ключевые слова: градиентный спуск, стохастический поиск гиперпараметров, оценка гиперпараметров, выбор модели, глубокое обучение, классификация, регрессия.

1 Введение

В работе решается задача оптимизации гиперпараметров моделей глубокого обучения. Под *моделью* понимается суперпозиция функций, решающая задачу классификации или регрессии. Под *гиперпараметрами* модели понимается параметры распределения параметров модели.

Одна из проблем построения моделей глубокого обучения — большое число параметров модели [?], которое достигает нескольких миллионов, а оптимизация модели достигает десятков дней [?]. Задача выбора модели глубокого обучения включает в себя выбор стратегии построения модели, эффективной по вычислительным ресурсам. Проблема оптимизации параметров модели глубокого обучения является вычислительно сложной в силу невыпуклости оптимизируемой

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

[2]Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН, strijov@ccas.ru

функции потерь. Поэтому задача поиска параметров оптимизации является важной, и нахождение оптимальныых гиперпараметров сильно влияет на итоговое качество модели [?, ?].

В данной работе сравниваются градиентные методы оптимизации гиперпараметров. Основным достоинством подобных алгоритмов является их возможность одновременной оптимизации щначительного количества гиперпараметров. В качестве базового алгоритма выступает выбор гиперпараметров модели с использованием случайного поиска. В работах [?, ?, ?] в качестве целевой функции потерь рассматривается потеря на валидационной подвыборке с $L2$ регуляризацией. В данной работе рассматривается общая задачи оптимизации гиперпараметров. Рассматриваеые алгоритмы и целевые функции потерь реализованы и представлены в качестве библиотеки для оптимизации гиперпараметров моделей [?]. Основным теоретическим вкладом данной работы является анализ рассматриваемых алгоритмов оптимизации гиперпараметров при использовании функции потерь общего вида, а также исследование качества и устойчивости итоговых моделей в случае использования кросс-валидации и вариационной оценки правдоподобия. В экспериментальной части в качестве критерия выбора модели выступают вариационная нижняя оценка правдоподобия модели и ошибка на валидационной части выборки. В отличие от [?], где также производится сравнение алгоритмов оптимизации гиперпараметров, в данной работе исследуется поведение алгоритмов на выборках большой мощности, таких как WISDM [?] и MNIST [?]. Численные эксперименты показывают, что при значительном количестве гиперпараметров, сопоставимым с количеством параметров модели, рассматриваемые алгоритмы предпочтительнее стохастических.

Обзор литературы В работах [?, ?] предлагаются стратегии выбора гиперпараметров модели, основанные на случайном выборе параметров. Другим методом, представленным в литературе [?, ?], является обучение вероятностных моделей для предсказания гиперпараметров. В работе [?] отмечается, что данный метод нахождения оптимальных гиперпараметров является неэффективными в случае, когда число гиперпараметров велико.

В работах [?, ?, ?, ?, ?] предлагаются методы оптимизации гиперпараметров, основанные на градиентных алгоритмах оптимизации: восстанавливается вся история изменения параметров в ходе оптимизации, в качестве функции для оптимизации гиперпараметров рассматривается функция потерь от конечного значения параметров, которое выражается через начальное значение параметров. Данная процедура является неэффективной по памяти, т.к. для хранения всей истории оптимизации параметров требуется большое количество памяти. В работе [?] предлагается жадный вариант градиентной оптимизации гиперпараметров. В работе [?] рассматривается оптимизация параметров с моментом, позволяющая эффективно хранить историю параметров в памяти. В работе [?] предлагается метод, рассматривающий траекторию оптимизации параметров как линейную, что также

Алгоритм	**Тип алгоритма**	**Преимущества алгоритма**	**Недостатки алгоритма**
Случайный поиск	стохастический	простота реализации	алгоритм неэффективен при большом количестве гиперпараметров (проклятие размерности)
Жадный алгоритм [?]	градиентный	Возможность одновременной оптимизации параметров и гиперпараметров	Жадность алгоритма
HOAG [?]	градиентный	Быстрая сходимость	Алгоритм требователен к настройкам параметров
DrMAD [?]	градиентный	Алгоритм учитывает алгоритм оптимизации параметров модели и его параметры	Алгоритм страдает от проблем неустойчиовтси градиентного спуска (градиентный взрыв и затухание); Алгоритм работает в очень жетских предположениях.

Table 1: Преимщуества и недостатки рассматриваемых алгоритмов

позволяет эффективно хранить историю параметров. В работе [?] рассматривается аппроксимация градиента оптимизируемой функции.

Для решения заданной рассматриваемой задачи требуется выбрать критерий выбора модели [?, ?]. В качестве критерия выбора модели в ряде работ [?, ?, ?, ?, ?] выступает правдоподобие модели. В работах [?,?,?] рассматривается проблема выбора модели и оценки гиперапараметров в задачах регрессии. Одним из методов получения приближенного значения интеграла правдоподобия является вариационный метод получения нижней оценки интеграла [?]. В работе [?] рассматривается стохастическая версия вариационного метода. В работе [?] рассматривается взаимосвязь градиентных методов получения вариационной нижней оценки интеграла с методом Монте-Карло. Альтернативным критерием выбора модели является минимальная длина описания [?], являющаяся показателем статистической сложности модели и заданной выборки. В работе [?] рассматривается алгоритм получения вариационной нижней оценки правдоподобия для оптимизации гиперпараметров моделей глубокого обучения, проводится связь между правдоподобием модели и минимальной длиной описания.

Альтернативным методом выбора модели является выбор модели на основе скользящего контроля [?, ?]. Проблемой такого подхода является возможная высокая вычислительная сложность [?,?]. В работах [?,?] рассматривается проблема смещения оценок качества модели и гиперпараметров, получаемых при использовании k-fold метода скользящего контроля, при котором выборка делится на k-частей с обучением на $k-1$ части и валидацией результата на оставшейся части выборки.

2 Постановка задачи

Задана выборка

$$\mathfrak{D} = \{(\mathbf{x}_i, y_i)\}, i = 1, \ldots, m, \tag{1}$$

состоящая из множества пар «объект-метка»,

$$\mathbf{x}_i \in \mathbf{X} \subset \mathbb{R}^n, \quad y_i \in \mathbf{y} \subset \mathbb{Y},$$

где $\mathbf{X}$ — матрица объектов, $\mathbf{y}$ — вектор меток зависимой переменной y. Метка y объекта $\mathbf{x}$ принадлежит либо конечному множеству: $y \in \mathbb{Y} = \{1, \ldots, Z\}$ в случае задачи классификации, где Z — число классов, либо некоторому подмножеству вещественных чисел $y \in \mathbb{R}$ в случае задачи регрессии.

