Rest-Katyusha: Exploiting the Solution's Structure via Scheduled Restart Schemes

Link:
[Link to publication record in Edinburgh Research Explorer](#)

Document Version:
Publisher's PDF, also known as Version of record

Published In:
Advances in Neural Information Processing Systems 31 (NeurIPS 2018)

Rest-Katyusha: Exploiting the Solution's Structure via Scheduled Restart Schemes

Junqi Tang
School of Engineering
University of Edinburgh
Edinburgh, UK
J.Tang@ed.ac.uk

Mohammad Golbabaee
School of Engineering
University of Edinburgh
Edinburgh, UK
M.Golbabaee@ed.ac.uk

Francis Bach
INRIA - Sierra Project-team
CNRS - ENS - INRIA
Paris, France
Francis.Bach@inria.fr

Mike Davies
School of Engineering
University of Edinburgh
Edinburgh, UK
Mike.Davies@ed.ac.uk

Abstract

We propose a structure-adaptive variant of the state-of-the-art stochastic variance-reduced gradient algorithm Katyusha for regularized empirical risk minimization. The proposed method is able to exploit the intrinsic low-dimensional structure of the solution, such as sparsity or low rank which is enforced by a non-smooth regularization, to achieve even faster convergence rate. This provable algorithmic improvement is done by restarting the Katyusha algorithm according to restricted strong-convexity constants. We demonstrate the effectiveness of our approach via numerical experiments.

1 Introduction

Many applications in supervised machine learning and signal processing share the same goal, which is to estimate the minimizer of a population risk via minimizing the empirical risk $\frac{1}{n}\sum_{i=1}^{n} f_i(a_i, x)$, where $a_i, x \in \mathbb{R}^d$, each f_i is a convex and smooth function [1]. In supervised machine learning, a_i is often referred to as the training data sample, while in signal/image processing applications it is the representation of measurements. In practice the number of data samples or measurements is limited, and from them we attempt to infer $x^\dagger \in \mathbb{R}^d$ which is the unique minimizer of the population risk:

$$x^\dagger = \arg\min_x \mathbb{E}_a \bar{f}(a, x). \tag{1}$$

The ultimate goal is to get a vector $x^\star$ which is a good approximation of $x^\dagger$ from the empirical risk. Since in many interesting applications, the dimension of parameter space d is of the same order or even larger than the number of data samples n, minimizing the empirical risk alone will introduce overfitting and hence leads to poor estimation of the true parameter $x^\dagger$ [2, 3]. In general, avoiding overfitting is a key issue in both machine learning and signal processing, and the most common approach is to add some regularization while minimizing the empirical risk [4, 5, 6]:

$$x^\star \in \arg\min_{x \in \mathbb{R}^d} \left\{ F(x) := f(x) + \lambda g(x) \right\}, \quad f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{2}$$

where for the sake of compactness of notation, we denote $f_i(x) := f_i(a_i, x)$. Each f_i is assumed to be convex and have L-Lipschitz continuous gradient, while the regularization term $g(x)$ is assumed to be a simple convex function and possibly non-smooth.

1.1 Accelerated stochastic variance-reduced optimization

To handle the empirical risk minimization in the "big data" and "big dimension" regimes, stochastic gradient-based iterative algorithms are most often considered. The most basic one is often referred to as *stochastic gradient descent* (SGD) [7, 8], in every iteration of which only one or a few functions f_i are randomly selected, and only their gradients are calculated as an estimation of the full gradient. However, the convergence rate of SGD is sub-linear even when the loss function F is strongly-convex.

To further accelerate the stochastic gradient descent algorithm, researchers have recently developed techniques which progressively reduce the variance of stochastic gradient estimators, starting from SAG [9, 10], SDCA [11], then SVRG [12, 13] and SAGA [14]. Such methods enjoy a linear convergence rate when the cost function F is μ-strongly-convex and each f_i has L-Lipschitz continuous gradients, that is, to achieve an output $\hat{x}$ which satisfies $F(\hat{x}) - F(x^\star) \leq \delta$, the total number of stochastic gradient evaluations needed is $O(n + L/\mu) \log \frac{1}{\delta}$. *Nesterov's acceleration* [15, 16, 17] has also been successfully applied to construct variance-reduced methods which have an accelerated linear-convergence rate [18, 19, 20, 21, 22, 23, 24, 25]:

$$O\left(n + \sqrt{\frac{nL}{\mu}}\right) \log \frac{1}{\delta}. \tag{3}$$

It is worth noting that this convergence rate has been shown to be worst-case optimal [21]. However, all of these algorithms need explicit knowledge of the strong-convexity parameter μ. Very recently, [26] has shown theoretically that it is *impossible* for an accelerated incremental gradient method to achieve this ideal linear rate without the knowledge of μ. Since in general the strong-convexity is hard to be estimated accurately, researchers propose adaptive restart schemes [27, 28, 29, 30, 25, 31] for accelerated first-order methods, either by the means of enforcing monotonicity on functional decay, or by estimating the strong-convexity on the fly.

1.2 Solution's low-dimensional structure, restricted strong convexity, and fast convergence

In many interesting large-scale optimization problems in machine learning, the solution $x^\star$ in (2) has some low-dimensional structure such as sparsity [4], group-sparsity [32], low-rank [33] or piece-wise smoothness [6], enforced by the non-smooth regularization. It is intuitive that an optimal algorithm for this type of problem should take into account and exploit such solution's structure. We believe that, when being utilized properly, this prior information of the solution will facilitate the convergence of an iterative algorithm.

One important theoretical cornerstone is the *modified restricted strong convexity* framework presented by Agarwal et al. [34]. In the context of statistical estimation with high-dimensional data where the usual strong-convexity assumption is vacuous, these authors have shown that the proximal gradient descent method is able to achieve global linear convergence up to a point x which satisfies $\|x - x^\star\|_2 = o(\|x^\star - x^\dagger\|_2)$, the accuracy level of *statistical precision*. Moreover, the results based on this restricted strong-convexity framework indicate that the convergence rate of the proximal gradient become faster when the model complexity of the solution is lower.

Inspired by Agarwal et al. [34], Qu and Xu [35] extend this framework to analyse some variance-reduced stochastic gradient methods such as proximal SVRG [13]. Most recently, based on the same framework, researchers proposed a two-stage APCG algorithm [36], an accelerated coordinate descent method able to exploit the solution's structure for faster convergence.

1.3 This work

In this paper we extend the theoretical framework for randomized first order methods established in [36] to design and analyse a structure-adaptive variant of Katyusha [23]. Our proposed method Rest-Katyusha is a restarted version of the original Katyusha method for non-strongly convex functions, where the restart period is determined by the modified restricted strong-convexity (RSC). The

convergence analysis of Rest-Katyusha algorithm is provided, wherein we prove linear convergence up to a statistical accuracy with an accelerated convergence rate characterized by the RSC property.

Like all other accelerated gradient methods which require the explicit knowledge of strong-convexity parameter to achieve accelerated linear convergence, the vanilla Rest-Katyusha method also need to explicitly know the RSC parameter. We therefore propose a practical heuristic (adaptive Rest-Katyusha) which estimates the RSC parameter on the fly and adaptively tune the restart period, and we show that this adaptive scheme mimics the convergence behavior of the vanilla Rest-Katyusha. Finally we validate the effectiveness of our approach via numerical experiments.

2   Restarted Katyusha Algorithm

The Katyusha algorithm [23] listed in Algorithm 1 is an accelerated stochastic variance-reduced gradient method extended from the linear-coupling framework for constructing accelerated methods [37]. Its main loop (denoted as $\mathcal{A}$ in Algorithm 1) at iteration s is described as the following:

For $k = 0, 1, 2, ..., m$
$$x_{k+1} = \theta z_k + \tfrac{1}{2}\hat{x}^s + (\tfrac{1}{2} - \theta)y_k; \qquad\qquad\qquad \rightarrow \text{Linear coupling}$$
$$\nabla_{k+1} = \nabla f(\hat{x}^s) + \nabla f_i(x_{k+1}) - \nabla f_i(\hat{x}^s); \quad \rightarrow \text{Variance reduced stochastic gradient}$$
$$z_{k+1} = \arg\min_z \tfrac{3\theta L}{2}\|z - z_k\|_2^2 + \langle \nabla_{k+1}, z\rangle + \lambda g(z); \quad \rightarrow \text{Proximal mirror descent}$$
$$y_{k+1} = \arg\min_y \tfrac{3L}{2}\|y - x_{k+1}\|_2^2 + \langle \nabla_{k+1}, y\rangle + \lambda g(y); \rightarrow \text{Proximal gradient descent}$$

The output sequence of $\mathcal{A}$ is defined as $\hat{x}^{s+1} = \frac{1}{m}\sum_{j=1}^m y_j$, $y^{s+1} = y_m$, $z^{s+1} = z_m$. It is one of the state-of-the-art methods for empirical risk minimization and matches the complexity lower-bound for minimizing smooth-convex finite-sum functions, proven by Lan and Zhou [21]. Most notably, it is a primal method which directly[1] accelerates stochastic variance-reduction methods. To achieve acceleration in the sense of Nesterov, Katyusha introduces the three-point coupling strategy which includes a combination of Nesterov's momentum and a stabilizing negative momentum which cancels the effect of noisy updates due to stochastic gradients. However, its accelerated linear convergence is only established when the regularization term $g(x)$ is strongly-convex, and fails to benefit from the strong convexity from the data-fidelity term [31], or the intrinsic restricted strong-convexity [36].

