Московский физико-технический институт
(национальный исследовательский университет)
Физтех-школа прикладной математики и информатики
Кафедра интеллектуальных систем

Богданов Александр Иванович

Аппроксимации градиента с помощью оракула нулевого порядка и техники запоминания

03.03.01 — Прикладные математика и физика

Выпускная квалификационная работа бакалавра

Научный руководитель:
к.ф.-м.н. А. Н. Безносиков

Москва — 2024

Аннотация

В данной работе рассматривается проблема оптимизации «черного ящика». В такой постановке задачи не имеется доступа к градиенту целевой функции, поэтому его необходимо как-то оценить. Предлагается новый способ аппроксимации градиента `JAGUAR`, который запоминает информацию из предыдущих итераций и требует $\mathcal{O}(1)$ обращений к оракулу. Эта аппроксимацию реализована для алгоритма Франка-Вульфа и для него доказана сходимость для выпуклой постановки задачи. Помимо детерминированной постановки рассматривается и стохастическая задача минимизации на множестве Q с шумом в оракуле нулевого порядка, такая постановка довольно непопулярна в литературе, но я было доказано, что `JAGUAR` является робастной и в таком случае. Проведены эксперименты по сравнению оценщика градиента `JAGUAR` с уже известными в литературе и подтверждено его доминирование.

Содержание

1 Введение

Методы без проекций, такие как условный градиент в алгоритме Франка-Вульфа [1], широко используются для решения различных задач оптимизации. В последнее десятилетие методы условного градиента вызывают все больший интерес в сообществе машинного обучения, поскольку во многих случаях вычислительно дешевле решить линейную задачу минимизации на подходящем выпуклом множестве (например, на l_p-шарах или симплексе Δ_d), а затем сделать проекцию на него [2, 3, 4, 5, 6, 7, 8].

В оригинальной работе Франка-Вульфа [1] авторы использовали истинный градиент в своем алгоритме, однако современные задачи машинного обучения и искусственного интеллекта требуют использования различных оценок градиента, это связано со значительным увеличением размера датасетов и сложности современных моделей. Примерами таких градиентных оценок в алгоритмах типа Франка-Вульфа являются координатные методы [9, 10, 11] и стохастическая градиентная аппроксимация с батчами [12, 13, 14].

Но иногда встречаются еще более сложные ситуации, когда нельзя вычислить градиент в общем случае, потому что он недоступен по разным причинам, например, целевая функция не дифференцируема или вычисление градиента вычислительно сложно [15, 16, 17, 18, 19]. Такая постановка называется оптимизацией «черного ящика» [20], и в этом случае необходимо использовать методы оценки градиента нулевого порядка через конечные разности целевой функции (иногда с дополнительным шумом) для аппроксимации градиента [21, 22].

За последние годы исследований по теме оптимизации «черного ящика» можно выделить два основных метода аппроксимации градиента с помощью конечных разностей. Первый оценивает градиент в m координатах [23, 24, 25]:

$$\frac{d}{m}\sum_{i\in I}\frac{f(x+\tau e_i)-f(x-\tau e_i)}{2\tau}e_i, \tag{1}$$

где $I \subset \overline{1,d} : |I| = m$, e_i – вектор из стандартного базиса в $\mathbb{R}^d$ и τ – параметр

сглаживания.

Эта конечная разность аппроксимирует градиент в координатах m и требует $\mathcal{O}(m)$ вызовов оракула. Если m мало, то такая оценка будет неточной, если m велико, то на каждой итерации нужно делать много обращений к оракулу нулевого порядка. В случае $m = d$ этот метод называется *полная аппроксимация*.

Второй использует в конечной разности не стандартный базис, а случайные вектора e [17, 22, 26, 27]:

$$d\frac{f(x+\tau e) - f(x-\tau e)}{2\tau}e, \tag{2}$$

где e может быть равномерно распределено на l_p-сфере $RS_p^d(1)$, тогда эта схема называется l_p-*сглаживание*. В последних работах авторы обычно используют $p = 1$ [28, 29] или $p = 2$ [30, 31, 32]. Кроме того, e может быть взято из нормального распределения с нулевым средним и единичной ковариационной матрицей [17].

Аппроксимации (1) и (2) имеют очень большую дисперсию или требуют много обращений к нулевому оракулу, поэтому возникает необходимость как-то уменьшить ошибку аппроксимации, не увеличивая при этом количество обращений к нулевому оракулу. В стохастической оптимизации довольно широко используется метод запоминания информации с предыдущих итераций, например, в SVRG [33], SAGA [34], SARAH [35] и SEGA [36] авторы предлагают запоминать градиент с предыдущих итераций для лучшей сходимости метода. Я решил использовать эту технику в задаче оптимизации «черного ящика» и запоминать градиентные аппроксимации из предыдущих итераций для уменьшения размера батча без существенной потери точности.

В этой работе я попытаюсь ответить на следующие вопросы:

- *Можно ли создать метод нулевого порядка, который будет использовать информацию из предыдущих итераций и аппроксимировать истинный градиент так же точно, как и полная аппроксимация* (1), *но потребует* $\mathcal{O}(1)$ *вызовов оракула нулевого порядка?*

- *Можно ли реализовать этот метод аппроксимации в алгоритме Франка-Вольфа для детерминированных и стохастических постановок задач минимизации?*

- *Является ли оценка сходимости этого метода лучше, чем для разностных схем* (1) *и* (2)*?*

В более реалистичной постановке оракул нулевого порядка возвращает зашумленное значение целевой функции, то есть выдает не $f(x)$, а $f(x) + \delta(x)$. В литературе рассматриваются различные виды шума $\delta(\cdot)$: он может быть стохастическим [26, 32, 37, 38] или детерминированным [39, 40, 41, 42, 43, 44]. Поэтому возникает еще один исследовательский вопрос:

- *Как различные типы шума влияют на теоретические гарантии и практические результаты для предложенных мной подходов?*

В соответствии с вопросами исследования, мой вклад может быть обобщен следующим образом:

- Представлен метод `JAGUAR`, который аппроксимирует истинный градиент целевой функции $\nabla f(x)$ в точке x. Использование памяти предыдущих итераций позволяет достичь точности, близкой к полной аппроксимации (1), но `JAGUAR` требует не $\mathcal{O}(d)$, а $\mathcal{O}(1)$ обращений к оракулу нулевого порядка. Сглаживание l_p (2) также требует $\mathcal{O}(1)$ обращения к оракулу, но поскольку в нем нет техники памяти, этот метод имеет большую дисперсию и не является робастным. (см. раздел (3.1))

- Внедрена аппроксимация `JAGUAR` в алгоритм Франка-Вульфа для стохастических и детерминированных задач минимизации и доказал сходимость в обоих случаях (см. разделы 3.2.1 и 3.2.2).

- Проведены вычислительные эксперименты, сравнения аппроксимации `JAGUAR` с l_2-*сглаживанием* (2) и *полной аппроксимаций* (1) на различных задачах минимизации (см. раздел 4).

В литератре некоторые авторы считают координатные методы [9] тоже градиентной аппроксимаций, но эти методы используют истинный градиент целевой функции f, поэтому ее нельзя напрямую применить в оптимизации «черного ящика».

Метод l_p-сглаживания не требует дифференцируемости целевой функции, поскольку рассматривает сглаженную версию функции f вида $f_\gamma(x) = \mathbb{E}_e\left[f(x+\gamma e)\right]$. В общем случае метод l_p-сглаживания может аппроксимировать градиент с помощью $\mathcal{O}(1)$ вызовов оракула [43], но он может быть не робастным в постановке Франка-Вульфа, поскольку в [45] авторам приходится собирать большой батч направлений e для достижения сходимости. Отметим, что в [45] рассматривается нестохастический шум.