Задана дифференцируемая по параметрам модель, приближающая зависимую переменную y:

$$f : \mathbb{R}^n \to \mathbb{Y}, \quad \mathbf{w} \in \mathbb{R}^u.$$

Рассмотрим модель f и ее вероятностную интерпретацию для случая задач регрессии и классификации.

Регрессия. Положим, что зависимая переменная распределена нормально:

$$\mathbf{y} = \mathcal{N}\big(\mathbf{f}, \mathbf{I}\big), \tag{2}$$

где $\mathbf{f} = \mathbf{f}(\mathbf{w}, \mathbf{X})$.

Определим правдоподобие выборки $p(\mathbf{y}|\mathbf{X}, \mathbf{w})$:

$$\log p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = -\frac{m}{2}\log(2\pi) - \frac{1}{2}(\mathbf{y} - \mathbf{f}(\mathbf{w}, \mathbf{X}))^{\mathrm{T}}\mathbf{I}(\mathbf{y} - \mathbf{f}(\mathbf{w}, \mathbf{X})).$$

Классификация. В случае двуклассовой классификации положим, что зависимая переменная распределена биномиально:

$$\mathbf{y} = \mathcal{B}\big(1 - \mathbf{f}, \mathbf{f}\big), \tag{3}$$

где вектор-функция $\mathbf{f}$ задает вероятность принадлежности объектов $\mathbf{X}$ к первому классу. В случае многоклассовой классификации зависимая переменная распределена мультиномиально, r-я компонента $\mathbf{f}$ задает вероятность принадлежности классу r. Тогда правдоподобие выборки задается как

$$\log p(\mathbf{y}|\mathbf{X}, \mathbf{w}) = \sum_{\mathbf{x}, y \in \mathbf{X}, \mathbf{x}} \sum_{r=1}^{Z} [y = r] \log f_r(\mathbf{w}, \mathbf{x}),$$

где f_r — r-я компонента функции $\mathbf{f}$.

Для задач классификации (**??**) и регрессии (**??**) задано параметрическое априорное распределение $p(\mathbf{w}|\mathbf{A})$ вида:

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{A}^{-1}), \tag{4}$$

где $\mathbf{A}^{-1} = \mathrm{diag}[\alpha_1, \ldots, \alpha_u]^{-1}$ — матрица ковариаций диагонального вида. Гипотезы (**??**),(**??**),(**??**) не противоречат друг другу в силу неограниченности нормального распределения [**?**].

Задача оптимизации гиперпараметров зависит как от критерия выбора модели, так и от метода оптимизации параметров модели. Проиллюстрируем задачу оптимизации гиперпараметров *двусвязным байесовским выводом*. Для дальнейшей формализации задачи в общем виде введем переобозначение:

$$\boldsymbol{\theta} = \mathbf{w}, \quad \mathbf{h} = [\alpha_1, \ldots, \alpha_u], \tag{5}$$

где $\boldsymbol{\theta}$ — множество оптимизируемых параметров модели, $\mathbf{h}$ — множество гиперпараметров модели.

На *первом уровне* байесовского вывода производится оптимизация параметров модели f по заданной выборке $\mathfrak{D}$:

$$\hat{\boldsymbol{\theta}} = \arg\max\big(-L(\boldsymbol{\theta}, \mathbf{h})\big) = p(\mathbf{w}|\mathbf{X}, \mathbf{y}, \mathbf{A}) = \frac{p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w}|\mathbf{A})}{p(\mathbf{y}|\mathbf{X}, \mathbf{A})}. \tag{6}$$

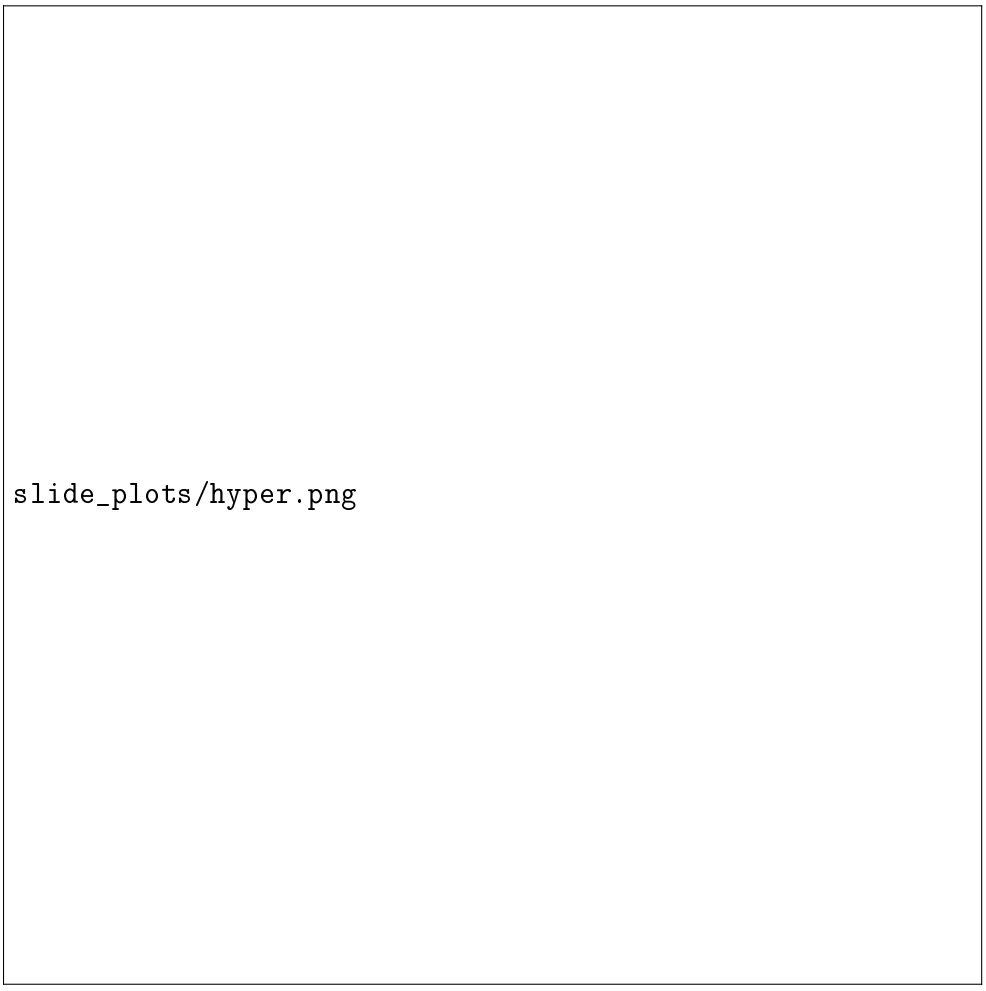

Figure 1: Зависимость правдодобия модели от значения гиперпараметра α. TODO: переделать

На *втором уровне* производится оптимизация апостериорного распределения гиперпараметров $\mathbf{h}$:

$$p(\mathbf{A}|\mathbf{X}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{X}, \mathbf{A})p(\mathbf{A}),$$

где знак «$\propto$» означает равенство с точностью до нормирующего множителя.