Algorithm 1 Katyusha (x^0, m, S, L)	**Algorithm 2** Rest-Katyusha $(x^0, \mu_c, S_0, \beta, T, L)$
Initialize: $y^0 = z^0 = \hat{x}^0$;	**Initialize:** $m = 2n$, $S = \left\lceil \beta\sqrt{32 + \frac{24L}{m\mu_c}} \right\rceil$;
for $s = 0, \ldots, S-1$ **do**	First stage —- warm start:
$\quad \theta \leftarrow \frac{2}{s+4}$, calculate $\nabla f(\hat{x}^s)$,	$x^1 =$ Katyusha (x^0, m, S_0, L)
$\quad (\hat{x}^{s+1}, y^{s+1}, z^{s+1})$	Second stage —- exploit the restricted strong-
$\quad = \mathcal{A}(\hat{x}^s, y^s, z^s, \theta, \nabla f(\hat{x}^s), m)$	convexity via periodic restart:
	for $t = 1, ..., T$ **do**
end for	$\quad x^{t+1} =$ Katyusha (x^t, m, S, L)
Output: $\hat{x}^S$	**end for**

Restart to rescue: it is well-known that if the cost function $F(x)$ is μ-strongly convex, one can periodically restart the accelerated full gradient method [16], and improve it from a sublinear convergence rate $F(x^k) - F^\star \leq \frac{4L\|x^0 - x^\star\|_2^2}{k^2}$ to a linearly convergent algorithm. For instance if we set $k = \left\lceil 4\sqrt{L/\mu} \right\rceil$, then one can show that the suboptimality can be reduced by $\frac{1}{4}$:

$$F(x^k) - F^\star \leq \frac{4L\|x^0 - x^\star\|_2^2}{k^2} \leq \frac{4L[F(x^0) - F^\star]}{\mu k^2} \leq \frac{1}{4}[F(x^0) - F^\star]. \qquad (4)$$

Then we can recursively apply this statement (algorithmically speaking, we restart the algorithm every $\left\lceil 4\sqrt{L/\mu} \right\rceil$ iteration), and only $k \geq \left\lceil 4\sqrt{\frac{L}{\mu}} \right\rceil \log_4 \frac{1}{\delta}$ iterations are needed to make $F(x^k) - F^\star \leq \delta$,

---

[1]On the other hand, one can indirectly accelerate SVRG/SAGA via Catalyst [38].

and hence an accelerated linear rate is achieved. The restart scheme has been recently applied to improve the convergence of the accelerated coordinate descent method [39, 30] and accelerated variance-reduced dual-averaging method [25] for strongly-convex functions.

Inspired by Nesterov [16] we first propose the Katyusha method with periodic restarts, and meanwhile demonstrate that when the restart period is appropriately chosen, the proposed method is able to exploit the restricted strong-convexity property to achieve an accelerated linear convergence, even when the cost function itself is not strongly-convex. We propose to warm start the algorithm prior to the periodic restart stage, by running the Katyusha algorithm for a number of epochs, which in theory should be proportional to the suboptimality of the starting point x^0. We present our Rest-Katyusha method as Algorithm 2.

3 Convergence analysis of Rest-Katyusha

3.1 Generic assumptions

We start by listing out the assumptions we may engage with in our analysis:

A. 1. *(Decomposable regularizer) [34] Given a orthogonal subspace pair $(\mathcal{M}, \mathcal{M}^\perp)$ in $\mathbb{R}^d$, $g(.)$ is decomposable which means:*

$$g(a+b) = g(a) + g(b), \forall a \in \mathcal{M}, b \in \mathcal{M}^\perp. \tag{5}$$

In this paper we focus on cases where the regularizer is decomposable, which includes many popular regularization which can enforce low-dimensional structure, such as ℓ_1 norm, $\ell_{1,2}$ norm and nuclear norm penalty. The subspace $\mathcal{M}$ is named the *model subspace*, while its orthogonal complement $\mathcal{M}^\perp$ is called *perturbation subspace*. Similar notion of decomposition would extend the scope of this work to more general gauge functions $g(.)$, such as the so-called *analysis priors*, e.g., total variation regularization (for more details see Vaiter et al. [40]).

A. 2. *(Restricted strong convexity) [34] The function $f(.)$ satisfies restricted strong convexity with respect to $g(.)$ with parameters (γ, τ) if the following inequality holds true:*

$$f(x) - f(x^\star) - \langle \nabla f(x^\star), x - x^\star \rangle \geq \frac{\gamma}{2} \|x - x^\star\|_2^2 - \tau g^2(x - x^\star), \quad \forall x \in \mathbb{R}^d. \tag{6}$$

In [34], γ is referred as the *lower curvature parameter*, while τ is named the *tolerance parameter*. It is clear that if $\tau = 0$, **A.2** reduces to usual strong-convexity assumption. While in the high-dimensional setting, the strong-convexity does not hold, but it has been shown in literature that such milder assumption of RSC does hold in many situations. This notion of RSC distinguishes from other forms of weak strong-convexity assumption based on the Polyak-Lojasiewicz inequality [41] for the purpose of this work, because it encodes the direction-restricting effect of the regularization, and hence has been shown to have a direct connection with the low-dimensional structure of $x^\star$. Next we define a crucial property for our structure-driven analysis, which is called the *subspace compatibility*:

Definition 3.1. *[34] With predefined $g(x)$, we define the subspace compatibility of a model subspace $\mathcal{M}$ as:*

$$\Phi(\mathcal{M}) := \sup_{v \in \mathcal{M} \setminus \{0\}} \frac{g(v)}{\|v\|_2}, \tag{7}$$

when $\mathcal{M} \neq \{0\}$.

The subspace compatibility $\Phi(\mathcal{M})$ captures the model complexity of subspace $\mathcal{M}$. For example if $g(.) = \|.\|_1$ and $\mathcal{M}$ is a subspace which is on a s-sparse support in $\mathbb{R}^d$, we will have $\Phi(\mathcal{M}) = \sqrt{s}$.

A. 3. *Each $f_i(.)$ has L-Lipschitz continuous gradient:*

$$\|\nabla f_i(x) - \nabla f_i(x')\|_2 \leq L\|x - x'\|_2, \forall x, x' \in \mathbb{R}^d. \tag{8}$$

This form of smoothness assumption is classic for variance-reduced stochastic gradient methods.

4

A. 4. *The regularization parameter λ and $x^\dagger$ satisfies:*

$$\lambda \geq (1 + \frac{1}{c})g^*(\nabla f(x^\dagger)), \tag{9}$$

with constant $c \geq 1$.

Assumption **A.4** with the choice of $c = 1$ is the fundamental assumption of the analytical framework developed by Negahban et al. [3]. We relax the requirement to $c \geq 1$ for more general results. It is seemly a sophisticated and demanding assumption but indeed is reasonable and suits well the purpose of this work, which is to develop fast algorithms to speedily solve structured problems (which is always the result of sufficient regularization). Moreover, recall that the goal of finding the solution $x^\star$ via optimizing the regularized empirical risk is to get a meaningful approximation of the true parameter $x^\dagger$ which is the unique minimizer of the population risk. Especially in the high dimensional setting where $d > n$, the choice of regularization is rather important since there is no good control over the statistical error $\|x^\dagger - x^\star\|$ for an arbitrarily chosen λ. Because of this issue, in this work we only focus on the "meaningful" regularized ERM problems which are able to provide trustworthy approximation. Similar to **A.4**, Negahban et al.[3] has shown that $\lambda \geq 2g^*(\nabla f(x^\dagger))$ provides a sufficient condition to bound the statistical error $\|x^\star - x^\dagger\|_2^2$:

Proposition 3.2. *[3, Theorem 1, informal] Under **A.1**, **A.2**, **A.4**, with $c = 1$, if furthermore the curvature parameter γ, tolerance parameter τ and the subspace compatibility $\Phi(\mathcal{M})$ satisfy $\tau\Phi^2(\mathcal{M}) < \frac{\gamma}{64}$, then for any optima $x^\star$, the following inequality holds:*

$$\|x^\star - x^\dagger\|_2^2 \leq O\left(\frac{\lambda^2}{\gamma^2}\Phi^2(\mathcal{M}) + \frac{\lambda}{\gamma}g(x^\dagger_{\mathcal{M}^\perp})\right), \tag{10}$$

where $O(.)$ hides deterministic constants for the simplicity of notation.