Полная аппроксимация также используется в литературе [46, 47, 48], но на каждой итерации нам необходимо делать $\mathcal{O}(d)$ вызовов оракула, а поскольку в современных приложениях d огромно, это может быть проблемой. Также этот метод требует гладкости объективной функции f.

В таблице 1 приведено сравнение постановок задач, методов аппроксимации и результатов для них.

Метод	Постановка		Шум		Размер батча	Аппроксимация
	Гладкая	Нулевой порядок	Стохастический	Детерминированный		
ZO-SCGS [45]	✗	✓	✗	✓	$\mathcal{O}\left(1/\varepsilon^2\right)$	*l_2-сглаживание* (2)
FZFW [47]	✓	✓	✗	✗	$\mathcal{O}\left(\sqrt{d}\right)$	*полная аппроксимация* (1)
DZOFW [46]	✓	✓	✗	✗	$\mathcal{O}\left(d\right)$	*полная аппроксимация* (1)
MOST-FW [48]	✓	✓	✗	✗	$\mathcal{O}\left(d\right)$	*полная аппроксимация* (1)
BCFW [9]	✓	✗	✗	✗	$\mathcal{O}\left(1\right)$	координатный
SSFW [49]	✓	✗	✗	✗	$\mathcal{O}\left(1\right)$	координатный
FW с JAGUAR (эта работа)	✓	✓	✓	✓	$\mathcal{O}\left(1\right)$	JAGUAR (Алгоритмы 1 и 2)

Таблица 1: Сравнение различных методов нулевого порядка и координатных методов алгоритма Франка-Вульфа.

2 Постановка задачи

В данной работе рассматривается оптимизационная задача:

$$f^* := \min_{x \in Q} f(x). \tag{3}$$

2.1 Детерминированный случай

Предполагается, что доступ есть только к оракулу нулевого порядка, и он возвращает зашумленное значение функции $f(x)$:

$$f_\delta(x) := f(x) + \delta(x).$$

На функцию и шум накладываются классические ограничения необходимые для анализа:

- Функция $f(x)$ L-гладкая на множестве Q, т.е.

$$\exists L > 0 : \forall x, y \in Q \hookrightarrow \|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|. \tag{4}$$

- Функция $f(x)$ выпуклая на множестве Q, т.е.

$$\forall x, y \in Q \hookrightarrow f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle. \tag{5}$$

- Шум $\delta(x)$ оракула ограничен, т.е.

$$\exists \Delta > 0 : \forall x \in Q \hookrightarrow |\delta(x)|^2 \leq \Delta^2. \tag{6}$$

2.2 Стохастический случай

В этом разделе рассматривается стохастическая версия задачи: (3):

$$f(x) := \mathbb{E}_{\xi \sim \pi} [f(x, \xi)], \tag{7}$$

где ξ – случайный вектор из обычно неизвестного распределения π.

Снова предполагается, что нет доступа к истинному значению градиента $\nabla f(x, \xi)$, и оракул нулевого порядка возвращает зашумленное значение функции $f(x, \xi)$:

$$f_\delta(x, \xi) := f(x, \xi) + \delta(x, \xi).$$

На функцию и шум также накладываются классические ограничения необходимые для анализа:

- Функция $f(x, \xi)$ $L(\xi)$-гладкая на множестве Q, т.е.

$$\forall x, y \in Q \hookrightarrow \|\nabla f(x, \xi) - \nabla f(y, \xi)\| \leq L(\xi) \|x - y\|, \tag{8}$$

  где $L^2 := \mathbb{E}\left[L(\xi)^2\right]$.

  С учетом этого предположения, функция $f(x, \xi)$ является L-гладкой на множестве Q:

$$\begin{aligned} \forall x, y \in Q \hookrightarrow \|\nabla f(x) - \nabla f(y)\|^2 &= \|\mathbb{E}\left[\nabla f(x, \xi) - \nabla f(y, \xi)\right]\|^2 \\ &\leq \mathbb{E}\left[\|\nabla f(x, \xi) - \nabla f(y, \xi)\|^2\right] \\ &\leq L^2 \|x - y\|^2. \end{aligned}$$

- Функция $f(x, \xi)$ выпуклая на множестве Q, т.е.

$$\forall x, y \in Q \hookrightarrow f(y, \xi) \geq f(x, \xi) + \langle \nabla f(x, \xi), y - x \rangle. \tag{9}$$

- Шум оракула ограничен некоторой константой $\Delta > 0$, т.е.

$$\exists \Delta > 0 : \forall x \in Q \hookrightarrow \mathbb{E}\left[|\delta(x, \xi)|^2\right] \leq \Delta^2. \tag{10}$$

  С учетом этого предположения, если определить $\delta(x) := \mathbb{E}\left[\delta(x, \xi)\right]$, то $|\delta(x)|^2 \leq \Delta^2$, так как $|\delta(x)|^2 = |\mathbb{E}\left[\delta(x, \xi)\right]|^2 \leq \mathbb{E}\left[|\delta(x, \xi)|^2\right] \leq \Delta^2$.

- Второй момент $\nabla f(x, \xi)$ ограничен, т.е.

$$\exists \sigma_\nabla \geq 0 : \forall x \in Q \hookrightarrow \mathbb{E}\left[\|\nabla f(x, \xi) - \nabla f(x)\|^2\right] \leq \sigma_\nabla^2. \tag{11}$$

- Второй момент $f(x, \xi)$ ограничен, т.е.

$$\exists \sigma_f \geq 0 : \forall x \in Q \hookrightarrow \mathbb{E}\left[|f(x, \xi) - f(x)|^2\right] \leq \sigma_f^2. \tag{12}$$

3 Основные результаты

3.1 JAGUAR

Выше были рассмотрены методы аппроксимации градиента с помощью конечных разностей (1) и (2). В этом разделе представлены новые методы оценки градиента JAGUAR: JAGUAR-d для детерминированной и JAGUAR-s для стохастической задач, основанные на уже исследованных методах и использующие память предыдущих итераций.

Идея метода JAGUAR схожа с известными методами уменьшения дисперсии, такими как SAGA [34] или SVRG [33], но эти методы применяют уменьшение дисперсии к батчам. Однако при оптимизации нулевого порядка нужно аппроксимировать градиент, поэтому необходимо применить технику уменьшения дисперсии к координатам [36]. Следовательно, метод JAGUAR использует память некоторых координат предыдущих градиентов, а не запоминает градиенты по батчам в прошлых точках. В литературе уже есть работы, сочетающие оптимизацию нулевого порядка и уменьшение дисперсии, но суть их в том, что они меняют вычисление градиента на безградиентную аппроксимацию (1) в пакетных алгоритмах с уменьшением дисперсии, таких как SVRG или SPIDER [50], а не используют технику уменьшения дисперсии для координат, как в алгоритме 1.

Для детерминированного алгоритма аппроксимации используется разностная схема:

$$\widetilde{\nabla}_i f_\delta(x) := \frac{f_\delta(x + \tau e_i) - f_\delta(x - \tau e_i)}{2\tau} e_i, \tag{13}$$

где e_i – вектор из стандартного базиса в $\mathbb{R}^d$. Сам алгоритм аппроксимации градиента для детерминированной задачи JAGUAR-d в точке x выглядит так (алгоритм 1):

Алгоритм 1 `JAGUAR-d`

Вход: $x, h \in \mathbb{R}^d$

Сэмплируем $i \in \overline{1, d}$ равномерно и независимо

Считаем $\widetilde{\nabla}_i f_\delta(x) = \frac{f_\delta(x+\tau e_i) - f_\delta(x-\tau e_i)}{2\tau} e_i$

$h = h - \langle h, e_i \rangle e_i + \widetilde{\nabla}_i f_\delta(x)$

Выход: h

В стохастической постановке (7) есть две версии разностных схем (13). Первая называется двухточечной обратной связью (ДОС) [26, 31, 32, 51, 52], в данном случае аппроксимация градиента функции $f(x, \xi)$:

$$\widetilde{\nabla}_i f_\delta(x, \xi) := \frac{f_\delta(x + \tau e_i, \xi) - f_\delta(x - \tau e_i, \xi)}{2\tau} e_i. \tag{14}$$