Полагая распределение параметров $p(\mathbf{A})$ равномерным на некоторой большой окрестности, получим задачу оптимизации гиперпараметров:

$$Q(\boldsymbol{\theta}, \mathbf{h}) = p(\mathbf{y}|\mathbf{X}, \mathbf{A}) = \int_{\mathbf{w}\in\mathbb{R}^u} p(\mathbf{y}|\mathbf{X}, \mathbf{w})p(\mathbf{w}|\mathbf{A}) \to \max_{[\alpha_1,\ldots,\alpha_u]\in\mathbb{R}^n}. \tag{7}$$

Сформулируем задачу оптимизации гиперпараметров в общем виде. Обозначим за $\mathbf{h} \in \mathbb{R}^h$ вектор гиперпараметров модели (**??**). Обозначим за $\boldsymbol{\theta} \in \mathbb{R}^s$ множество всех оптимизируемых параметров (**??**). Пусть задана дифференцируемая функция потерь $L(\boldsymbol{\theta}, \mathbf{h})$, по которой производится оптимизация функции f (**??**). Пусть также задана

дифференцируемая функция $Q(\boldsymbol{\theta}, \mathbf{h})$, определяющая итоговое качество модели f и приближающая интеграл (**??**).

Требуется найти параметры $\hat{\boldsymbol{\theta}}$ и гиперпараметры $\hat{\mathbf{h}}$ модели, доставляющие минимум следующему функционалу:

$$\hat{\mathbf{h}} = \arg\max_{\mathbf{h} \in \mathbb{R}^h} Q(\hat{\boldsymbol{\theta}}(\mathbf{h}), \mathbf{h}), \tag{8}$$

$$\hat{\boldsymbol{\theta}}(\mathbf{h}) = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^s} L(\boldsymbol{\theta}, \mathbf{h}). \tag{9}$$

Рассмотрим вид переменной $\boldsymbol{\theta}$ и функций L, Q для различных методов выбора модели и оптимизации ее параметров.

Базовый метод Пусть оптимизация параметров и гиперпараметров производится по всей выборке $\mathfrak{D}$ по одной и той же функции:

$$L(\boldsymbol{\theta}, \mathbf{h}) = Q(\boldsymbol{\theta}) = \log p(\mathbf{y}, \mathbf{w} | \mathbf{X}, \mathbf{A}) = \log p(\mathbf{y} | \mathbf{X}, \mathbf{w}) + \log p(\mathbf{w} | \mathbf{A})$$

.

Вспомогательная переменная $\boldsymbol{\theta}$, по которой производится оптимизация модели f, соответствует параметрам модели:

$$\boldsymbol{\theta} = \mathbf{w}.$$

Кросс-валидация Разобьем выборку $\mathfrak{D}$ на k равных частей:

$$\mathfrak{D} = \mathfrak{D}_1 \sqcup \cdots \sqcup \mathfrak{D}_k.$$

Запустим k оптимизаций модели, каждую на своей части выборки. Положим $\boldsymbol{\theta} = [\mathbf{w}_1, \ldots, \mathbf{w}_k]$, где $\mathbf{w}_1, \ldots, \mathbf{w}_k$ — параметры модели при оптимизации k.

Положим функцию L равной среднему значению минус логарифма апостериорной вероятности по всем $k-1$ разбиениям $\mathfrak{D}$:

$$L(\boldsymbol{\theta}, \mathbf{h}) = -\frac{1}{k} \sum_{q=1}^{k} \Big(\frac{k}{k-1} \log p(\mathbf{y} \setminus \mathbf{y}_q | \mathbf{X} \setminus \mathbf{X}_q, \mathbf{w}_q) + \log p(\mathbf{w}_q | \mathbf{A}) \Big). \tag{10}$$

Положим функцию Q равной среднему значению правдоподобия выборки по частям выборки $\mathfrak{D}_q$, на которых не проходила оптимизация параметров:

$$Q(\boldsymbol{\theta}, \mathbf{h}) = \frac{1}{k} \sum_{q=1}^{k} k \log p(\mathbf{y}_q | \mathbf{X}_q, \mathbf{w}_q).$$

Вариационная оценка правдоподобия Положим $L = -Q$, равной вариационной оценке правдоподобия модели:

$$\log p(\mathbf{y}|\mathbf{X}, \mathbf{A}) \geq -\mathrm{D}_{\mathrm{KL}}\big(q(\mathbf{w})||p(\mathbf{w}|\mathbf{A})\big) + \int_{\mathbf{w}} q(\mathbf{w}) \log p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \mathbf{A}) d\mathbf{w} \approx \tag{11}$$

$$\approx \sum_{i=1}^{m} \log p(y_i|\mathbf{x}_i, \mathbf{w}_i) - D_{\mathrm{KL}}\big(q(\mathbf{w})||p(\mathbf{w}|\mathbf{A})\big) = -L(\boldsymbol{\theta}, \mathbf{h}) = Q(\boldsymbol{\theta}),$$

где q — нормальное распределение с диагональной матрицей ковариаций:

$$q \sim \mathcal{N}(\boldsymbol{\mu}_q, \mathbf{A}_q^{-1}), \tag{12}$$

где $\mathbf{A}_q = \mathrm{diag}[\alpha_1^q, \ldots, \alpha_u^q]^{-1}$ — диагональная матрица ковариаций, $\boldsymbol{\mu}_q$ — вектор средних компонент. Расстояние D_{KL} между двумя гауссовыми величинами задается как

$$D_{\mathrm{KL}}\big(q(\mathbf{w})||p(\mathbf{w}|\mathbf{f})\big) = \frac{1}{2}\big(\mathrm{Tr}[\mathbf{A}\mathbf{A}_q^{-1}] + (\boldsymbol{\mu} - \boldsymbol{\mu}_q)^{\mathsf{T}}\mathbf{A}(\boldsymbol{\mu} - \boldsymbol{\mu}_q) - u + \ln\,|\mathbf{A}^{-1}| - \ln\,|\mathbf{A}_q^{-1}|\big).$$

В качестве оптимизируемых параметров $\boldsymbol{\theta}$ выступают параметры распределения q:

$$\boldsymbol{\theta} = [\alpha_1, \ldots, \alpha_u, \mu_1, \ldots, \mu_u].$$

3 Градиентные методы оптимизации гиперпараметров

Рассмотрим случай, когда оптимизация (**??**) параметров $\boldsymbol{\theta}$ производится с использованием градиентных методов.