Such a bound reveals desirable properties of the regularized ERM when the range of λ satisfies assumption **A.4**. For instance, if $x^\dagger$ is the s-sparse ground truth vector of a noisy linear measurement system $y = Ax^\dagger + w$, where w denotes the zero-mean sub-Gaussian noise (with variance σ^2) and the measurement matrix A satisfies a certain restricted eigenvalue condition [3, 42], and we use a Lasso estimator $x^\star \in \arg\min_x \frac{1}{2n}\|Ax - y\|_2^2 + \lambda\|x\|_1$. In such case, let $\mathcal{M}$ be a subspace in $\mathbb{R}^d$ on s-sparse support where $x^\dagger \in \mathcal{M}$ and hence $g(x^\dagger_{\mathcal{M}^\perp}) = 0$, this proposition implies that:

$$\|x^\star - x^\dagger\|_2^2 \leq O\left(\frac{\lambda^2 s}{\gamma^2}\right) \approx O\left(\frac{\sigma^2}{\gamma^2}\frac{s\log d}{n}\right), \tag{11}$$

which implies the optimal convergence of the statistical error in terms of sample size and dimension for M-estimators. The details of this claim are presented in [3, Corollary 2].

3.2 Main results

Base on the assumption of the restricted strong convexity on $f(.)$ w.r.t $g(.)$, and also with the definition of subspace compatibility, one can further derive a more expressive form of RSC, which is named *Effective RSC* [34] which has a directly link to the structure of solution.

Lemma 3.3. *(Effective RSC) [36, Lemma 3.3] Under **A.1**, **A.2**, **A.4**, while x satisfies $F(x) - F(x^\star) \leq \eta$ for a given value $\eta > 0$ and any minimizer $x^\star$, with $\varepsilon := 2\Phi(\mathcal{M})\|x^\dagger - x^\star\|_2 + 4g(x^\dagger_{\mathcal{M}^\perp})$ we have:*

$$F(x) - F^\star \geq \mu_c\|x - x^\star\|_2^2 - 2\tau(1 + c)^2v^2, \tag{12}$$

where $\mu_c = \frac{\gamma}{2} - 8\tau(1 + c)^2\Phi^2(\mathcal{M})$ and $v = \frac{\eta}{\lambda} + \varepsilon$.

Here we refer μ_c as the effective restricted strong convexity parameter, which will provide us a direct link between the convergence speed of an algorithm and the low-dimensional structure of the solution. Note that this lemma relaxes the condition on λ in [34, Lemma 11], which is restricted to $c = 1$. Our main theorem is presented as the following:

Theorem 3.4. *Under **A.1** - **4**, denote $\varepsilon := 2\Phi(\mathcal{M})\|x^\dagger - x^\star\|_2 + 4g(x^\dagger_{\mathcal{M}^\perp})$, $\mathcal{D}(x^0, x^\star) := 16(F(x^0) - F^\star) + \frac{6L}{n}\|x^0 - x^\star\|_2^2$, $\mu_c = \frac{\gamma}{2} - 8\tau(1 + c)^2\Phi^2(\mathcal{M})$, if we run Rest-Katyusha with*

$$S_0 \geq \left\lceil \left(1 + \tfrac{2}{\rho\lambda}\right) \sqrt{\tfrac{6L\tau(1+c)^2 \mathcal{D}(x^0, x^\star)}{8n\mu_c + 3L}} \right\rceil, \quad S = \left\lceil \beta \sqrt{32 + \tfrac{12L}{n\mu_c}} \right\rceil \text{ with } \beta \geq 2, \text{ then the following}$$

inequality holds:

$$\mathbb{E}[F(x^{T+1}) - F^\star] \leq \max\left\{ \varepsilon, \left(\frac{1}{\beta^2}\right)^T \frac{\mathcal{D}(x^0, x^\star)}{(S_0 + 4)^2} \right\} \tag{13}$$

with probability at least $1 - \rho$.

Corollary 3.5. *Under the same assumptions, parameter choices and notations as Theorem 3.4, the total number of stochastic gradient evaluations required by Rest-Katyusha to get an $\delta > \varepsilon$-accuracy is:*

$$O\left(n + \sqrt{\frac{nL}{\mu_c}}\right) \log \frac{1}{\delta} + O(n)S_0. \tag{14}$$

Proof technique. We extend the proof technique of Agarwal et al. [34] to the proximal gradient descent and also Qu and Xu [35] for SVRG which are both based on applying induction statements to roll up the residual term of (12) which is the second term at the RHS. The complete proofs of Theorem 3.4 and Corollary 3.5 can be found in the supplementary material.

Accelerated linear convergence. Under the RSC assumption, Theorem 3.4 and Corollary 3.5 demonstrate a local accelerated linear convergence rate of Rest-Katyusha up to a statistical accuracy $\delta > \varepsilon$. We derive this result based on extending the framework provided by Agarwal et al. [34], by which they established fast structure-dependent linear convergence of proximal gradient descent method up to a statistical accuracy of $\delta > \varepsilon$. To the best of our knowledge, this is the first structure-adaptive convergence result for an accelerated incremental gradient algorithm. Note that, this result can be trivially extended to a global accelerated linear convergence result (with $S_0 = S$) with the same setting of Agarwal et al. [34] where a side constraint $g(x) \leq R$ for some radii R is added to restrict the early iterations with additional re-projections unto this constraint set[2].

Structure-adaptive convergence. The effective RSC $\mu_c = \tfrac{\gamma}{2} - 8\tau(1 + c)^2 \Phi^2(\mathcal{M})$ links the convergence speed of Rest-Katyusha with the intrinsic low-dimensional structure of the solution which is due to the regularization. For instance, if $F(x) := \tfrac{1}{2n}\|Ax - b\|_2^2 + \lambda\|x\|_1$, $\|x^\star\|_0 = s$ and (**A.4**) holds $c = 1$, then we have $\mu_c = \tfrac{\gamma}{2} - 32\tau s$, meanwhile for a wide class of random design matrices we have $\tau = O(\tfrac{\log d}{n})$ and $\gamma > 0$. More specifically, if the rows of the random design matrix A are drawn i.i.d. from $\mathcal{N}(0, \Sigma)$ with covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$ which has largest singular value $r_{\max}(\Sigma)$ and smallest singular value $r_{\min}(\Sigma)$, then $\gamma \geq \tfrac{r_{\min}(\Sigma)}{16}$ and $\tau \leq r_{\max}(\Sigma)\tfrac{81 \log d}{n}$ with high probability as shown by Raskutti et al. [42].

High probability statement. Since our proofs utilize the effective RSC which holds in a neighborhood of $x^\star$ as demonstrated in Lemma 3.3, we need to bound the functional suboptimality $F(x^t) - F^\star$ in the worst case instead of in expectation. Hence inevitably the Markov inequality has to be applied to provide the convergence statement with high probability (details can be found in the main proof).

Optimizing the choice of β. Theorem 3.4 shows that the complexity of the main loop of Rest-Katyusha is $\left\lceil \beta\sqrt{32 + 12L/(n\mu_c)} \right\rceil \log_{\beta^2} \tfrac{1}{\delta}$, which suggest a trade-off between the choice of β and the total computation. With some trival computation one can derive that in theory the best choice of β is exactly the Euler's number (≈ 2.7). Numerically, we observe that slightly larger choice of β often provides better performance in practice (illustrative examples can be found in supplemental material).