Вторая называется одноточечной обратной связью (ООС) [30, 38, 53, 54, 55], в этом случае аппроксимация градиента функции $f(x, \xi)$:

$$\widetilde{\nabla}_i f_\delta(x, \xi^\pm) := \frac{f_\delta(x + \tau e_i, \xi^+) - f_\delta(x - \tau e_i, \xi^-)}{2\tau} e_i. \tag{15}$$

Ключевое различие между приближениями (14) и (15) заключается в том, что схема (14) более точна, но ее сложно реализовать на практике, так как для этого необходимо получить одну и ту же реализацию ξ в двух разных точках $x + \tau e$ и $x - \tau e$, поэтому схема (15) более интересна с практической точки зрения. Для дальнейшего упрощения выкладок считается, что в случае двухточечной обратной связи (14) $\xi^+ = \xi^- = \xi$. Алгоритм аппроксимации градиента для стохастической задачи `JAGUAR-s` в точке x (алгоритм 2):

Алгоритм 2 `JAGUAR-s`

1: **Вход:** $x, h, g \in \mathbb{R}^d$; $\eta \in [0, 1]$

2: Сэмплируем $i \in \overline{1, d}$ равномерно и независимо

3: Сэмплируем ξ: ξ^+ и ξ^- независимо (в ДОС $\xi^+ = \xi^-$)

4: Считаем $\widetilde{\nabla}_i f_\delta(x, \xi^\pm) = \frac{f_\delta(x+\tau e_i, \xi^+) - f_\delta(x-\tau e_i, \xi^-)}{2\tau} e_i$

5: $h = h - \langle h, e_i \rangle e_i + \widetilde{\nabla}_i f_\delta(x, \xi^+, \xi^-)$

6: $\rho = h - d \cdot \langle h, e_i \rangle e_i + d \cdot \widetilde{\nabla}_i f_\delta(x, \xi^+, \xi^-)$

7: $g = (1 - \eta)g + \eta\rho$

8: **Выход:** g, h

Алгоритм `JAGUAR-s` (алгоритм 2) аналогичен `JAGUAR-d` (алгоритм 1), но в строках 6 и 7 используются части SEGA [36] и моментум [56] для сходимости в стохастическом случае.

В стохастическом случае необходима часть SEGA [36] ρ_k в алгоритме 2, поскольку важно свойство «несмещенности» (см. доказательство леммы 4), т.е.

$$\mathbb{E}_k \left[\rho^k\right] = \widetilde{\nabla} f_\delta(x^k) := \sum_{i=1}^{d} \frac{f_\delta(x + \tau e_i) - f_\delta(x - \tau e_i)}{2\tau} e_i,$$

где $\mathbb{E}_k \left[\rho^k\right]$ – условное математическое ожидание на шаге k. Использование части SEGA ρ^k ухудшает оценки в d раз по сравнению с использованием h^k в качестве градиентной аппроксимации (см. леммы 2 и 3).

В стохастическом случае необходим моментум [56] η_k в алгоритме 2, поскольку при оценке выражения $\mathbb{E}\left[\left\|\widetilde{\nabla} f_\delta(x, \xi^\pm_{\overline{1,d}}) - \nabla f(x)\right\|^2\right]$ в стохастическом случае появляются выражения, содержащие σ_∇^2 и σ_f^2, и они мешают сходимости (см. лемму 1).

3.1.1 Применение `JAGUAR`

Алгоритм аппроксимации градиента `JAGUAR-d` может быть использован с любыми итерационными схемами, которые на каждом шаге k возвращают

новую точку x^k. Используя эти точки, мы получается последовательность h^k, которая в некотором смысле служит памятью компонент градиента из прошлых моментов. Поэтому в методах инкрементальной оптимизации имеет смысл использовать h^k в качестве оценки истинного градиента $\nabla f(x^k)$. Используя следующую унифицированную схему, можно описать такой итерационный алгоритм, решающий задачу (3) (алгоритм 3).

Алгоритм 3 Итерационный алгоритм с использованием `JAGUAR-d`

1: **Вход:** `Proc` и h^0
2: **for** $k = 0, 1, 2, ..., N$ **do**
3: $h^{k+1} = \texttt{JAGUAR-d}(x^k, h^k)$
4: $x^{k+1} = \texttt{Proc}(x^k, \texttt{grad\_est} = h^{k+1})$
5: **end for**
6: **Выход:** x^{N+1}

Используя алгоритм аппроксимации градиента `JAGUAR-s`, можно описать такой итерационный алгоритм, решающий задачу (3) + (7) (алгоритм 4).

Алгоритм 4 Итерационный алгоритм с использованием `JAGUAR-s`

1: **Вход:** `Proc` и h^0
2: **for** $k = 0, 1, 2, ..., N$ **do**
3: $h^{k+1}, g^{k+1} = \texttt{JAGUAR-s}(x^k, h^k, g^k, \eta_k)$
4: $x^{k+1} = \texttt{Proc}(x^k, \texttt{grad\_est} = g^{k+1})$
5: **end for**
6: **Выход:** x^{N+1}

В алгоритмах 3 и 4, `Proc`(x^k, `grad_est`) – это некоторая последовательность действий, которая переводит x^k в x^{k+1}, используя `grad_est` в качестве истинного градиента. Ниже приведен анализ аппроксимации градиента `JAGUAR`. Необходимо оценить близость истинного градиента $\nabla f(x^k)$ / $\nabla f(x^k, \xi)$ и выхода алгоритма `JAGUAR` h^k / g^k на шаге k.

3.1.2 Анализ аппроксимаций JAGUAR

Для анализа алгоритма 3 используется обозначение:

$$\widetilde{\nabla} f_\delta(x) := \sum_{i=1}^{d} \frac{f_\delta(x+\tau e_i) - f_\delta(x-\tau e_i)}{2\tau} e_i.$$

Для анализа алгоритма 4 используется обозначение:

$$\widetilde{\nabla} f_\delta(x, \xi_{\overline{1,d}}^{\pm}) := \sum_{i=1}^{d} \frac{f_\delta(x+\tau e_i, \xi_i^+) - f_\delta(x-\tau e_i, \xi_i^-)}{2\tau} e_i.$$

Для упрощения выкладок в случае двухточечной обратной связи $\sigma_f = 0$, а в детерминированном случае (3) $\sigma_\nabla = \sigma_f = 0$.

Лемма 1. *При предположениях 8, 10, 11 и 12 в случае OOC* (15) *выполняется следующее неравенство:*

$$\mathbb{E}\left[\left\|\widetilde{\nabla} f_\delta(x, \xi_1^+, \xi_1^-, ..., \xi_d^+, \xi_d^-) - \nabla f(x)\right\|^2\right] \leq dL^2\tau^2 + \frac{8d\sigma_f^2}{\tau^2} + 2d\sigma_\nabla^2 + \frac{2d\Delta^2}{\tau^2}.$$

Лемма 2. *При предположениях 8, 10, 11 и 12 в случае OOC* (15) *выполняется следующее неравенство:*

$$\begin{aligned}\mathbb{E}\left[\left\|h^k - \nabla f(x^k)\right\|^2\right] \leq &\left(1 - \frac{1}{2d}\right)\mathbb{E}\left[\left\|h^{k-1} - \nabla f(x^{k-1})\right\|^2\right] \\ &+ 2dL^2\mathbb{E}\left[\left\|x^k - x^{k-1}\right\|^2\right] \\ &+ L^2\tau^2 + \frac{8\sigma_f^2}{\tau^2} + 2\sigma_\nabla^2 + \frac{2\Delta^2}{\tau^2}.\end{aligned}$$

Лемма 3. *При предположениях 8, 10, 11 и 12 в случае OOC* (15) *выполняется следующее неравенство:*

$$\begin{aligned}\mathbb{E}\left[\left\|\rho^k - \nabla f(x^k)\right\|^2\right] \leq &4d\mathbb{E}\left[\left\|h^{k-1} - \nabla f(x^{k-1})\right\|^2\right] \\ &+ 2dL^2\mathbb{E}\left[\left\|x^k - x^{k-1}\right\|^2\right] \\ &+ 4d^2\left(L^2\tau^2 + \frac{8\sigma_f^2}{\tau^2} + 2\sigma_\nabla^2 + \frac{2\Delta^2}{\tau^2}\right).\end{aligned}$$

В случае двухточечной обратной связи $\sigma_f = 0$.