Определение. Назовем оператором оптимизации алгоритм T выбора вектора параметров $\boldsymbol{\theta}'$ по параметрам предыдущего шага $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}' = T(\boldsymbol{\theta}, \mathbf{h}).$$

Рассмотрим оператор градиентного спуска, производящий η шагов оптимизации:

$$\hat{\boldsymbol{\theta}} = T \circ T \circ \cdots \circ T(\boldsymbol{\theta}_0, \mathbf{h}) = T^{\eta}(\boldsymbol{\theta}_0, \mathbf{h}), \tag{13}$$

где

$$T(\boldsymbol{\theta}, \mathbf{h}) = \boldsymbol{\theta} - \gamma \nabla L(\boldsymbol{\theta}, \mathbf{h}),$$

γ — длина шага градиентного спуска, $\boldsymbol{\theta}_0$ — начальное значение параметров $\boldsymbol{\theta}$. В данной работе в качестве опреатора оптимизации параметров модели выступает стохастический градиентный спуск:

$$T(\boldsymbol{\theta}, \mathbf{h})_{\mathrm{SGD}} = \boldsymbol{\theta} - \gamma \nabla L(\boldsymbol{\theta}, \mathbf{h})|_{\mathfrak{D}=\hat{\mathfrak{D}}},$$

где $\hat{\mathfrak{D}}$ — случайная подвыборка исходной выборки $\mathfrak{D}$.

Перепишем задачу оптимизации (??), (??) в следующем виде

$$\hat{\mathbf{h}} = \arg\max_{\mathbf{h}\in\mathbb{R}^h} Q(T^\eta(\boldsymbol{\theta}_0, \mathbf{h})), \tag{14}$$

где $\boldsymbol{\theta}_0$ — начальное значение параметров $\boldsymbol{\theta}$.

Оптимизационную задачу (??) предлагается решать с использованием градиентного спуска. Вычисление градиента от функции $Q(T^\eta(\boldsymbol{\theta}_0, \mathbf{h}))$ по гиперпараметрам $\mathbf{h}$ является вычислительно сложным в силу внутренней процедуры оптимизации $T(\boldsymbol{\theta}_0, \mathbf{h})$. Общая схема оптимизации гиперпараметров представлена следующим образом:

1. От 1 до l:
2. Инициализировать парметры $\boldsymbol{\theta}$ при условии гиперпараметров $\mathbf{h}$.
3. Приближенно решить задачу оптимизации (??) и получить новый вектор параметров $\mathbf{h}'$
4. $\mathbf{h} = \mathbf{h}'$.

где l — количество итераций оптимизации гиперпараметров. Рассмотрим методы приближенного решения данной задачи оптимизации.

Жадный алгоритм В качестве правила обновления вектора гиперпараметров $\mathbf{h}$ на каждом шаге оптимизации (??) выступает градиентный спуск с учетом обновления параметров $\boldsymbol{\theta}$ на данном шаге:

$$\mathbf{h}' = \mathbf{h} - \gamma_{\mathbf{h}} \nabla_{\mathbf{h}} Q\big(T(\boldsymbol{\theta}, \mathbf{h}), \mathbf{h}\big) = \mathbf{h} - \gamma_{\mathbf{h}} \nabla_{\mathbf{h}} Q\big(\boldsymbol{\theta} - \gamma \nabla L(\boldsymbol{\theta}, \mathbf{h}), \mathbf{h}\big)\big),$$

где $\gamma_{\mathbf{h}}$ — длина шага оптимизации гиперпараметров.

Алгоритм HOAG Предлагается получить приближенное значения градиента гиперпараметров $\nabla_{\mathbf{h}} Q\big(T^\eta(\boldsymbol{\theta}_0, \mathbf{h})\big)$ на основе следующей формулы:

$$\nabla_{\mathbf{h}} Q\big(T^\eta(\boldsymbol{\theta}_0, \mathbf{h})\big) = \nabla_{\mathbf{h}} Q(\boldsymbol{\theta}, \mathbf{h}) - (\nabla^2_{\boldsymbol{\theta},\mathbf{h}} L(\boldsymbol{\theta}, \mathbf{h}))^{\mathrm{T}} \mathbf{H}(\boldsymbol{\theta})^{-1} \nabla_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \mathbf{h}),$$

где $\mathbf{H}$ — гессиан функции L по параметрам $\boldsymbol{\theta}$.

Процедура получения приближенного значения градиента гиперпараметров $\nabla_{\mathbf{h}} Q$ производится итеративно:

1. Провести η шагов оптимизации: $\boldsymbol{\theta} = T(\boldsymbol{\theta}_0, \mathbf{h})$.
2. Решить линейную систему для вектора $\boldsymbol{\lambda}$: $\mathbf{H}(\boldsymbol{\theta})\boldsymbol{\lambda} = \nabla_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \mathbf{h})$.
3. Приближенное значение градиентов гиперпараметра вычисляется как: $\hat{\nabla}_{\mathbf{h}} Q = \nabla_{\mathbf{h}} Q(\boldsymbol{\theta}, \mathbf{h}) - \nabla_{\boldsymbol{\theta},\mathbf{h}} L(\boldsymbol{\theta}, \mathbf{h})^T \boldsymbol{\lambda}$.

Итоговое правило обновления:

$$\mathbf{h}' = \mathbf{h} - \gamma_{\mathbf{h}} \hat{\nabla}_{\mathbf{h}} Q. \tag{15}$$

В данной работе для приближенного решения шага 2 алгоритма HOAG используется стохастический градиентный спуск в силу сложности вычисления гессиана $\mathbf{H}(\boldsymbol{\theta})$.

Алгоритм DrMad Для получения градиента от оптимизируемой функции Q как от функции от начальных параметров $\boldsymbol{\theta}_0$ предлагаетя пошагово восстановить η шагов оптимизации $T(\boldsymbol{\theta}_0)$ в обратном порядке аналогично методу обратного распространения ошибок. Для упрощения данной процедуры вводится предположение,что траектория изменения параметров $\boldsymbol{\theta}$ линейна:

$$\boldsymbol{\theta}^{\tau} = \boldsymbol{\theta}_0 + \frac{\tau}{\eta} T(\boldsymbol{\theta}). \tag{16}$$

Алгоритм вычисления приближенного значения градиента $\nabla \mathbf{h}$ является частным случаем алгоритма обратного распространения ошибки и представим в следующем виде:

1. Провести η шагов оптимизации: $\boldsymbol{\theta} = T(\boldsymbol{\theta}_0, \mathbf{h})$.
2. Положим $\hat{\nabla}\mathbf{h} = \nabla_{\mathbf{h}} Q(\boldsymbol{\theta}, \mathbf{h})$.
3. Положим $d\mathbf{v} = \mathbf{0}$.
4. Для $\tau = \eta \dots 1$ повторить:
5. Вычислить значения параметров $\boldsymbol{\theta}^{\tau}$(**??**).
6. $d\mathbf{v} = \gamma \hat{\nabla}_{\boldsymbol{\theta}}$.
7. $\hat{\nabla}\mathbf{h} = \hat{\nabla}\mathbf{h} - d\mathbf{v} \nabla_{\mathbf{h}} \nabla_{\boldsymbol{\theta}} Q$.
8. $\hat{\nabla}\boldsymbol{\theta} = \hat{\nabla}\boldsymbol{\theta} - d\mathbf{v} \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} Q$.