4 Adaptive Rest-Katyusha

Motivated by the theory above, we further propose our practical adaptive restart heuristic of Rest-Katyusha which is able to estimate the effective RSC on the fly. Based on the convergence theory, we observe that, with the choice of restart period $S = \left\lceil \beta\sqrt{32 + 12L/(n\mu_0)} \right\rceil$ with a conservative

---

[2]In [34], a side constraint is manually added to the regularized ERM problem, hence in their setting, the effective restricted strong-convexity is valid globally. They provide global linear convergence result of proximal gradient descent (with additional re-projection steps) at a cost of additional side-constraints.

estimate $\mu_0 \le \mu_c$, then we are always guaranteed to have:

$$\mathbb{E}_{\xi_t \backslash \xi_{t-1}}[F(x^{t+1}) - F^\star] \le \frac{1}{\beta^2}[F(x^t) - F^\star], \tag{15}$$

due to the fact that an underestimation of the RSC will leads to a longer restart period that we actually need[3]. The intuition behind our adaptive restart heuristic is: if we overestimate μ_c, the above inequality will be violated. Hence an adaptive estimation of μ_c can be achieved via a convergence speed check. However the above inequality cannot be evaluated directly in practice since it is in expectation and demands the knowledge of $F^\star$. In [29, Prop. 4], it has been shown that with the composite gradient map:

$$\mathcal{T}(x) = \arg\min_q \frac{L}{2}\|x - q\|_2^2 + \langle \nabla f(x), q - x \rangle + \lambda g(q), \tag{16}$$

the value of $F(x) - F^\star$ can be lower bounded:

$$F(x) - F^\star \ge \|\mathcal{T}(x) - x\|_2^2, \tag{17}$$

and also it can be upper bounded by $O(\|\mathcal{T}(x) - x\|_2^2)$ if local quadratic growth is assumed, which reads:

$$\exists \alpha > 0,\ r > 0,\ F(x) - F^\star \ge \alpha\|x - x^\star\|_2^2, \forall x\ s.t.\ \|x - x^\star\|_2^2 < r. \tag{18}$$

The details of this result can be found in [29]. Hence in our adaptive restart heuristic we check the convergence speed via evaluating the composite gradient map at the snapshot points where full gradients have already been calculated. Because of this, the only main additional computational overhead of this adaptive restart scheme is the proximal operation of $g(.)$ at the restart points.

Algorithm 3 Adaptive Rest-Katyusha $(x^0, \mu_0, S_0, \beta, T, L)$

Initialize: Epoch length $m = 2n$; Initial restart period $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$;

$x^1 =$ Katyusha (x^0, m, S_0, L)
Calculate the composite gradient map:
$\mathcal{T}(x^1) = \arg\min_x \frac{L}{2}\|x - x^1\|_2^2 + \langle \nabla f(x^1), x - x^1 \rangle + \lambda g(x)$.
for $t = 1, \ldots, T$ **do**
    $x^{t+1} =$ Katyusha (x^t, m, S, L)
    ——Track the convergence speed via the composite gradient maps:
      $\mathcal{T}(x^{t+1}) = \arg\min_x \frac{L}{2}\|x - x^{t+1}\|_2^2 + \langle \nabla f(x^{t+1}), x - x^{t+1} \rangle + \lambda g(x)$.
    —— Update the estimate of RSC and adaptively tune the restart period
      if $\|\mathcal{T}(x^{t+1}) - x^{t+1}\|_2^2 \le \frac{1}{\beta^2}\|\mathcal{T}(x^t) - x^t\|_2^2$

      then $\mu_0 \leftarrow 2\mu_0$, **else** $\mu_0 \leftarrow \mu_0/2$. $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$

end for

The adaptive Rest-Katyusha method is presented in Algorithm 3. We highlight the heuristic estimating procedure for RSC parameter in the orange lines, which is additional to the original Katyusha algorithm. The algorithm start with an initial guess μ_0 and correspondingly the restart period S, meanwhile we calculate the composite gradient map $\mathcal{T}(x^1)$ at x^1 and record the value of $\|\mathcal{T}(x^1) - x^1\|_2^2$ which we use as the estimation of $F(x^1) - F^\star$ (and so on). Then after S outer-loops, we restart the algorithm and meanwhile and evaluate again the composite gradient map. If $\|\mathcal{T}(x^2) - x^2\|_2^2 \ge \frac{1}{\beta^2}\|\mathcal{T}(x^1) - x^1\|_2^2$, then we suspect that the RSC parameter has been overestimated, and hence we reduce μ_0 by a half, otherwise we double the estimation. We also update the restart period by $S = \left\lceil \beta\sqrt{32 + 12L/(n\mu_0))} \right\rceil$ with the modified μ_0. The forthcoming iterations follow the same updating rule as described.

---

[3]An inaccurate estimate of the RSC will lead to a compromised convergence rate. Detailed discussion and analysis of Rest-Katyusha with a rough RSC estimate can be found in the Appendix.

5 Numerical experiments

In this section we describe our numerical experiments on our proposed algorithm Rest-Katyusha (Alg.2) and also the adaptive Rest-Katyusha (Alg.3). We focus on the Lasso regression task:

$$x^\star \in \arg\min_{x \in \mathbb{R}^d} \left\{ F(x) := \frac{1}{2n} \|Ax - b\|_2^2 + \lambda \|x\|_1 \right\}. \tag{19}$$

To enforce sparsity on regression parameter we use the ℓ_1 penalty with various degrees of regularization parameters chosen from the set $\lambda \in \{1 \times 10^p, 2 \times 10^p, 5 \times 10^p, p \in \mathbb{Z}\}$. For comparison, the performance of (proximal) SVRG and the original Katyusha method is also shown in the plots. We run all the algorithms with their theoretical step sizes for Madelon and REGED dataset, while for RCV1 dataset we adopt minibatch scheme for all the algorithms and grid-search the step sizes which optimize these algorithms' performance.

Table 1: Datasets for the experiments and minibatch size we adopt for the algorithms

DATA SET	SIZE (n, d)	MINIBATCH	REF.
(A) MADELON	(2000, 500)	1	[43]
(B) RCV1	(20242, 47236)	80	[43]
(C) REGED	(500, 999)	1	[44]

Figure 1: Lasso experiments
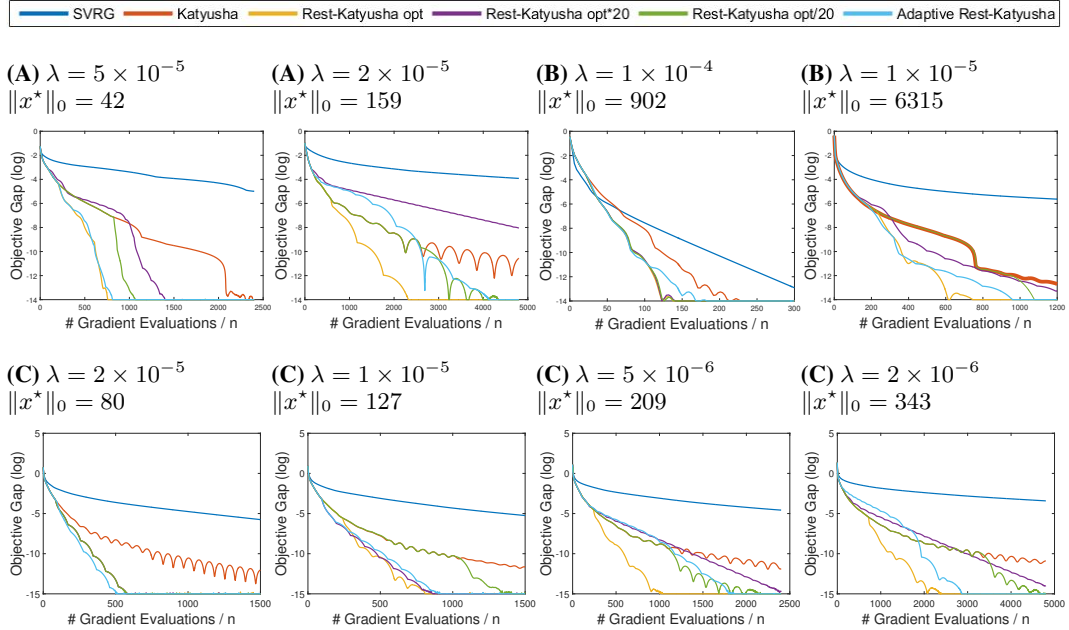
In all our experiments we set $\beta = 5$ and $S_0 = S$ for convenience. We first do a grid-search on the estimate of μ_c for Rest-Katyusha which provides the best convergence performance, and denote it as "Rest-Katyusha opt" in the plots. Meanwhile we also run Rest-Katyusha with RSC estimation which is 20 times larger or smaller than the optimal one, where we denote as "Rest-Katyusha opt*20" and "Rest-Katyusha opt/20" respectively. At the 5th plot in Figure 1 the curves for Rest-Kat opt, opt*20 and opt/20 are indistinguishable which shows that in these particular experiments their performance are almost identical. For the adaptive Rest-Katyusha we fix our starting estimate of μ_c as 10^{-5} throughout all the experiments.