Лемма 4. *При предположении 8 выполняется следующее неравенство:*

$$\begin{aligned}\mathbb{E}\left[\left\|g^k - \nabla f(x^k)\right\|^2\right] \leq (1-\eta_k)^2\,\mathbb{E}\left[\left\|\nabla f(x^{k-1}) - g^{k-1}\right\|^2\right] \\ + \frac{4L^2}{\eta_k}\mathbb{E}\left[\left\|x^k - x^{k-1}\right\|^2\right] \\ + \eta_k^2\mathbb{E}\left[\left\|\nabla f(x^k) - \rho^k\right\|^2\right] \\ + 3\eta_k\left(dL^2\tau^2 + \frac{2d\Delta^2}{\tau^2}\right).\end{aligned}$$

3.2 Применение JAGUAR в алгоритме Франка-Вульфа

Обычный алгоритм Франка-Вульфа выглядит следующим образом (алгоритм 5):

Алгоритм 5 Алгоритм Франка-Вульфа

1: **Вход:** $x_0 \in Q$, γ_k

2: **for** k = 0, 1, 2, ... , N **do**

3: $s^k = \arg\min_{s \in Q} \left\langle s, \nabla f(x^k)\right\rangle$

4: $x^{k+1} = x^k + \gamma_k(s^k - x^k)$

5: **end for**

6: **Выход:** x^{N+1}

На множество Q накладываются общие ограничения:

- Множество Q – компактное, т.е.

$$\exists D > 0 : \forall x, y \in Q \hookrightarrow \|x - y\| \leq D \tag{16}$$

- Множество Q – выпуклое, т.е.

$$\forall 0 \leq \alpha \leq 1, \forall x, y \in Q \hookrightarrow \alpha x + (1-\alpha)y \in Q \tag{17}$$

В следующих разделах рассмотрены детерминированные и стохастические алгоритмы Франка-Вульфа с использованием аппроксимации градиента JAGUAR.

3.2.1 Детерминированный случай

В этом разделе представляется алгоритм Франка-Вульфа, который решает задачу (3) с помощью аппроксимации градиента `JAGUAR` (алгоритм 1).

Алгоритм 6 Детерминированный алгоритм Франка-Вульфа с `JAGUAR`

1: **Вход:** $x^0 \in Q$, $h^0 = \widetilde{\nabla} f_\delta(x^0)$, γ_k, τ

2: **for** $k = 0, 1, 2, ..., N$ **do**

3: $\quad h^{k+1} = \texttt{JAGUAR-d}\left(x^k, h^k\right)$

4: $\quad s^k = \arg\min_{x \in Q} \left\langle s, h^{k+1} \right\rangle$

5: $\quad x^{k+1} = x^k + \gamma_k (s^k - x^k)$

6: **end for**

7: **Вход:** x^{N+1}

Используя заданную форму функции `Proc` в алгоритме 6, можно тщательно подобрать шаг γ_k.

Теорема 1 (Богданов А., Подбор шага для детерминированного алгоритма Франка-Вульфа с `JAGUAR`)**.** *В предположениях 16, 17 и 4, 5 и 6 для h^k, полученного алгоритмом 6, можно взять*

$$\gamma_k = \frac{4}{k + 8d},$$

тогда выполняется следующая оценка:

$$H_k = \mathcal{O}\left(\left\| \widetilde{\nabla} f_\delta(x^k) - \nabla f(x^k) \right\|^2 + \frac{d^2 \max\{L^2 D^2, H_0\}}{(k+d)^2} \right).$$

Если дополнительно $h^0 = \widetilde{\nabla} f_\delta(x^0) = \sum_{i=1}^{d} \frac{f_\delta(x^0 + \tau e_i) - f_\delta(x^0 - \tau e_i)}{2\tau} e_i$, можно получить:

$$H_k = \mathcal{O}\left(\left\| \widetilde{\nabla} f_\delta(x^k) - \nabla f(x^k) \right\|^2 + \frac{d^2 L^2 D^2}{(k+8d)^2} \right),$$

где используется обозначение $H_k := \mathbb{E}\left[\left\| h^k - \nabla f(x^k) \right\|^2 \right]$.
Подробное доказательство теоремы приведено в Приложении C.

Из Теоремы 1 следует, что после $\mathcal{O}\left(\frac{\sqrt{d}D}{\tau}\right)$ шагов получается такая же оценка, что и в полной аппроксимации.

Теорема 2 (Богданов А., Скорость сходимости детерминированного алгоритма Франка-Вульфа с `JAGUAR` (Алгоритм 6))**.** *В предположениях 16, 17, 4, 5 и 6 можно взять*

$$\gamma_k = \frac{4}{k + 8d},$$

тогда алгоритм Франка-Вульфа с `JAGUAR` *(алгоритм 6) имеет следующую скорость сходимости:*

$$\mathbb{E}\left[f(x^k) - f^*\right] = \mathcal{O}\left(\frac{d\max\{LD^2, F_0\}}{N + 8d} + \sqrt{d}LD\tau + \frac{\sqrt{d}\Delta D}{\tau}\right).$$

Подробное доказательство теоремы приведено в Приложении C.

Следствие 1. *В соответствии с условиями теоремы 2, выбирая* γ_k, τ, Δ *как*

$$\gamma_k = \frac{4}{k + 8d},\ \tau = \mathcal{O}\left(\frac{\varepsilon}{\sqrt{d}LD}\right),\ \Delta = \mathcal{O}\left(\frac{\varepsilon^2}{dLD^2}\right),$$

чтобы получить ε*-приближенное решение* $(\mathbb{E}\left[f(x^k) - f^*\right] \leq \varepsilon)$ *необходимо*

$$\mathcal{O}\left(\frac{d\max\{LD^2, F_0\}}{\varepsilon}\right) \text{ итераций.}$$

Подробное доказательство следствия приведено в Приложении C.

Результаты Теоремы 2 совпадают с результатами [1, 57], в которых авторы использовали истинный градиент и получили результат вида $\mathbb{E}\left[f(x^N) - f^*\right] = \mathcal{O}\left(\max\{LD^2; f(x^0) - f^*\}/N\right)$. В случае нулевого порядка неизбежно появляются члены вида $\mathcal{O}\left(\text{poly}(\tau) + \text{poly}(\Delta/\tau)\right)$, поскольку они имеют решающее значение для аппроксимации истинного градиента и всегда влияют на сходимость методов нулевого порядка [39, 41, 58, 59]. Фактор d, который появляется в теоретических оценках по сравнению с результатом первого порядка, связан со структурой метода нулевого порядка.

3.2.2 Стохастический случай

В этом разделе рассматривается алгоритм Франка-Вульфа, который решает задачу (3) + (7) с помощью аппроксимации градиента JAGUAR (алгоритм 2).