Итоговое правило обновления гиперпараметров аналогично (**??**). В работе [**?**] отмечается неустойчивость алгоритма при высоких значениях длины шага градиентного спуска γ. Поэтому вместо исходного правила (**??**) в данной работе первые 5% значений параметров не рассматриваются, а также учитывается только каждый τ_k шаг оптимизации:

$$\boldsymbol{\theta}^{\tau} = \boldsymbol{\theta}_{\tau_0} + \frac{\tau}{\eta} T(\boldsymbol{\theta}), \quad \tau \in \{\tau_0, \dots, \eta\}, \tau \bmod \tau_k = 0, \tag{17}$$

где $\tau_0 = [0.05 \cdot \eta]$.

4 Вычислительный эксперимент

Для анализа рассматриваемых алгоритомв оптимизации гиперпараметров был проведен ряд вычислительных экспериментов на выборках MNIST [**?**], WISDM [**?**], а также на синтетических данных.

Рассматривались следующие критерии качества:

1. Наилучшее значение $\hat{Q} = \max_{j \in \{1,\dots,l\}} Q^j$.

2. Среднее число итераций алгоритма для сходимости. Под данным показателем понимается число шагов оптимизациии гиперпараметров, при котором ошибка Q изменяется не более чем на 1% от своего наилучшего значения:

$$\arg\min_j : \frac{Q^j - Q^0}{\hat{Q} - Q^0} \geq 0.99,$$

   где Q^0 — значение функции Q до начала оптимизации гиперпараметров.

3. Внешний критерий качества моделей E:

$$E = \text{RMSE} = \left(\frac{1}{m} \sum_1^m (f(\mathbf{x}_i, \mathbf{w}) - y_i) \right)^{\frac{1}{2}}$$

   в случае задачи регрессии,

$$E = \text{Accuracy} = 1 - \frac{1}{m} \sum_1^m [f(\mathbf{x}_i, \mathbf{w}) \neq y_i]$$

   в случае задачи классификации.

4. Внешний критерий качества моделей E_σ при возмущении параметров модели:

$$E_\sigma = \text{RMSE}_\sigma = \left(\frac{1}{m} \sum_1^m (f(\mathbf{x}_i, \mathbf{w} + \boldsymbol{\varepsilon} - y_i) \right)^{\frac{1}{2}}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}\big(\mathbf{0}, \sigma \mathbf{I}\big).$$

В качестве улучшаемого алгоритма рассматривался случайный поиск параметров с количеством итераций поиска, совпадающих с количеством итераций оптимизации гиперпараметров l: $l = 50$ для синтетической выборки и выборки WISDM, $l = 25$ для выборки MNIST. Рассмаитрваемые алгоритмы представлены в Табл. **??**. Пример поведения траекторий параметров под действием алгоритмов приведен на Рис. **??**. В качестве функций Q и L рассматривались функции кросс-валидации (**??**) с $k = 4$ и вариацонной оценки правдоподобия (**??**).

На всех выборках гиперпараметры инициализировались случайно из равномерного распределения:

$$\mathbf{h} \sim \mathcal{U}(a, b)^h,$$

Алгоритм	**Тип алгоритма**	**Сложность работы одной итерации**	**Предположения для корректности**
Случайный поиск	стохастический	$O(\eta s\|\hat{\mathfrak{D}}\|)$	-
Жадный алгоритм [?]	градиентный	$O(\eta(s+h)\|\hat{\mathfrak{D}}\|)$	$\mathbf{H}(\boldsymbol{\theta}) = \mathbf{I}$
HOAG [?]	градиентный	$O(\eta s\|\hat{\mathfrak{D}}\| + h^2\|\hat{\mathfrak{D}}\| + o)$, где o — время решения уравнения пункте 3	первые производные Q и вторые производные L — липшецевы; $\det\mathbf{H} \neq \mathbf{0}$;
DrMAD [?]	градиентный	$O(\eta s\|\hat{\mathfrak{D}}\|)$	Траектория оптимизации параметров $\boldsymbol{\theta} = \boldsymbol{\theta}_0, \dots \boldsymbol{\theta}_\eta$ — линейная

Table 2: Основные свойства рассматриваемых алгоритмов

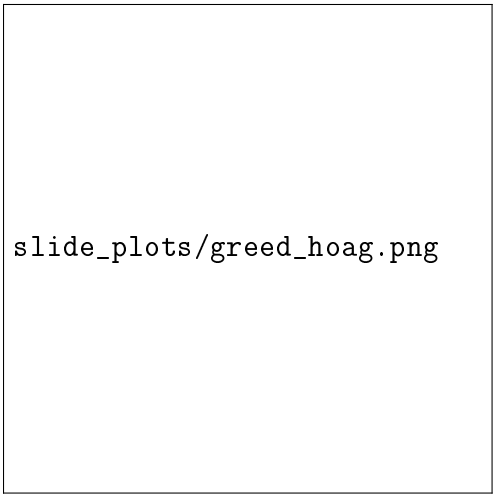

Figure 2: Графики траекторий параметров для разных алгоритмов. TODO: переделать

где $a = -2, b = 10$ для синтетической выборки и $a = -4, b = 10$ для выборок WISDM и MNIST.

Длина градиентного шага $\gamma_{\mathbf{h}}$ подбиралась для каждого алгоритма из сетки значений вида $\{r \cdot 10^s, s \leq 1, r \in \{1, 25, 50, 75\}\}$ таким образом, чтобы итоговое значение гиперпараметров $\mathbf{h}$ удовлетворяло следующему правилу:

$$a_{\min} \leq \min(\mathbf{h}), \quad \max(\mathbf{h}) \leq b_{\max},$$

где $a_{\min} = -2.5, b_{\max} = 10.5$ для синтетической выборки и $a_{\min} = -5, b_{\max} = 11$ для для выборок WISDM и MNIST. Калибровка значения γ проводилась на небольшом количестве итераций оптимизаций гиперпараметров l: $l = 50$ для синтетической выборки, $l = 10$ для выборки WISDM $l = 5$ для выборки MNIST. В случае, если алгоритмы показывали неустойчивую работу непосредственно во время запуска эксперимента (взрыв градиента или численное переполнение), то длина шага $\gamma_{\mathbf{h}}$ понижалась. Для алгоритма DrMad параметр τ_k, отвечающий за количество рассматриваемых шагов оптимизации был установлен как $\tau_k = 1$ для синтетической выборки и выборки WISDM, $\tau_k = 10$ для выборки MNIST.