From these experiments we observe that as our theory has predicted, the Rest-Katyusha achieves accelerated linear convergence even when there is no explicit strong-convexity in the cost function (RCV1 and REGED dataset), and the convergence speed has a direct relationship with the sparsity

of solution. For the lasso experiments while the solution is sparser, the linear convergence speed of Rest-Katyusha indeed become faster. Meanwhile when we run Rest-Katyusha with an inaccurate RSC estimate, we still observe a compromised linear convergence, as predicted by our theory. In all the experiments, we have observe that the adaptive Rest-Katyusha indeed achieves a good estimation of the RSC parameter and properly adapts the choice of restart period automatically on the fly, hence its performance is often comparable with the best tuned Rest-Katyusha. As the experimental results shown in [34, 35, 36], the linear convergence we observe is towards an arbitrary accuracy instead of a threshold nearby the solution. This conservative aspect of the theory is inherently due to the artifact of the RSC framework [34] and we include the extension for arbitrary accuracy regime as our future work.

6 Conclusion

We developed a restart variant of the Katyusha algorithm for regularized empirical risk minimization tasks, which is provably able to actively exploit the intrinsic low-dimensional structure of the solution for the acceleration of convergence. Based on the convergence result we further constructed an adaptive restart heuristic which aimed at estimating the RSC parameter on the fly and adaptively tune the restart period. The efficiency of this approach is validated through numerical experiments. In future work, we aim to develop more refined and provably-good adaptive restart schemes for Rest-Katyusha algorithm to further exploit the solution's structure for acceleration.

7 Acknowledgements

References

[1] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[2] Martin J Wainwright. Structured regularizers for high-dimensional problems: Statistical and computational issues. *Annual Review of Statistics and Its Application*, 1:233–253, 2014.

[3] Sahand N Negahban, Pradeep Ravikumar, Martin J Wainwright, and Bin Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science*, pages 538–557, 2012.

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

[5] Robert Tibshirani, Martin Wainwright, and Trevor Hastie. *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC, 2015.

[6] Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[7] Tong Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116. ACM, 2004.

[8] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMP-STAT'2010*, pages 177–186. Springer, 2010.

[9] Nicolas L. Roux, Mark Schmidt, and Francis R. Bach. A stochastic gradient method with an exponential convergence _rate for finite training sets. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2663–2671. Curran Associates, Inc., 2012.

[10] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, pages 1–30, 2013.

[11] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.

[12] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

[13] L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[14] A. Defazio, F. Bach, and S. Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654, 2014.

[15] Yurii Nesterov. A method of solving a convex programming problem with convergence rate o (1/k2). In *Soviet Mathematics Doklady*, volume 27, pages 372–376, 1983.

[16] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, UCL, 2007.

[17] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

[18] Qihang Lin, Zhaosong Lu, and Lin Xiao. An accelerated proximal coordinate gradient method. In *Advances in Neural Information Processing Systems*, pages 3059–3067, 2014.

[19] Zeyuan Allen-Zhu, Zheng Qu, Peter Richtárik, and Yang Yuan. Even faster accelerated coordinate descent using non-uniform sampling. In *International Conference on Machine Learning*, pages 1110–1119, 2016.

[20] Shai Shalev-Shwartz and Tong Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *International Conference on Machine Learning*, pages 64–72, 2014.

[21] Guanghui Lan and Yi Zhou. An optimal randomized incremental gradient method. *arXiv preprint arXiv:1507.02000*, 2015.

[22] Yuchen Zhang and Xiao Lin. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 353–361, 2015.

[23] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *arXiv preprint arXiv:1603.05953*, 2016.

[24] Aaron Defazio. A simple practical accelerated method for finite sums. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 676–684. Curran Associates, Inc., 2016.

[25] Tomoya Murata and Taiji Suzuki. Doubly accelerated stochastic variance reduced dual averaging method for regularized empirical risk minimization. In *Advances in Neural Information Processing Systems*, pages 608–617, 2017.

[26] Yossi Arjevani. Limitations on variance-reduction and acceleration schemes for finite sums optimization. In *Advances in Neural Information Processing Systems*, pages 3543–3552, 2017.

[27] B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. *Foundations of computational mathematics*, 15(3):715–732, 2015.

[28] Vincent Roulet and Alexandre d'Aspremont. Sharpness, restart and acceleration. In *Advances in Neural Information Processing Systems*, pages 1119–1129, 2017.

[29] Olivier Fercoq and Zheng Qu. Adaptive restart of accelerated gradient methods under local quadratic growth condition. *arXiv preprint arXiv:1709.02300*, 2017.

[30] Olivier Fercoq and Zheng Qu. Restarting the accelerated coordinate descent method with a rough strong convexity estimate. *arXiv preprint arXiv:1803.05771*, 2018.

[31] Jialei Wang and Lin Xiao. Exploiting strong convexity from data with primal-dual first-order algorithms. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3694–3702, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[32] Francis R Bach. Consistency of the group lasso and multiple kernel learning. *Journal of Machine Learning Research*, 9(Jun):1179–1225, 2008.

[33] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[34] A. Agarwal, S. Negahban, and M. J. Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *The Annals of Statistics*, 40(5):2452–2482, 2012.

[35] Chao Qu and Huan Xu. Linear convergence of svrg in statistical estimation. *arXiv preprint arXiv:1611.01957*, 2016.

[36] Junqi Tang, Francis Bach, Mohammad Golbabaee, and Mike Davies. Structure-adaptive, variance-reduced, and accelerated stochastic optimization. *arXiv preprint arXiv:1712.03156*, 2017.

[37] Zeyuan Allen-Zhu and Lorenzo Orecchia. Linear coupling: An ultimate unification of gradient and mirror descent. *arXiv preprint arXiv:1407.1537*, 2014.

[38] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, pages 3384–3392, 2015.

[39] Olivier Fercoq and Zheng Qu. Restarting accelerated gradient methods with a rough strong convexity estimate. *arXiv preprint arXiv:1609.07358*, 2016.

[40] Samuel Vaiter, Mohammad Golbabaee, Jalal Fadili, and Gabriel Peyré. Model selection with low complexity priors. *Information and Inference: A Journal of the IMA*, 4(3):230–287, 2015.

[41] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

[42] Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Restricted eigenvalue properties for correlated gaussian designs. *Journal of Machine Learning Research*, 11(Aug):2241–2259, 2010.

[43] M. Lichman. UCI machine learning repository, 2013.

[44] Causality workbench team. A genomics dataset, 09 2008.

Rest-Katyusha: Exploiting the Solution's Structure via Scheduled Restart Schemes
(Supplementary Material)

Junqi Tang
School of Engineering
University of Edinburgh
Edinburgh, UK
J.Tang@ed.ac.uk

Mohammad Golbabaee
School of Engineering
University of Edinburgh
Edinburgh, UK
M.Golbabaee@ed.ac.uk

Francis Bach
INRIA - Sierra Project-team
CNRS - ENS - INRIA
Paris, France
Francis.Bach@inria.fr

Mike Davies
School of Engineering
University of Edinburgh
Edinburgh, UK
Mike.Davies@ed.ac.uk

In this supplementary material we include in section A the proof of our main result which establishes the structure-adaptive and accelerated linear convergence rate for Rest-Katyusha algorithm. Further, in section B we also extend our analysis to the case where we underestimate the RSC parameter. We also provide an additional numerical result for testing the convergence rate of Rest-Katyusha with different choices of the input parameter β.

A   Proof of Theorem 3.4 and Corollary 3.5

We first state the convergence result for Katyusha algorithm for non-strongly convex functions:

Lemma A.1. *[1, Theorem 4.1] Under A.3, starting at x^0, with epoch length $m = 2n$, denote $\mathcal{D}(x^0, x^\star) := 16(F(x^0) - F^\star) + \frac{6L}{n}\|x^0 - x^\star\|_2^2$, the s-th snapshot point $\hat{x}^s$ of Katyusha algorithm satisfies:*

$$\mathbb{E}[F(\hat{x}^s)] - F^\star \leq \frac{\mathcal{D}(x^0, x^\star)}{(s+4)^2}. \tag{1}$$

Now based on the inequality of effective RSC by Lemma 3.3 in the main text we are able to provide the proof of our main result.