Алгоритм 7 Стохастический алгоритм Франка-Вульфа с JAGUAR

1: **Вход:** $x^0 \in Q$, $h^0 = g^0 = \widetilde{\nabla} f_\delta(x^0)$, γ_k, η_k, τ
2: **for** $k = 0, 1, 2, ..., N$ **do**
3: g^{k+1}, h^{k+1} = JAGUAR-s $\left(x^k, h^k, g^k, \eta_k\right)$
4: $s^k = \arg\min_{x \in Q} \left\langle s, g^{k+1} \right\rangle$
5: $x^{k+1} = x^k + \gamma_k(s^k - x^k)$
6: **end for**
7: **Вход:** x^{N+1}

Можно получить теорему, аналогичную теореме 1, если тщательно подобрать размеры шагов γ_k и η_k.

Теорема 3 (Богданов А., Подбор шага для стохастического алгоритма Франка–Вульфа с JAGUAR)**.** *В предположениях 16, 17, 8, 9, 10, 11 и 12 в случае одноточечной обратной связи, для g^k, полученного алгоритмом 7, можно взять*

$$\gamma_k = \frac{4}{k + 8d^{3/2}} \quad и \quad \eta_k = \frac{4}{(k + 8d^{3/2})^{2/3}},$$

тогда выполняется следующая оценка:

$$G_k = \mathcal{O}\Bigg(\frac{L^2 D^2 + \max\{d^2\sigma_f^2/\tau^2 + d^2\sigma_\nabla^2; dG_0\}}{(k + 8d^{3/2})^{2/3}} + \frac{d^4 \left\| h^0 - \nabla f(x^0) \right\|^2}{(k + 8d^{3/2})^{8/3}} + dL^2\tau^2 + \frac{d\Delta^2}{\tau^2} \Bigg).$$

Если дополнительно $h^0 = g^0 = \widetilde{\nabla} f_\delta(x^0, \xi^{\pm}_{1,d})$, *то получается:*

$$G_k = \mathcal{O}\left(\left\|\widetilde{\nabla} f_\delta(x^k) - \nabla f(x^k)\right\|^2 + \frac{L^2D^2 + d^2\sigma_f^2/\tau^2 + d^2\sigma_\nabla^2}{(k+8d^{3/2})^{2/3}}\right),$$

где используется обозначение $G_k := \mathbb{E}\left[\left\|g^k - \nabla f(x^k)\right\|^2\right]$. *В случае двухточечной обратной связи* $\sigma_f^2 = 0$.
Подробное доказательство теоремы приведено в Приложении B.

Полученная оценка хуже по сравнению с детерминированным случаем в Теореме 1, поскольку рассматривается более сложная постановка.

Теорема 4 (Богданов А., Скорость сходимости стохастического алгоритма Франка-Вульфа с `JAGUAR` (Алгоритм 7))**.** *В предположениях 16, 17, 8, 9 10, 11 и 12 в случае одноточечной обратной связи можно взять:*

$$\gamma_k = \frac{4}{k+8d^{3/2}} \quad u \quad \eta_k = \frac{4}{(k+8d^{3/2})^{2/3}},$$

тогда алгоритм Франка-Вульфа с `JAGUAR` *(Алгоритм 7) имеет следующую скорость сходимости:*

$$F_N = \mathcal{O}\left(\frac{LD^2 + d\sigma_f D/\tau + d\sigma_\nabla D + \sqrt{d}F_0}{(N+8d^{3/2})^{1/3}} + \sqrt{d}LD\tau + \frac{\sqrt{d}\Delta D}{\tau}\right),$$

где используется обозначение $F_k := \mathbb{E}\left[f(x^k) - f^*\right]$. *В случае двухточечной обратной связи* $\sigma_f = 0$.
Подробное доказательство теоремы приведено в Приложении C.

Следствие 2. *В соответствии с условиями теоремы 4, выбирая* $\gamma_k, \eta_k, \tau, \Delta$ *как*

$$\gamma_k = \frac{4}{k+8d^{3/2}},\ \eta_k = \frac{4}{(k+8d^{3/2})^{2/3}},\ \tau = \mathcal{O}\left(\frac{\varepsilon}{\sqrt{d}LD}\right),\ \Delta = \mathcal{O}\left(\frac{\varepsilon^2}{dLD^2}\right),$$

чтобы получить ε*-приближенное решение* ($\mathbb{E}\left[f(x^N) - f^*\right] \le \varepsilon$) *необходимо*

$$\mathcal{O}\left(\max\left\{\left[\frac{LD^2 + d\sigma_\nabla D + \sqrt{d}(f(x^0) - f^*)}{\varepsilon}\right]^3; \frac{d^{9/2}\sigma_f^3 L^3 D^6}{\varepsilon^6}\right\}\right) \text{ итераций.}$$

В случае двухточечной обратной связи $\sigma_f = 0$ *и последнее выражение принимает вид*

$$\mathcal{O}\left(\left[\frac{LD^2 + d\sigma_\nabla D + \sqrt{d}(f(x^0) - f^*)}{\varepsilon}\right]^3\right) \text{ итераций.}$$

Подробное доказательство следствия приведено в Приложении C.

Поскольку в алгоритме аппроксимации `JAGUAR` (алгоритм 7) использовались части SEGA и импульса, то не получается та же скорость сходимости, что и в теоремах 1 и 2 даже при переходе от стохастических к детерминированным настройкам, т.е, при задании $\sigma_\Delta = \sigma_f = 0$ в теоремах 3 и 4. Те же проблемы возникают и в случае первого порядка [13, 56], это связано с трудностями реализации стохастического градиента в алгоритмах типа Франка-Вульфа.

Можно применить `JAGUAR-d` (алгоритм 1) к стохастической задаче (7) и получить те же оценки, что и в Теоремах 1 и 2, только сглаженный член вида $\mathcal{O}\left(\text{poly}(\tau) + \text{poly}(\Delta/\tau)\right)$ будет содержать слагаемые вида $\mathcal{O}\left(\text{poly}(\sigma_\Delta^2) + \text{poly}(\sigma_f^2/\tau)\right)$. Поэтому, если $\sigma_\Delta^2, \sigma_f^2 \sim \Delta$, то детерминированный алгоритм 1 подходит для стохастической задачи (7). Однако это означает, что нужно использовать большие батчи, поэтому необходимо использовать SEGA и импульсные части в `JAGUAR-s` аппроксимации.

4 Вычислительный эксперимент

В этом разделе представлены результаты экспериментов по применению аппроксимации нулевого порядка JAGUAR к различным задачам оптимизации «черного ящика». Результаты включают детерминированный и стохастический случаи алгоритма Франка-Вульфа.

4.1 Постановка эксперимента

Рассматривается модель логистической регрессии на множестве Q вида:

$$\min_{w \in Q} \left\{ f(w) = \frac{1}{m} \sum_{k=1}^{m} \log\left(1 + \exp\left[-y_k (Xw)_k\right]\right) + \frac{1}{2C}\|w\|^2 \right\}.$$

Также рассматривается модель SVM на множестве Q вида:

$$\min_{w \in Q, b \in \mathbb{R}} \left\{ f(w,b) = \frac{1}{m} \sum_{k=1}^{m} \left(1 - y_k[(Xw)_k - b]\right)_+ + \frac{1}{2C}\|w\|^2 \right\}.$$

В обеих задачах используется регуляризационный член $C = 10$. В качестве минимизирующего множества Q рассматриваются симплекс Δ_d и l_2-шар. Для решения задачи классификации используются классические датасеты MNIST [60] и Mushrooms [61].

В эксперименте сравниваются различные методы аппроксимации. В качестве базовых оценок градиентов рассматриваются l_2-*сглаживание* (2) и *полная аппроксимация* (1). Показывается, что алгоритм, использующий аппроксимацию JAGUAR (алгоритмы 1 и 7), работает лучше всего.

4.2 Детерминированный алгоритм Франка-Вульфа

В этом разделе рассматриваем детерминированный шум вида $f_\delta(x) = \text{round}(f(x), 5)$, т.е. округление значения функции f до пятого знака после запятой. На рисунке 1 показана сходимость детерминированного алгоритма

Франка-Вульфа с аппроксимацией нулевого порядка. У алгоритм Франка-Вульфа с JAGUAR (алгоритм 5) результаты лучше, чем у базовых алгоритмов. Это наблюдение подтверждает теоретические выводы.