Синтетическая выборка Синтетические данные были порождены по следующему правилу:

$$\mathbf{y} = \mathbf{X} + \boldsymbol{\varepsilon}, \quad \mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

где $m = 40$, $n = 1$. В качестве модели $\mathbf{f}$ выступает регрессия с признаками $\{\mathbf{X}^0, \ldots, \mathbf{X}^9, \mathbf{sin}(\mathbf{X}), \mathbf{cos}(\mathbf{X})\}$.

Было проведено 5 запусков для каждого алгоритма. Графики итоговых полиномов представлены на Рис. **??**. Как видно из графиков, с использованием вариационной оценки удалось получить полиномы, близкие к линейным моделям. Подобные модели показывают наилучшее правдоподобие в силу слабого переобучения и хорошего качества на тестовой выборке.

WISDM Выборка WISDM состоит из набора записей акселерометра. Каждой записи соответствуют три координаты по осям акселерометра. В качестве набора объектов рассматривалось наборы из 199 последовательных записей акселерометра. В качестве набора меток рассматривалась евклидовая норма соответствующих 200-х записей акселерометра.

Рассматривалась нейросеть с 10 нейронами на скрытом слое:

$$\mathbf{f} = \mathbf{W}_2 \cdot \mathbf{RELU}(\mathbf{W}_1\mathbf{X} + \mathbf{b}_1) + \mathbf{b}_2,$$

где $\mathbf{W}_1, \mathbf{b}_1$ — параметры первого слоя нейросети, $\mathbf{W}_2, \mathbf{b}_2$ — параметры второго слоя нейросети,

$$\mathbf{RELU}(\mathbf{x}) = \max(\mathbf{0}, \mathbf{x}).$$

Графики сходимости алгоритмов, а также качества полученных моделей представлены на Рис. **??**, **??**. Как видно из графиков, градиентные алгоритмы

DrMad и HOAG показывают значительно худший результат по сравнению с жадным алгоритмом оптимизации. Случайный поиск показыват достаточно хорошие результаты в случае небольшого числа оптимизируемых гиперпараметров $\mathbf{h}$. В случае, когда в качестве функции Q используется вариационная нижняя оценка правдоподобия (??) и количество гиперпараметров велико, эффективно работающими алгоритмами оказалась жадная оптимизация и HOAG. HOAG имеет большее время сходимости и требует более сложных вычислений в процессе оптимизации.

MNIST Выборка MNIST состоит из множества изображений рукописных цифр. Рассматривалась нейросеть с 300 нейронами на скрытом слое.

Графики сходимости алгоритмов, а также качества полученных моделей представлены на Рис. ??, ??, ??, ??. Как видно из графиков, модели, достигающиие наилучшей оценки правдоподобия, имеют наихудшее итоговое качество, но более устойчивы к возмущению параметров модели. Для дополнительного анализа данной проблемы были проведены эксперименты по оптимизации моделей на выборке с добавленным шумом с использованием значений гиперпараметров $\mathbf{h}$, полученных ранее:

$$\hat{\mathfrak{D}} = \mathfrak{D} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \hat{\sigma}\mathbf{I}),$$

где $\hat{\sigma}$ варьировалась в отрезке от 0 до 0.5. График зависимости качества моделей от значения $\hat{\sigma}$ приведен на ... Гиперпараметры, достигающие наибольших значений вариационной оценки (??) менее подвержены шуму в обучающей выборке, что можно интерпретировать как меньшую подверженность к переобучению.

Как можно видеть по результатам экспериментов, градиентные методы показывают лучший результат, чем случай поиск в случае большого количество гиперпараметров. Наилучшие результаты были получены жадным поиском. Алгоритм DrMad, показавший результаты хуже, чем жадный алгоритм и HOAG, является упрощенной версией алгоритма, представленного в [?]. Данный алгоритм позволяет проводить оптимизацию не только гиперпараметров, но параметров алгоритма оптимизации T. Поэтому возможным развитием метода DrMad является получение оптимальных значений параметров оптимизации.

5 Discussion

- HOAG и жадный работаю приблизительно одинаково
- Несмотря на близкую ассимптотику, HOAG предпочтитетелен, если сама внутренняя оптимизация является дорогой
- С другой стороны - HOAG требует мета-настройки, а жадный алгоритм можно использовать по расписанию
- DrMAD показывает неустойчивые результаты.

Алгоритм	L, Q	$Q(\boldsymbol{\theta}, \mathbf{h})$	Сходимость	E	$E_{0.25}$	$E_{0.5}$
Синтетическая выборка						
Случайный поиск	(??)	**-171.6**	**26.2 ± 20.0**	**1.367**	?	?
Жадная оптимизация	(??)	-172.5	30.0 ± 24.5	1.421	?	?
DrMAD	(??)	-174.1	40.2 ± 16.1	1.403	?	?
HOAG	(??)	-174.7	29.4 ± 24.0	**1.432**	?	?
Случайный поиск	(??)	-63.5	32.4 ± 18.7	1.368	?	?
Жадная оптимизация	(??)	-25.5	**1.2 ± 0.4**	1.161	?	?
DrMAD	(??)	**-25.1**	10.6 ± 0.8	1.157	?	?
HOAG	(??)	-25.8	10.8 ± 1.5	**1.141**	?	?
WISDM						
Случайный поиск	(??)	**-1086661.1**	22.0 ± 19.3	**0.660**	?	?
Жадная оптимизация	(??)	-1086707.1	**15.4 ± 17.2**	0.707	?	?
DrMAD	(??)	-1086708.2	29.2 ± 8.0	0.694	?	?
HOAG	(??)	-1086733.5	28.2 ± 7.13	0.701	?	?
Случайный поиск	(??)	-35420.4	14.4 ± 7.8	0.732	?	?
Жадная оптимизация	(??)	**-3552.9**	**1.0 ± 0.0**	**0.702**	?	?
DrMAD	(??)	-26091.4	50.0 ± 0.0	0.729	?	?
HOAG	(??)	-16566.6	49.0 ± 0.0	0.733	?	?
MNIST						
Случайный поиск	(??)	-3305.1	13.3 ± 8.1	**0.0179**	?	?
Жадная оптимизация	(??)	**-3416.7**	13.8 ± 9.3	0.0193	?	?
DrMAD	(??)	?	?	?	?	?
HOAG	(??)	-3748.6	**8.6 ± 7.3**	0.0217	?	?
Случайный поиск	(??)	-1304556.4	14.2 ± 5.7	**0.0187**	?	?
Жадная оптимизация	(??)	**-11136.2**	**7.8 ± 3.6**	0.0231	?	?
DrMAD	(??)	?	?	?	?	?
HOAG	(??)	-280061.6	24.0 ± 0.0	0.0189	?	?