Proof. At each iteration, the algorithm chooses an index i uniformly at random to perform the calculation of one stochastic variance-reduced gradient. The update sequences y_{k+1} and z_{k+1} within t-th outer-loop of Rest-Katyusha depend on the realization of the following random variable which we denote as ξ_k^t:

$$\xi_t = \{i_k^t, i_{k-1}^t, ..., i_1^t, i_0^t, i_k^{t-1}, ..., i_0^{t-1}, ..., i_k^0, ..., i_0^0\}, \tag{2}$$

and for the randomness within a single outer-loop of Rest-Katyusha we specifically denote $\xi_t \backslash \xi_{t-1}$ as

$$\xi_t \backslash \xi_{t-1} = \{i_k^t, i_{k-1}^t, ..., i_1^t, i_0^t\}. \tag{3}$$

According to Lemma A.1, setting $m = 2n$, for the first stage $t = 0$:

$$\mathbb{E}_{\xi_0}[F(x^1)] - F^\star \leq \epsilon_1 := \frac{4}{n(S_0+4)^2}\left[4n\left(F(x^0) - F^\star\right) + \frac{3L}{2}\|x^0 - x^\star\|_2^2\right].$$

Then, applying Markov's inequality, with probability at least $1 - \frac{\rho}{2}$ we have:

$$F(x^1) - F^\star \leq \frac{2}{\rho}\epsilon_1. \tag{4}$$

Then we define three sequences ϵ_t, ρ_t and v_t: $\epsilon_{t+1} = \frac{1}{\beta^2}\epsilon_t$, $\rho_{t+1} = \frac{1}{\beta}\rho_t$ (with $\rho_1 := \rho$), $v_t = \frac{2\epsilon_t}{\lambda\rho_t} + \varepsilon$. Next we use and induction argument to upper bound $\mathbb{E}_{\xi_{t-1}}F(x^t) - F^\star$.

Induction step 1: We turn to the first iteration of the second stage, note that due to the effective RSC, we can write:

$$\|x - x^\star\|_2^2 \leq \frac{1}{\mu_c}\left[F(x) - F^\star + 2\tau(1+c)^2 v^2\right], \tag{5}$$

hence we can have the following:

$$\begin{aligned}
\mathbb{E}_{\xi_1 \setminus \xi_0}[F(x^2) - F^\star] &\leq \frac{16}{(S+4)^2}[F(x^1) - F^\star] + \frac{6L}{n\mu_c(S+4)^2}\left[F(x^1) - F^\star + 2\tau(1+c)^2 v_1^2\right] \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}[F(x^1) - F^\star] + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}v_1^2,
\end{aligned}$$

and then we take expectation over ξ_0 we have:

$$\begin{aligned}
\mathbb{E}_{\xi_1}[F(x^2) - F^\star] &\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\mathbb{E}_{\xi_0}[F(x^1) - F^\star] + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}v_1^2 \\
&= \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\epsilon_1 + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}\left(\frac{2\epsilon_1}{\rho\lambda} + \varepsilon\right)^2 \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\epsilon_1 + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}\left(\frac{2\epsilon_1}{\rho\lambda} + \epsilon_1\right)^2.
\end{aligned}$$

then we set:

$$\frac{12L\tau(1+c)^2}{n\mu_c}\left[\left(\frac{2}{\rho\lambda} + 1\right)\epsilon_1\right]^2 \leq \left(16 + \frac{6L}{n\mu_c}\right)\epsilon_1, \tag{6}$$

equivalently:

$$\left(\frac{2}{\rho\lambda} + 1\right)^2 \epsilon_1 \leq \frac{8n\mu_c + 3L}{6L\tau(1+c)^2}, \tag{7}$$

and denote $\mathcal{D}(x^0, x^\star) := 16[F(x^0) - F^\star] + \frac{6L}{n}\|x^0 - x^\star\|_2^2$, we have:

$$\epsilon_1 := \frac{\mathcal{D}(x^0, x^\star)}{(S_0 + 4)^2} \leq \frac{8n\mu_c + 3L}{6L\tau(1+c)^2(\frac{2}{\rho\lambda} + 1)^2}. \tag{8}$$

Hence in order to satisfy inequality (6), it is enough to set:

$$S_0 \geq \left\lceil \left(1 + \frac{2}{\rho\lambda}\right)\sqrt{\frac{6L\tau(1+c)^2\mathcal{D}(x^0, x^\star)}{8n\mu_c + 3L}} \right\rceil. \tag{9}$$

By this choice of S_0, according to inequality (6) we can write:

$$\mathbb{E}_{\xi_1}[F(x^2) - F^\star] \leq \frac{32 + \frac{12L}{n\mu_c}}{(S+4)^2}\epsilon_1, \tag{10}$$

to get $\mathbb{E}_{\xi_1}[F(x^2) - F^\star] \leq \frac{1}{\beta^2}\epsilon_1 = \epsilon_2$, it is enough to set:

$$S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil. \tag{11}$$

Induction step 2: For the $(t+1)$-th iteration, according to the induction hypothesis, we have $\mathbb{E}_{\xi_{t-1}} F(x^t) - F^\star \leq \frac{\epsilon_{t-1}}{\beta^2} = \epsilon_t$, and hence with probability $1 - \frac{\rho_t}{2}$ we have:

$$
\begin{aligned}
\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] &\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2} \mathbb{E}_{\xi_{t-1}}[F(x^t) - F^\star] + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2} v_t^2 \\
&= \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2} \epsilon_t + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2} \left(\frac{2\epsilon_t}{\rho_t\lambda} + \varepsilon \right)^2 \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2} \epsilon_t + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2} \left(\frac{2\epsilon_t}{\rho_t\lambda} + \epsilon_t \right)^2.
\end{aligned}
$$

then we set:

$$
\frac{12L\tau(1+c)^2}{n\mu_c} \left[\left(\frac{2}{\rho_t\lambda} + 1 \right) \epsilon_t \right]^2 \leq \left(16 + \frac{6L}{n\mu_c} \right) \epsilon_t, \tag{12}
$$

equivalently:

$$
\left(\frac{2}{\rho_t\lambda} + 1 \right)^2 \epsilon_t \leq \frac{8n\mu_c + 3L}{6L\tau(1+c)^2}, \tag{13}
$$

Now because $\rho_t = \frac{1}{\beta}\rho_{t-1}$, $\epsilon_t = \frac{1}{\beta^2}\epsilon_{t-1}$, we have:

$$
\left(\frac{2}{\rho_t\lambda} + 1 \right)^2 \epsilon_t = \left(\frac{2}{\rho_{t-1}\lambda} + \frac{1}{\beta} \right)^2 \epsilon_{t-1} \leq \left(\frac{2}{\rho_{t-1}\lambda} + 1 \right)^2 \epsilon_{t-1} \leq \dots \leq \left(\frac{2}{\rho\lambda} + 1 \right)^2 \epsilon_1. \tag{14}
$$

Hence by the same choice of S_0 given by (9), inequality (12) holds and consequently we can have:

$$
\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{32 + \frac{12L}{n\mu_c}}{(S+4)^2} \epsilon_t, \tag{15}
$$

to get $\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{1}{\beta^2}\epsilon_t = \epsilon_{t+1}$, it is enough to set:

$$
S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil. \tag{16}
$$

Hence we finish the induction – by the choice of:

$$
S_0 \geq \left\lceil \left(1 + \frac{2}{\rho\lambda} \right) \sqrt{\frac{6L\tau(1+c)^2 \mathcal{D}(x^0, x^\star)}{8n\mu_c + 3L}} \right\rceil, \quad S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil, \tag{17}
$$

then we will have:

$$
\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{\epsilon_t}{\beta^2} \tag{18}
$$

where $\epsilon_{t+1} = \frac{1}{\beta^2}\epsilon_t$ and $\epsilon_1 = \frac{\mathcal{D}(x^0, x^\star)}{(S_0+4)^2} = \frac{4}{n(S_0+4)^2} \left[4n\left(F(x^0) - F^\star \right) + \frac{3L}{2}\|x^0 - x^\star\|_2^2 \right]$, with probability $1 - \sum_{i=1}^{t} \frac{\rho_i}{2} \geq 1 - \frac{\rho}{2}\frac{\beta}{\beta-1} \geq 1 - \rho$ (since $\beta \geq 2$). Now we have finished the proof of Theorem 3.4.