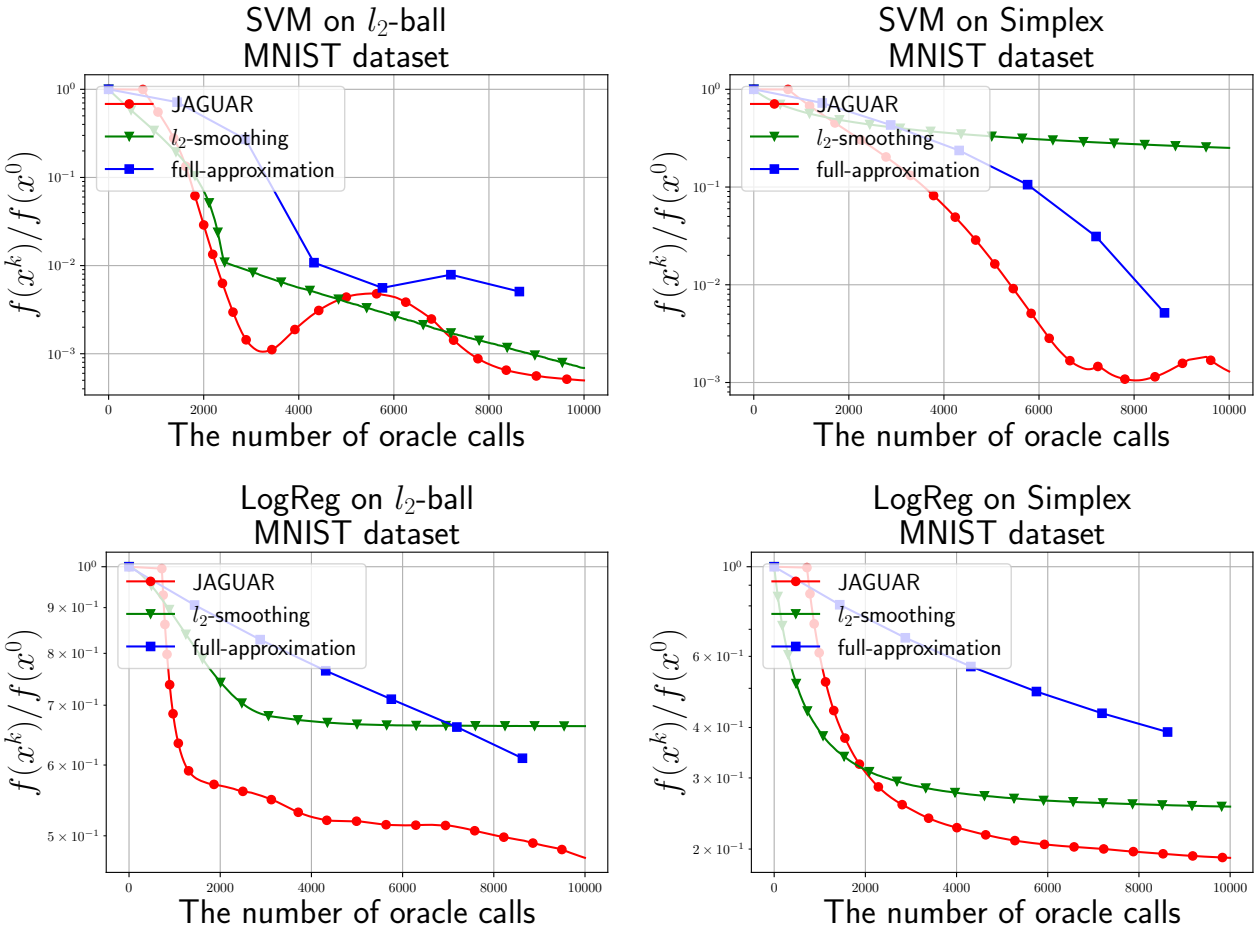

Рис. 1: Детерминированный алгоритм Франка-Вульфа.

4.3 Стохастический алгоритм Франка-Вульфа

В этом разделе рассматривается стохастический шум вида $f_\delta(x, \xi) = f(x) + \xi; \xi \sim \mathcal{N}(0, 0.1)$. На рисунке 2 показана сходимость стохастического алгоритма Франка-Вульфа с аппроксимацией нулевого порядка. Теоретические выводы подтверждаются наблюдениями. Алгоритм Франка-Вульфа с JAGUAR (алгоритм 7) устойчив к шуму и превосходит базовые алгоритмы.

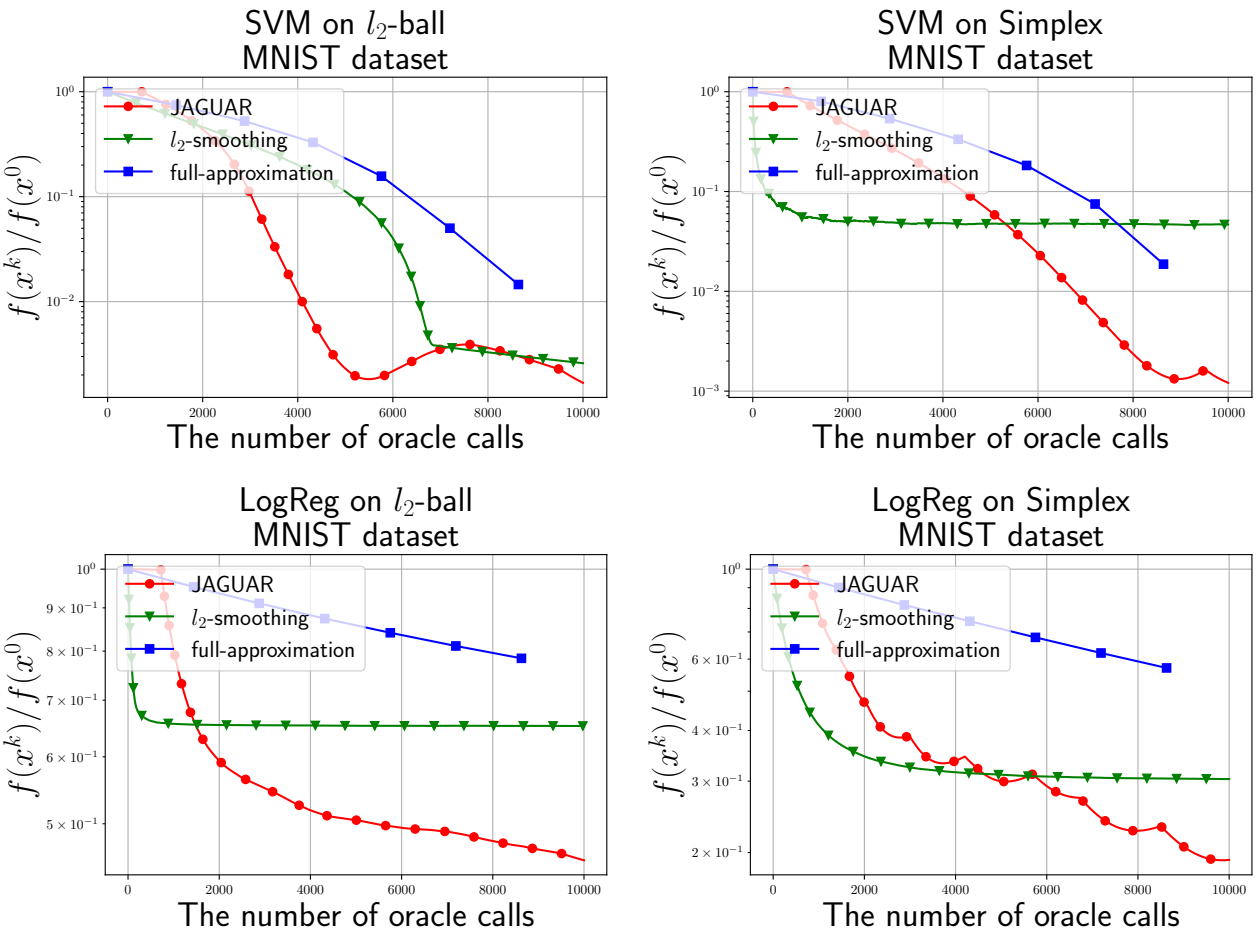

Рис. 2: Стохастический алгоритм Франка-Вульфа.