Table 3: Результаты экспериментов

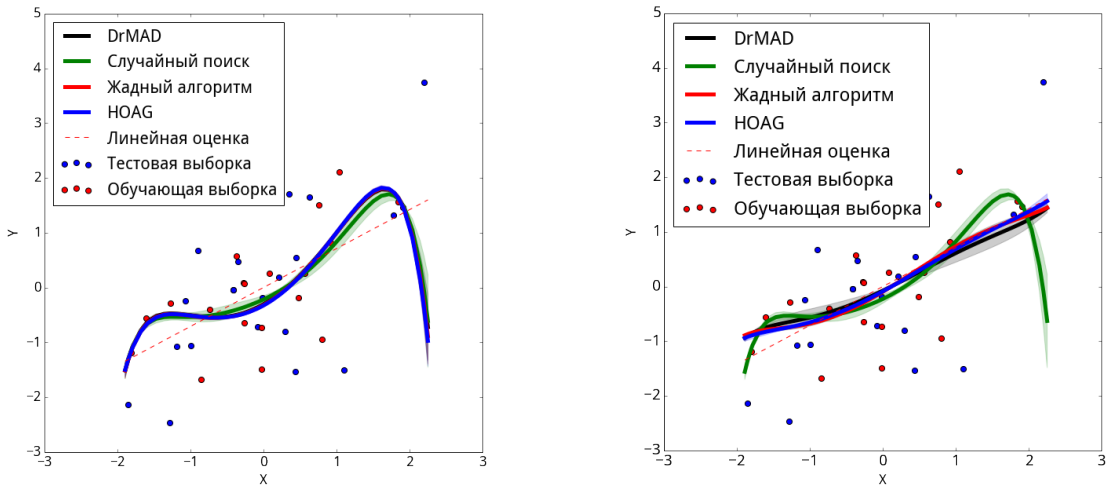

Figure 3: Графики итоговых полиномов для синтетической выборки: a — кросс-валидация, b — вариационная оценка

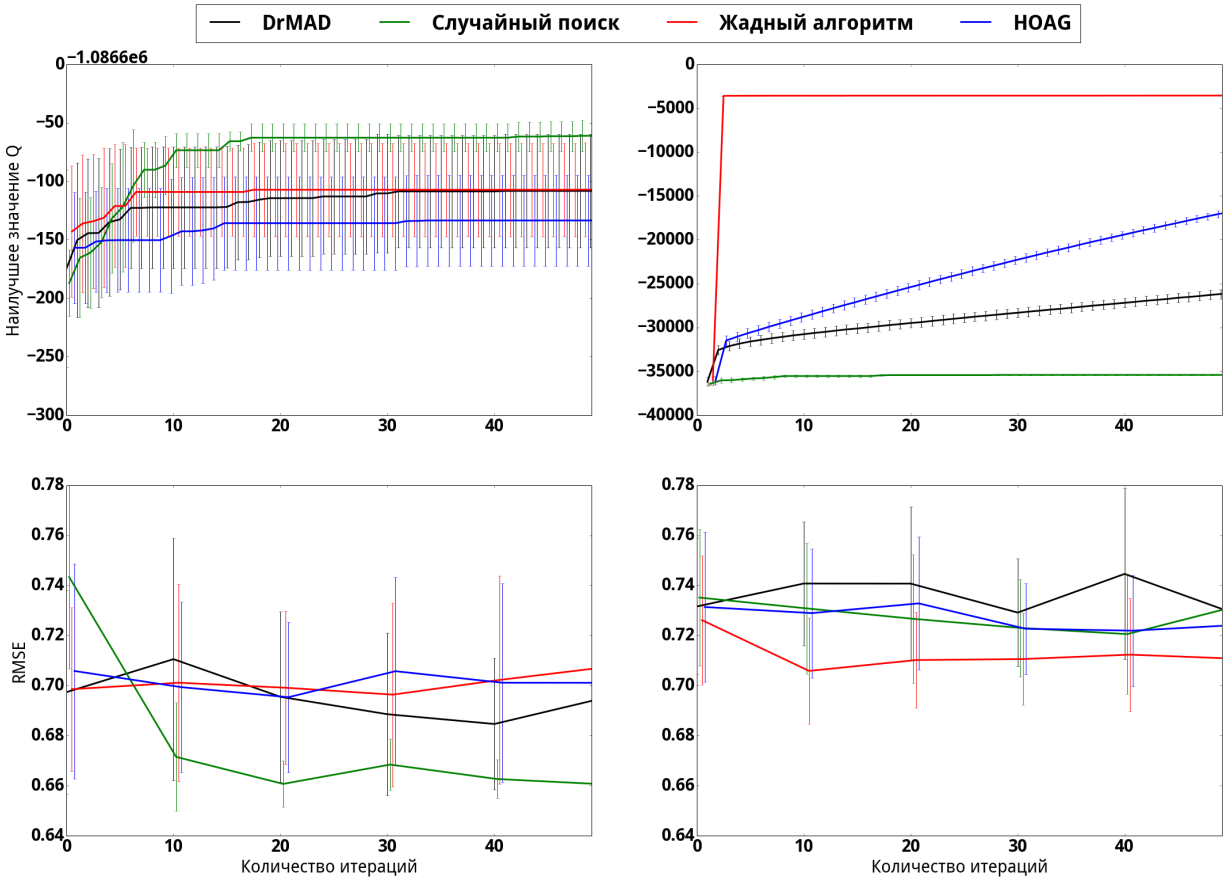

Figure 4: Графики завивисимости функции $\hat{Q}$ и качества модели от количества итераций оптимизации для кросс-валидации: кросс-валидация (слева), вариационная оценка (справа)

- Возможны два улучшения: использование хорошего оптимизатора вместо GD и установка более сложной траектории
- В отличие от HOAG и жадного, для DrMAD есть предпосылки для оптимизации параметров оптимизации

6 Заключение

В работе было проведено сравнение градиентных методов оптимизации гиперпараметров. В качестве базового алгоритма выступал выбор гиперпараметров модели с использованием случайного поиска. В качестве критерия выбора модели выступали вариационная нижняя оценка правдоподобия модели и ошибка на валидационной части выборки, исследованы их свойства и устойчивость получаемых моделей. Было проведено исследование поведения алгоритмов на выборках WISDM и MNIST. В дальнейшем планируется обобщение рассматриваемых методов для проведения мета-оптимизации и получения оценок параметров градиентных оптимизаций.

References

[1] *Salakhutdinov R., Hinton G.* Learning a Nonlinear Embedding by Preserving Class Neighbourhood Structure // Journal of Machine Learning Research - Proceedings Track. — 2007. — P. 412–419.

[2] *Sutskever I., Vinyals O., Le Q.* Sequence to Sequence Learning with Neural Networks // Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. — 2014. — P. 3104–3112.