Proof of Corollary 3.5. Finally we make a summary of this result for the proof of Corollary 3.5. First we write the number of snapshot point calculation we need to achieve $\mathbb{E}_{\xi_{t-1}} F(x^t) - F^\star \leq \delta$ at the second stage:

$$
N_s = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil \log_{\beta^2} \frac{F(x^1) - F^\star}{\delta}. \tag{19}
$$

When $\frac{2n\mu_c}{L} \leq \frac{3}{4}$, $N_s = O\left(\sqrt{\frac{L}{2n\mu_c}} \log \frac{F(x^1)-F^\star}{\delta} \right)$; when $\frac{2n\mu_c}{L} \geq \frac{3}{4}$, $N_s = O\left(\log \frac{F(x^1)-F^\star}{\delta} \right)$. Hence it is enough to run $O\left((1 + \sqrt{\frac{L}{2n\mu_c}}) \log \frac{F(x^1)-F^\star}{\delta} \right) \geq O\left(\max(1, \sqrt{\frac{L}{2n\mu_c}}) \log \frac{F(x^1)-F^\star}{\delta} \right)$

3

epochs. Since we set the epoch length $m = 2n$ and hence the number of stochastic gradient $\nabla f_i(.)$ calculation is of $O(n)$. Therefore with some more straightforward calculation we conclude that the complexity of the Rest-Katyusha algorithm is:

$$N \geq O\left(n + \sqrt{\frac{nL}{\mu_c}}\right) \log \frac{\frac{1}{\rho(S_0+4)^2}\left[16(F(x^0) - F^\star) + \frac{6L}{n}\|x^0 - x^\star\|_2^2\right]}{\delta} + O(n)S_0. \quad (20)$$

$\square$

B   Rest-Katyusha with an underestimation of μ_c

We have already established the convergence result for Rest-Katyusha algorithm when it is restarted at a frequency $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil$, but in practice the effective RSC parameter μ_c is usually unknown and difficult to estimate accurately. We need to find some practical approaches to estimate μ_c and determine whether to restart or not on the fly. To lay down the basics, we now warm up with the analysis for Rest-Katyusha when only an underestimation of μ_c is given, to see how the convergence rate of the algorithm will change.

Algorithm 1 Rest-Katyusha with a rough RSC estimate $(x^0, \mu_0, \beta, S_0, T, L)$

Initialize: $m = 2n$, $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$;

x^1 = Katyusha (x^0, m, S_0, L)
for $t = 1, ..., T$ **do**
  x^{t+1} = Katyusha (x^t, m, S, L)
end for

We present the rough RSC estimate version of Rest-Katyusha. The only difference is that the restart period has changed from $\left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil$ to $\left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$, where μ_0 is an rough (under-)estimate of the effective RSC constant μ_c and $\beta \geq 2$ is a constant which controls the robustness of possible overestimation. With this restart period, we are able to establish accelerated linear convergence result in the regime where $0 < \mu_0 < \frac{\beta^2}{4}\mu_c$. In other words, with this restart period, as long as μ_c is no more than $\beta^2/4$ times overestimated by μ_0, the Rest-Katyusha is guaranteed to achieve accelerated linear convergence w.r.t. μ_0.

Theorem B.1. *Under A.1 - 4, denote* $\varepsilon := 2\Phi(\mathcal{M})\|x^\dagger - x^\star\|_2 + 4g(x^\dagger_{\mathcal{M}^\perp})$, $\mathcal{D}(x^0, x^\star) := 16(F(x^0) - F^\star) + \frac{6L}{n}\|x^0 - x^\star\|_2^2$, $\mu_c = \frac{\gamma}{2} - 8\tau(1+c)^2\Phi^2(\mathcal{M})$, *and* $0 < \mu_0 < \frac{\beta^2}{4}\mu_c$, *with* $\beta \geq 2$, *if we run Rest-Katyusha with* $S_0 \geq \left\lceil \left(1 + \frac{2}{\rho\lambda}\right)\sqrt{2\tau(1+c)^2\mathcal{D}(x^0, x^\star)} \right\rceil$, $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$, *then the following inequality holds:*

$$\mathbb{E}[F(x^{T+1}) - F^\star] \leq \max\left\{\varepsilon, \left(\frac{\mu_0}{\mu_c\beta^2}\right)^T \frac{\mathcal{D}(x^0, x^\star)}{(S_0 + 4)^2}\right\}, \quad (21)$$

with probability at least $1 - \rho$.

Corollary B.2. *Under the same assumptions, parameter choices and notations as Theorem B.1, the total number of stochastic gradient evaluation required by Rest-Katyusha to get an δ-accuracy is:*

$$O\left(n + \sqrt{\frac{nL}{\mu_0}}\right) \log_{\frac{\beta^2\mu_c}{\mu_0}} \frac{1}{\delta} + O(n)S_0, \quad (22)$$

Proof. At each iteration, the algorithm chooses an index i uniformly at random to perform the calculation of one stochastic variance-reduced gradient. The update sequences y_{k+1} and z_{k+1} within t-th outer-loop of Rest-Katyusha depend on the realization of the following random variable which we denote as ξ_k^t:

$$\xi_t = \{i_k^t, i_{k-1}^t, ..., i_1^t, i_0^t, i_k^{t-1}, ..., i_0^{t-1}, ..., i_k^0, ..., i_0^0\}, \quad (23)$$

and for the randomness within a single outer-loop of Rest-Katyusha we specifically denote $\xi_t \backslash \xi_{t-1}$ as

$$\xi_t \backslash \xi_{t-1} = \{i_k^t, i_{k-1}^t, ..., i_1^t, i_0^t\} \tag{24}$$

According to Lemma A.1, setting $m = 2n$, for first stage $t = 0$:

$$\mathbb{E}_{\xi_0}[F(x^1)] - F^\star \leq \epsilon_1 := \frac{4}{n(S_0 + 4)^2} \left[4n \left(F(x^0) - F^\star \right) + \frac{3L}{2} \|x^0 - x^\star\|_2^2 \right].$$

Then with probability at least $1 - \frac{\rho}{2}$ we have:

$$F(x^1) - F^\star \leq \frac{2}{\rho} \epsilon_1. \tag{25}$$

Then we denote $\alpha = \frac{\mu_0}{\mu_c}$ and also define three sequences ϵ_t, ρ_t and v_t: $\epsilon_{t+1} = \frac{\alpha}{\beta^2} \epsilon_t$, $\rho_{t+1} = \frac{\sqrt{\alpha}}{\beta} \rho_t$ (with $\rho_1 := \rho$), $v_t = \frac{2\epsilon_t}{\lambda \rho_t} + \varepsilon$. Next we use and induction argument to upper bound $\mathbb{E}_{\xi_{t-1}} F(x^t) - F^\star$.

Induction step 1: We turn to the first iteration of the second stage, note that due to the effective RSC, we can write:

$$\|x - x^\star\|_2^2 \leq \frac{1}{\mu_c} \left[F(x) - F^\star + 2\tau(1 + c)^2 v^2 \right], \tag{26}$$

hence we can have the following:

$$
\begin{aligned}
\mathbb{E}_{\xi_1 \backslash \xi_0}[F(x^2) - F^\star] &\leq \frac{16}{(S + 4)^2}[F(x^1) - F^\star] + \frac{6L}{n\mu_c(S + 4)^2} \left[F(x^1) - F^\star + 2\tau(1 + c)^2 v_1^2 \right] \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S + 4)^2}[F(x^1) - F^\star] + \frac{12L\tau(1 + c)^2}{n\mu_c(S + 4)^2} v_1^2,
\end{aligned}
$$

and then we take expectation over ξ_0 we have:

$$
\begin{aligned}
\mathbb{E}_{\xi_1}[F(x^2) - F^\star] &\leq \frac{16 + \frac{6L}{n\mu_c}}{(S + 4)^2} \mathbb{E}_{\xi_0}[F(x^1) - F^\star] + \frac{12L\tau(1 + c)^2}{n\mu_c(S + 4)^2} v_1^2 \\
&= \frac{16 + \frac{6L}{n\mu_c}}{(S + 4)^2} \epsilon_1 + \frac{12L\tau(1 + c)^2}{n\mu_c(S + 4)^2} \left(\frac{2\epsilon_1}{\rho\lambda} + \varepsilon \right)^2 \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S + 4)^2} \epsilon_1 + \frac{12L\tau(1 + c)^2}{n\mu_c(S + 4)^2} \left(\frac{2\epsilon_1}{\rho\lambda} + \epsilon_1 \right)^2.
\end{aligned}
$$

then we set:

$$\frac{12L\tau(1 + c)^2}{n\mu_c} \left[\left(\frac{2}{\rho\lambda} + 1 \right) \epsilon_1 \right]^2 \leq \left(16 + \frac{6L}{n\mu_c} \right) \epsilon_1, \tag{27}$$

equivalently:

$$\left(\frac{2}{\rho\lambda} + 1 \right)^2 \epsilon_1 \leq \frac{8n\mu_c + 3L}{6L\tau(1 + c)^2}, \tag{28}$$

and denote $\mathcal{D}(x^0, x^\star) := 16[F(x^0) - F^\star] + \frac{6L}{n}\|x^0 - x^\star\|_2^2$, we have:

$$\epsilon_1 := \frac{\mathcal{D}(x^0, x^\star)}{(S_0 + 4)^2} \leq \frac{8n\mu_c + 3L}{6L\tau(1 + c)^2(\frac{2}{\rho\lambda} + 1)^2}. \tag{29}$$