5 Заключение

В данной работе представлен алгоритм JAGUAR - новый метод аппроксимации градиента, разработанный для решения задач оптимизации «черного ящика», использующий память о предыдущих итерациях для оценки истинного градиента с высокой точностью, требуя при этом всего $\mathcal{O}(1)$ вызовов оракула. Исследование содержит строгие теоретические доказательства и обширную экспериментальную проверку, демонстрируя JAGUAR превосходную производительность как в детерминированных, так и в стохастических условиях. Ключевым вкладом является доказательство сходимости теорем для алгоритма Франка-Вульфа, устанавливающих скорость сходимости. Экспериментальные результаты показывают, что JAGUAR превосходит базовые методы в задачах оптимизации SVM и логистической регрессии. Полученные результаты подчеркивают эффективность и точность JAGUAR, что делает его перспективным подходом для будущих исследований и приложений в области оптимизации нулевого порядка.

Список литературы

[1] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[2] Larry J LeBlanc, Richard V Helgason, and David E Boyce. Improved efficiency of the frank-wolfe algorithm for convex network programs. *Transportation Science*, 19(4):445–462, 1985.

[3] Martin Jaggi. Sparse convex optimization methods for machine learning. 2011.

[4] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[5] Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

[6] Donald Goldfarb, Garud Iyengar, and Chaoxu Zhou. Linear convergence of stochastic frank wolfe variants. In *Artificial Intelligence and Statistics*, pages 1066–1074. PMLR, 2017.

[7] Ali Dadras, Karthik Prakhya, and Alp Yurtsever. Federated frank-wolfe algorithm. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*, 2022.

[8] Robert M Freund, Paul Grigas, and Rahul Mazumder. An extended frank–wolfe method with "in-face" directions, and its application to low-rank matrix completion. *SIAM Journal on optimization*, 27(1):319–346, 2017.

[9] Simon Lacoste-Julien, Martin Jaggi, Mark Schmidt, and Patrick Pletscher. Block-coordinate frank-wolfe optimization for structural svms. In *International Conference on Machine Learning*, pages 53–61. PMLR, 2013.

[10] Yu-Xiang Wang, Veeranjaneyulu Sadhanala, Wei Dai, Willie Neiswanger, Suvrit Sra, and Eric Xing. Parallel and distributed block-coordinate frank-

wolfe algorithms. In *International Conference on Machine Learning*, pages 1548–1557. PMLR, 2016.

[11] Anton Osokin, Jean-Baptiste Alayrac, Isabella Lukasewitz, Puneet Dokania, and Simon Lacoste-Julien. Minding the gaps for block frank-wolfe optimization of structured svms. In *international conference on machine learning*, pages 593–602. PMLR, 2016.

[12] Sashank J Reddi, Suvrit Sra, Barnabás Póczos, and Alex Smola. Stochastic frank-wolfe methods for nonconvex optimization. In *2016 54th annual Allerton conference on communication, control, and computing (Allerton)*, pages 1244–1251. IEEE, 2016.

[13] Mingrui Zhang, Zebang Shen, Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. One sample stochastic frank-wolfe. In *International Conference on Artificial Intelligence and Statistics*, pages 4012–4023. PMLR, 2020.

[14] Haihao Lu and Robert M Freund. Generalized stochastic frank–wolfe algorithm with stochastic "substitute" gradient for structured convex optimization. *Mathematical Programming*, 187(1):317–349, 2021.

[15] Ben Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. Learning structured prediction models: A large margin approach. In *Proceedings of the 22nd international conference on Machine learning*, pages 896–903, 2005.

[16] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26, 2017.

[17] Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17:527–566, 2017.

[18] Krzysztof Choromanski, Mark Rowland, Vikas Sindhwani, Richard Turner, and Adrian Weller. Structured evolution with compact architectures for scalable policy optimization. In *International Conference on Machine Learning*, pages 970–978. PMLR, 2018.

[19] Maryam Fazel, Rong Ge, Sham Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *International conference on machine learning*, pages 1467–1476. PMLR, 2018.

[20] Xiangru Lian, Yijun Huang, Yuncheng Li, and Ji Liu. Asynchronous parallel stochastic gradient for nonconvex optimization. *Advances in neural information processing systems*, 28, 2015.

[21] Yu. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. doi:10.1137/100802001. URL https://doi.org/10.1137/100802001.

[22] John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2): 674–701, 2012.

[23] Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1):1–38, 2014.

[24] Stephen J Wright. Coordinate descent algorithms. *Mathematical programming*, 151(1):3–34, 2015.

[25] Yurii Nesterov and Sebastian U. Stich. Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM Journal on Optimization*, 27(1):110–123, 2017. doi:10.1137/16M1060182. URL https://doi.org/10.1137/16M1060182.

[26] Alexander Gasnikov, Anton Novitskii, Vasilii Novitskii, Farshed Abdukhakimov, Dmitry Kamzolov, Aleksandr Beznosikov, Martin Takac,

Pavel Dvurechensky, and Bin Gu. The power of first-order smooth optimization for black-box non-smooth problems. In *International Conference on Machine Learning*, pages 7241–7265. PMLR, 2022.

[27] Alexander Gasnikov, Darina Dvinskikh, Pavel Dvurechensky, Eduard Gorbunov, Aleksander Beznosikov, and Alexander Lobanov. Randomized gradient-free methods in convex optimization. *arXiv preprint arXiv:2211.13566*, 2022.

[28] Alexander Gasnikov, Anastasia Lagunovskaya, Ilnura Usmanova, and Fedor Fedorenko. Gradient-free proximal methods with inexact oracle for convex stochastic nonsmooth optimization problems on the simplex. *Automation and Remote Control*, 77:2018–2034, 2016.

[29] Arya Akhavan, Evgenii Chzhen, Massimiliano Pontil, and Alexandre Tsybakov. A gradient estimator via l1-randomization for online zero-order optimization with two point feedback. *Advances in Neural Information Processing Systems*, 35:7685–7696, 2022.

[30] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[31] Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *The Journal of Machine Learning Research*, 18(1):1703–1713, 2017.

[32] Eduard Gorbunov, Pavel Dvurechensky, and Alexander Gasnikov. An accelerated method for derivative-free smooth stochastic convex optimization. *SIAM Journal on Optimization*, 32(2):1210–1238, 2022. doi:10.1137/19M1259225. URL https://doi.org/10.1137/19M1259225.

[33] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013.

[34] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. *Advances in neural information processing systems*, 27, 2014.

[35] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International conference on machine learning*, pages 2613–2621. PMLR, 2017.

[36] Filip Hanzely, Konstantin Mishchenko, and Peter Richtárik. Sega: Variance reduction via gradient sketching. *Advances in Neural Information Processing Systems*, 31, 2018.

[37] Francis Bach and Vianney Perchet. Highly-smooth zero-th order online optimization. In *Conference on Learning Theory*, pages 257–283. PMLR, 2016.

[38] Arya Akhavan, Massimiliano Pontil, and Alexandre Tsybakov. Exploiting higher order smoothness in derivative-free optimization and continuous bandits. *Advances in Neural Information Processing Systems*, 33:9017–9027, 2020.

[39] Andrej Risteski and Yuanzhi Li. Algorithms and matching lower bounds for approximately-convex optimization. *Advances in Neural Information Processing Systems*, 29, 2016.