[3] *Korzeń M., Jaroszewicz S., Klęsk P.* Logistic regression with weight grouping priors // *Computational Statistics & Data Analysis*. — 2013. — Vol. 64, no. Supplement C. — P. 281 – 298. http://www.sciencedirect.com/science/article/pii/S0167947313001151.

[4] *Fu Jie, Luo Hongyin, Feng Jiashi et al.* DrMAD: Distilling Reverse-Mode Automatic Differentiation for Optimizing Hyperparameters of Deep Neural Networks // *arXiv preprint arXiv:1601.00917*. — 2016.

[5] *Pedregosa Fabian.* Hyperparameter optimization with approximate gradient // Proceedings of the 33rd International Conference on Machine Learning. — 2016.

[6] *Luketina Jelena, Berglund Mathias, Greff Klaus, Raiko Tapani.* Scalable gradient-based tuning of continuous regularization hyperparameters // *arXiv preprint arXiv:1511.06727*. — 2015.

[7] *Kwapisz Jennifer R, Weiss Gary M, Moore Samuel A.* Activity recognition using cell phone accelerometers // *ACM SigKDD Explorations Newsletter.* — 2011. — Vol. 12, no. 2. — P. 74–82.

[8] *LeCun Yann.* The MNIST database of handwritten digits // *http://yann. lecun. com/exdb/mnist/.* — 1998.

[9] *Bergstra James, Bengio Yoshua.* Random search for hyper-parameter optimization // *Journal of Machine Learning Research.* — 2012. — Vol. 13, no. Feb. — P. 281–305.

[10] *Bergstra James S, Bardenet Rémi, Bengio Yoshua, Kégl Balázs.* Algorithms for hyper-parameter optimization // Advances in Neural Information Processing Systems. — 2011. — P. 2546–2554.

[11] *Snoek Jasper, Rippel Oren, Swersky Kevin et al.* Scalable Bayesian Optimization Using Deep Neural Networks. // ICML. — 2015. — P. 2171–2180.

[12] *Hutter Frank, Hoos Holger H, Leyton-Brown Kevin.* Sequential model-based optimization for general algorithm configuration // International Conference on Learning and Intelligent Optimization / Springer. — 2011. — P. 507–523.

[13] *Maclaurin D., Duvenaud D., Adams R.* Gradient-based Hyperparameter Optimization through Reversible Learning // Proceedings of the 32nd International Conference on Machine Learning (ICML-15) / Ed. by David Blei, Francis Bach. — JMLR Workshop and Conference Proceedings, 2015. — P. 2113–2122.

[14] *Domke Justin.* Generic Methods for Optimization-Based Modeling. // AISTATS / Ed. by Neil D. Lawrence, Mark A. Girolami. — Vol. 22 of *JMLR Proceedings.* — JMLR.org, 2012. — P. 318–326. http://dblp.uni-trier.de/db/journals/jmlr/jmlrp22.html#Domke12.

[15] *MacKay David J. C.* Information Theory, Inference & Learning Algorithms. — New York, NY, USA: Cambridge University Press, 2002.

[16] *Bishop C.* Pattern Recognition and Machine Learning (Information Science and Statistics). — Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[17] *Токмакова А. А., Стрижов В. В.* Оценивание гиперпараметров линейных и регрессионных моделей при отборе шумовых и коррелирующих признаков // *Информатика и её применения.* — 2012. — Т. 6(4). — С. 66–75. http://strijov.com/papers/Tokmakova2011HyperParJournal_Preprint.pdf.

[18] *Зайцев А. А., Стрижов В. В., Токмакова А. А.* Оценка гиперпараметров регрессионных моделей методом максимального правдоподобия // *Информационные технологии.* — 2013. — Vol. 2. — P. 11–15. http://strijov.com/papers/ZaytsevStrijovTokmakova2012Likelihood_Preprint.pdf.

[19] *Стрижов В. В.* Порождение и выбор моделей в задачах регрессии и классификации: Ph.D. thesis / Вычислительный центр РАН. — 2014. http://strijov.com/papers/Strijov2015ModelSelectionRu.pdf.

[20] *Hoffman Matthew D., Blei David M., Wang Chong, Paisley John.* Stochastic Variational Inference // *J. Mach. Learn. Res.* — 2013. — may. — Vol. 14, no. 1. — P. 1303–1347. http://dl.acm.org/citation.cfm?id=2502581.2502622.

[21] *Salimans Tim, Kingma Diederik P., Welling Max.* Markov Chain Monte Carlo and Variational Inference: Bridging the Gap. // ICML / Ed. by Francis R. Bach, David M. Blei. — Vol. 37 of *JMLR Proceedings*. — JMLR.org, 2015. — P. 1218–1226. http://dblp.uni-trier.de/db/conf/icml/icml2015.html#SalimansKW15.

[22] *Grünwald P.* A Tutorial Introduction to the Minimum Description Length Principle // Advances in Minimum Description Length: Theory and Applications. — MIT Press, 2005.

[23] *Graves A.* Practical Variational Inference for Neural Networks // Advances in Neural Information Processing Systems 24 / Ed. by J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett et al. — Curran Associates, Inc., 2011. — P. 2348–2356.

[24] *Arlot Sylvain, Celisse Alain.* A survey of cross-validation procedures for model selection // *Statist. Surv.* — 2010. — Vol. 4. — P. 40–79. http://dx.doi.org/10.1214/09-SS054.

[25] *Vishnu Abhinav, Narasimhan Jeyanthi, Holder Lawrence et al.* Fast and Accurate Support Vector Machines on Large Scale Systems // 2015 IEEE International Conference on Cluster Computing, CLUSTER 2015, Chicago, IL, USA, September 8-11, 2015. — 2015. — P. 110–119. http://dx.doi.org/10.1109/CLUSTER.2015.26.

[26] *Krstajic Damjan, Buturovic Ljubomir J., Leahy David E., Thomas Simon.* Cross-validation pitfalls when selecting and assessing regression and classification models // *Journal of Cheminformatics*. — 2014. — Vol. 6, no. 1. — P. 1–15. http://dx.doi.org/10.1186/1758-2946-6-10.

[27] *Hornung Roman, Bernau Christoph, Truntzer Caroline et al.* Full versus incomplete cross-validation: measuring the impact of imperfect separation between training and test sets in prediction error estimation. — 2014. http://nbn-resolving.de/urn/resolver.pl?urn=nbn:de:bvb:19-epub-20682-6.

[28] *Bengio Yoshua, Grandvalet Yves.* No Unbiased Estimator of the Variance of K-Fold Cross-Validation // *J. Mach. Learn. Res.* — 2004. — dec. — Vol. 5. — P. 1089–1105. http://dl.acm.org/citation.cfm?id=1005332.1044695.

[29] *Farcomeni Alessio.* Bayesian constrained variable selection // *Statistica Sinica*. — 2010. — P. 1043–1062.