Hence in order to satisfy inequality (27), it is enough to set:

$$S_0 \geq \left\lceil \left(1 + \frac{2}{\rho\lambda} \right) \sqrt{2\tau(1 + c)^2 \mathcal{D}(x^0, x^\star)} \right\rceil \geq \left\lceil \left(1 + \frac{2}{\rho\lambda} \right) \sqrt{\frac{6L\tau(1 + c)^2 \mathcal{D}(x^0, x^\star)}{8n\mu_c + 3L}} \right\rceil. \tag{30}$$

By this choice of S_0, according to inequality (27) we can write:

$$\mathbb{E}_{\xi_1}[F(x^2) - F^\star] \leq \frac{32 + \frac{12L}{n\mu_c}}{(S + 4)^2} \epsilon_1, \tag{31}$$

to get $\mathbb{E}_{\xi_1}[F(x^2) - F^\star] \leq \frac{\alpha}{\beta^2}\epsilon_1 = \epsilon_2$, it is enough to set:

$$S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil. \tag{32}$$

Induction step 2: For the $(t+1)$-th iteration, according to the induction hypothesis, we have $\mathbb{E}_{\xi_{t-1}}F(x^t) - F^\star \leq \frac{\alpha\epsilon_{t-1}}{\beta^2} = \epsilon_t$, and hence with probability $1 - \frac{\rho_t}{2}$ we have:

$$
\begin{aligned}
\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] &\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\mathbb{E}_{\xi_{t-1}}[F(x^t) - F^\star] + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}v_t^2 \\
&= \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\epsilon_t + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}\left(\frac{2\epsilon_t}{\rho_t\lambda} + \varepsilon\right)^2 \\
&\leq \frac{16 + \frac{6L}{n\mu_c}}{(S+4)^2}\epsilon_t + \frac{12L\tau(1+c)^2}{n\mu_c(S+4)^2}\left(\frac{2\epsilon_t}{\rho_t\lambda} + \epsilon_t\right)^2.
\end{aligned}
$$

then we set:

$$\frac{12L\tau(1+c)^2}{n\mu_c}\left[\left(\frac{2}{\rho_t\lambda} + 1\right)\epsilon_t\right]^2 \leq \left(16 + \frac{6L}{n\mu_c}\right)\epsilon_t, \tag{33}$$

equivalently:

$$\left(\frac{2}{\rho_t\lambda} + 1\right)^2\epsilon_t \leq \frac{8n\mu_c + 3L}{6L\tau(1+c)^2}, \tag{34}$$

Now because $\rho_t = \frac{\sqrt{\alpha}}{\beta}\rho_{t-1}$, $\epsilon_t = \frac{\alpha}{\beta^2}\epsilon_{t-1}$, we have:

$$\left(\frac{2}{\rho_t\lambda} + 1\right)^2\epsilon_t = \left(\frac{2}{\rho_{t-1}\lambda} + \frac{\sqrt{\alpha}}{\beta}\right)^2\epsilon_{t-1} \leq \left(\frac{2}{\rho_{t-1}\lambda} + 1\right)^2\epsilon_{t-1} \leq \dots \leq \left(\frac{2}{\rho\lambda} + 1\right)^2\epsilon_1. \tag{35}$$

Hence by the same choice of S_0 given by (30), inequality (33) holds and consequently we can have:

$$\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{32 + \frac{12L}{n\mu_c}}{(S+4)^2}\epsilon_t, \tag{36}$$

to get $\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{\alpha}{\beta^2}\epsilon_t = \epsilon_{t+1}$, it is enough to set:

$$S = \left\lceil 2\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil. \tag{37}$$

Hence we finish the induction – by the choice of:

$$S_0 \geq \left\lceil \left(1 + \frac{2}{\rho\lambda}\right)\sqrt{2\tau(1+c)^2\mathcal{D}(x^0, x^\star)} \right\rceil, \quad S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil, \tag{38}$$

then we will have:

$$\mathbb{E}_{\xi_t}[F(x^{t+1}) - F^\star] \leq \frac{\alpha\epsilon_t}{\beta^2} \tag{39}$$

where $\epsilon_{t+1} = \frac{\alpha}{\beta^2}\epsilon_t$ and $\epsilon_1 = \frac{\mathcal{D}(x^0, x^\star)}{(S_0+4)^2} = \frac{4}{n(S_0+4)^2}\left[4n\left(F(x^0) - F^\star\right) + \frac{3L}{2}\|x^0 - x^\star\|_2^2\right]$, with probability $1 - \frac{1}{2}\sum_{i=1}^t \rho_i \geq 1 - \frac{\rho}{2}\frac{\beta}{\beta - \sqrt{\alpha}} \geq 1 - \rho$. Now we have finished the proof of Theorem B.1.

Proof of Corollary B.2. Finally we make a summary of this result for the proof of Corollary B.2. First we write the number of snapshot point calculation we need to achieve $\mathbb{E}_{\xi_{t-1}}F(x^t) - F^\star \leq \delta$ at the second stage:

$$N_s = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil \log_{\frac{\beta^2}{\alpha}} \frac{F(x^1) - F^\star}{\delta}. \tag{40}$$

When $\frac{2n\mu_c}{L} \leq \frac{3}{4}$, $N_s = O\left(\sqrt{\frac{L}{2n\mu_c}} \log \frac{F(x^1)-F^\star}{\delta}\right)$; when $\frac{2n\mu_c}{L} \geq \frac{3}{4}$, $N_s = O\left(\log \frac{F(x^1)-F^\star}{\delta}\right)$.

Hence it is enough to run $O\left((1+\sqrt{\frac{L}{2n\mu_c}}) \log \frac{F(x^1)-F^\star}{\delta}\right) \geq O\left(\max(1, \sqrt{\frac{L}{2n\mu_c}}) \log \frac{F(x^1)-F^\star}{\delta}\right)$
epochs. Since we set the epoch length $m = 2n$ and hence the number of stochastic gradient $\nabla f_i(.)$ calculation is of $O(n)$. Therefore with some more trivial calculation we conclude that the complexity of the Rest-Katyusha algorithm is:
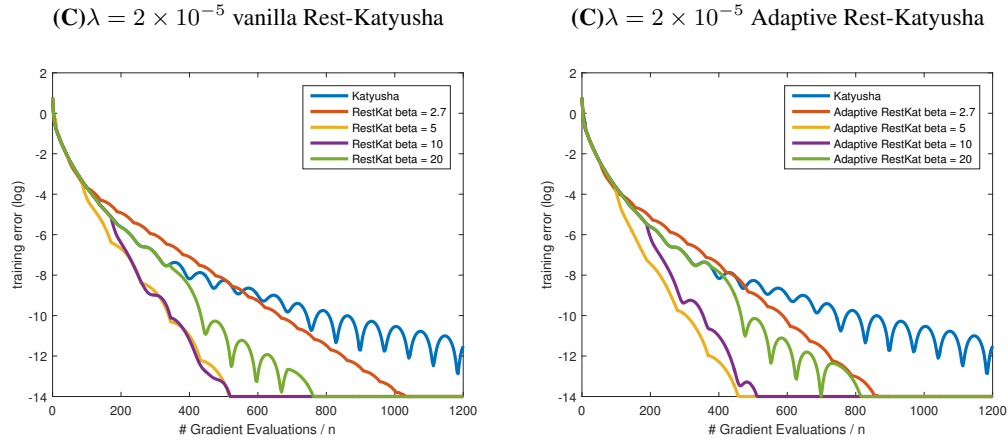
$$N \geq O\left(n + \sqrt{\frac{nL}{\mu_0}}\right) \log_{\frac{\beta^2 \mu_c}{\mu_0}} \frac{\frac{1}{\rho(S_0+4)^2}\left[16(F(x^0)-F^\star) + \frac{6L}{n}\|x^0 - x^\star\|_2^2\right]}{\delta} + O(n)S_0. \quad (41)$$

$\square$

C   Numerical test for different choices of β

In this section we provide additional experimental result on different choices of β. We choose to use the REGED dataset in this experiment as a example.

Figure 1: Comparison of different choices of β

(C)$\lambda = 2 \times 10^{-5}$ vanilla Rest-Katyusha   **(C)**$\lambda = 2 \times 10^{-5}$ Adaptive Rest-Katyusha

We test the Rest-Katsuha and Adaptive Rest-Katyusha on regularization level $\lambda = 2 \times 10^{-5}$ with 4 different choices of β including the theoretically optimal choice which is approximately 2.7. However we found out that the choice of β which provides the best practical performance is often slightly larger in experiments for real datasets. For this specific example, we can see that the best choice for β is 5 or 10 for both Rest-Katyusha and Adaptive Rest-Katyusha.

References

[1] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *arXiv preprint arXiv:1603.05953*, 2016.