[40] Lev Bogolubsky, Pavel Dvurechenskii, Alexander Gasnikov, Gleb Gusev, Yurii Nesterov, Andrei M Raigorodskii, Aleksey Tikhonov, and Maksim Zhukovskii. Learning supervised pagerank with gradient-based and gradient-free optimization methods. *Advances in neural information processing systems*, 29, 2016.

[41] Anit Kumar Sahu, Dusan Jakovetic, Dragana Bajovic, and Soummya Kar. Distributed zeroth order optimization over random networks: A kiefer-wolfowitz stochastic approximation approach. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 4951–4958. IEEE, 2018.

[42] Anastasia Sergeevna Bayandina, Alexander V Gasnikov, and Anastasia A Lagunovskaya. Gradient-free two-point methods for solving stochastic nonsmooth convex optimization problems with small non-random noises. *Automation and Remote Control*, 79:1399–1408, 2018.

[43] Darina Dvinskikh, Vladislav Tominin, Iaroslav Tominin, and Alexander Gasnikov. Noisy zeroth-order optimization for non-smooth saddle point problems. In *International Conference on Mathematical Optimization Theory and Operations Research*, pages 18–33. Springer, 2022.

[44] Aleksandr Lobanov, Andrew Veprikov, Georgiy Konin, Aleksandr Beznosikov, Alexander Gasnikov, and Dmitry Kovalev. Non-smooth setting of stochastic decentralized convex optimization problem over time-varying graphs. *Computational Management Science*, 20(1):48, 2023.

[45] Aleksandr Lobanov, Anton Anikin, Alexander Gasnikov, Alexander Gornov, and Sergey Chukanov. Zero-order stochastic conditional gradient sliding method for non-smooth convex optimization. *arXiv preprint arXiv:2303.02778*, 2023.

[46] Anit Kumar Sahu, Manzil Zaheer, and Soummya Kar. Towards gradient free and projection free stochastic optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3468–3477. PMLR, 2019.

[47] Hongchang Gao and Heng Huang. Can stochastic zeroth-order frank-wolfe method converge faster for non-convex problems? In *International conference on machine learning*, pages 3377–3386. PMLR, 2020.

[48] Zeeshan Akhtar and Ketan Rajawat. Zeroth and first order stochastic frank-wolfe algorithms for constrained optimization. *IEEE Transactions on Signal Processing*, 70:2119–2135, 2022.

[49] Aleksandr Beznosikov, David Dobre, and Gauthier Gidel. Sarah frank-wolfe:

Methods for constrained optimization with best rates and practical features. *arXiv preprint arXiv:2304.11737*, 2023.

[50] Kaiyi Ji, Zhe Wang, Yi Zhou, and Yingbin Liang. Improved zeroth-order variance reduced algorithms and analysis for nonconvex optimization. In *International conference on machine learning*, pages 3100–3109. PMLR, 2019.

[51] John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

[52] Aleksandr Beznosikov, Abdurakhmon Sadiev, and Alexander Gasnikov. Gradient-free methods with inexact oracle for convex-concave stochastic saddle-point problem. In *International Conference on Mathematical Optimization Theory and Operations Research*, pages 105–119. Springer, 2020.

[53] Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*, 2004.

[54] Alexander V Gasnikov, Ekaterina A Krymova, Anastasia A Lagunovskaya, Ilnura N Usmanova, and Fedor A Fedorenko. Stochastic online optimization. single-point and multi-point non-linear multi-armed bandits. convex and strongly-convex case. *Automation and remote control*, 78:224–234, 2017.

[55] Aleksandr Beznosikov, Vasilii Novitskii, and Alexander Gasnikov. One-point gradient-free methods for smooth and non-smooth saddle-point problems. In *Mathematical Optimization Theory and Operations Research: 20th International Conference, MOTOR 2021, Irkutsk, Russia, July 5–10, 2021, Proceedings 20*, pages 144–158. Springer, 2021.

[56] Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *The Journal of Machine Learning Research*, 21(1):4232–4280, 2020.

[57] Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International conference on machine learning*, pages 427–435. PMLR, 2013.

[58] Sijia Liu, Bhavya Kailkhura, Pin-Yu Chen, Paishun Ting, Shiyu Chang, and Lisa Amini. Zeroth-order stochastic variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

[59] Aleksandr Beznosikov, Eduard Gorbunov, and Alexander Gasnikov. Derivative-free method for composite optimization with applications to decentralized distributed optimization. *IFAC-PapersOnLine*, 53(2):4038–4043, 2020.

[60] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[61] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2 (3):1–27, 2011.

Приложение

A Вспомогательные леммы и факты

A.1 Квадрат нормы суммы

Для всех $x_1, \ldots, x_n \in \mathbb{R}^n$, где $n \in \{2, 4\}$:

$$\|x_1 + x_2 + \cdots + x_n\|^2 \leq n \|x_1\|^2 + \cdots + n \|x_n\|^2 .$$

A.2 Неравенство Коши-Шварца

Для всех $x, y \in \mathbb{R}^d$:

$$\langle x, y \rangle \leq \|x\| \, \|y\| .$$

A.3 Неравенства Юнга-Фенхеля

Для всех $x, y \in \mathbb{R}^d$ и $\beta > 0$:

$$2\langle x, y \rangle \leq \beta^{-1} \|x\|^2 + \beta \|y\|^2.$$

A.4 Лемма о рекурсии

Лемма 5. *Для всех $x \in [0; 1)$ рассматривается функцию*

$$\phi(x) := 1 - (1 - x)^\alpha - \max\{1, \alpha\} x.$$

Тогда для всех $0 \leq x < 1$ и $\alpha \in \mathbb{R}$ можно получить, что $\phi(x) \leq 0$.

Доказательство. □

Лемма 6 (Лемма о рекурсии)**.**

Доказательство. □

B Доказательство сходимости JAGUAR

B.1 Доказательство Леммы 1

B.2 Доказательство Леммы 2

B.3 Доказательство Леммы 3

B.4 Доказательство Леммы 4

C Доказательство сходимости алгоритма Франка-Вульфа с JAGUAR

C.1 Доказательство Теоремы 1

Начнем с того, что выпишем результат из Леммы 2 с $\sigma_f = \sigma_\nabla = 0$ и подставим $\gamma_k = \frac{4}{k+k_0}$:

$$\mathbb{E}\left[\left\|h^{k+1} - \nabla f(x^{k+1})\right\|^2\right] \leq \left(1 - \frac{1}{2d}\right)\mathbb{E}\left[\left\|h^k - \nabla f(x^k)\right\|^2\right] + \frac{32dL^2D^2}{(k+k_0)^2} + L^2\tau^2 + \frac{2\Delta^2}{\tau^2}.$$

Теперь используем Лемму 6 с $\alpha_0 = 0, \beta_0 = 1/2d$; $\alpha_1 = 2, \beta_1 = 32dL^2D^2$; $\alpha_2 = 0, \beta_2 = L^2\tau^2 + \frac{2\Delta^2}{\tau^2}$ и $i^* = 1$:

$$\mathbb{E}\left[\left\|h^k - \nabla f(x^k)\right\|^2\right] = \mathcal{O}\left(dL^2\tau^2 + \frac{d\Delta^2}{\tau^2} + \frac{\max\{d^2L^2D^2, \left\|h^0 - \nabla f(x^0)\right\|^2 \cdot k_0^2\}}{(k+k_0)^2}\right),$$

где $k_0 = (4d \cdot 2)^1 = 8d$. Если $h_0 = \widetilde{\nabla} f_\delta(x^0)$, то получим:

$$\mathbb{E}\left[\left\|h^k - \nabla f(x^k)\right\|^2\right] = \mathcal{O}\left(dL^2\tau^2 + \frac{d\Delta^2}{\tau^2} + \frac{d^2L^2D^2}{(k+8d)^2}\right).$$

На этом доказательство закончено.