Master Thesis 2017

Projection-based Regularized Dual Averaging for Stochastic Optimization with Applications to Classification and Regression

Asahi Ushio

(Student ID: 81615354)

Supervisor: Associate Professor Masahiro Yukawa

January 2018

School of Integrated Design Engineering,
Faculty of Science and Technology,
Keio University

ABSTRACT

In this paper, a framework for regularized stochastic optimization based on the regularized dual averaging (RDA) is presented. Our approach differs from the previous studies of RDA in three aspects. First, the squared-distance function to a closed convex set is employed as a part of the objective functions for stable learning. In the particular application of online regression, the squared-distance function is reduced to a normalized version of the typical squared-error (least square) function. Second, since the squared-distance function is second-differentiable, the step size can be constant. The original RDA framework, however, has undesirable increase of regularizer with a constant step size. Our approach are modified to be the regularization effect stable. Third, a sparsity-promoting metric is employed, originated from the proportionate-type adaptive filtering algorithms, and propose a weighted ℓ_1 regularization, which can enhance sparsity efficiently under a sparsity-promoting metric.

The three differences yield a better sparsity-seeking capability, leading to improved convergence properties. Extensive experiments such as classification and regression problem show the advantages of the proposed algorithm over the existing methods including AdaGrad and adaptive proximal forward-backward splitting (APFBS).

Acknowledgment

This thesis completes my studies in Yukawa Laboratory Keio Universities since 2015. I am deeply grateful to my adviser Associate Professor Masahiro Yukawa for his guidance, advices, and comments through all these years. It is my pressure to deeply thank all students of Yukawa Laboratory. This work was partially supported by JSPS Grants-in-Aid (15K06081, 15K13986, 15H02757).

Contents

Chapter 1

Introduction

Stochastic optimization (stochastic approximation [1] more in general) has drawn growing attention over the past years due particularly to the recent data deluge [2]. There are various kinds of stochastic optimization such as quasi-Newton method, which exploits an approximate the Hessian matrix, such as finite difference method based algorithms (SGD-QN [3], AdaDelta [4], and variance-based SGD [5, 6]), extended Gauss-Newton based algorithms [7–9], and stochastic LBFGS (Broyden Fletcher Goldfarb Shanno) [10–12]. Besides Hessian approximation, natural gradient methods [13–15] approximate the Fisher information matrix, and AdaGrad [16], RMSprop [17], and Adam [18] employ the root-mean-square (RMS) of the previous gradients. These methods can be seen as changing the geometry sequentially to improve the convergence, which is generally recognized as the variable-metric method [22, 23]. One can see that recently proposed algorithms can be characterized by the metric to be employed such as the Hessian [3–12], the Fisher information matrix [13–15], and RMS of the previous gradients [16–18].

We focus on the case where the solution to be estimated by stochastic optimization, is "sparse"; i.e., many components are zero. This often happens in a wide range of applications such as echo cancellation, channel estimation, text classification, etc. Sparseness has been exploited in adaptive filtering [19–21] which is closely related to stochastic optimization. The algorithms in [19–21] can also be regarded as variable-metric methods [22, 23]. More recently, sparsity-aware algorithms have been studied for stochastic optimization and online learning, including the adaptive proximal forward-backward splitting (APFBS) method [24, 25], the forward looking subgradients and forward backward splitting method (FOBOS) [26], and the regularized dual averaging (RDA) method [27]. In particular, the idea of RDA comes originally from the primal-dual subgradient methods [28] of Nesterov, and it is known to yield a sparser solution than the FOBOS method [27]. An approach similar to RDA is known as the follow-the-regularized-leader in online convex optimization [29]. Although RDA yields a sparse solution for some loss functions, the regularization parameter increases as time goes by when the loss function is smooth, resulting in an undesirably sparse solution (see Section **??** for more in detail). We address this issue and presents stochastic regularized optimization framework, in which we achieve fixed amount of regularization regardless of a step size.

It is widely known that the normalized least mean square (NLMS) algorithm [30, 31] often performs better and is more stable than the classical stochastic gradient descent (SGD) method referred to as the least mean square (LMS) algorithm [32]. The NLMS algorithm is usually derived based the so-called minimum disturbance principle [33], and operates iterative projections onto zero-instantaneous-error hyperplanes. In the present study, we highlight the fact that NLMS can be regarded as a SGD method for "normalized" squared errors, or equivalently the squared distances to the zero-instantaneous-error hyperplanes. The squared distance functions have actually been considered in the studies of the adaptive projected subgradient method (APSM) [34–36], which also handles ordinary distances, and APFBS.

In this paper, we present a stochastic regularized optimization algorithm named *projection-based regularized dual averaging (PDA)*. We consider the squared-metric-distance to the random closed convex set, where the randomness comes from the measurements. To be precise, we consider a specific stochastic optimization problem of minimizing the expectation of the squared distance function penalized by some convex regularizer. Here, the distance is defined with a time-dependent metric, which is denoted by $\boldsymbol{Q}_t$, that is designed to promote sparsity of our estimates. As a result, the estimation is updated with the $\boldsymbol{Q}_t$-gradient of the squared-distance function. We mention that metric $\boldsymbol{Q}_t$ causes undesirable biases when a usual sparsity promoting regularizer is adopted such as the unweighted ℓ_1-norm. (see Section 3.3 for more in detail). To offset the undesirable biases due to the metric $\boldsymbol{Q}_t$, PDA update involves weighted regularization by the squared diagonal elements of $\boldsymbol{Q}_t$.

PDA is also designed to attain a fixed amount of regularization with a constant step size, while the ordinary RDA keeps increasing the amount of regularization. The key ingredients of the proposed algorithm are summarized as (i) normalization of the input vector, (ii) variable-metric, and (iii) a fixed regularization with a constant step size.

This makes three practical advantages in stochastic optimization involving sparse structures. First, the use of the squared-distance function avoids such a situation that the gradient vector becomes undesirably large for large inputs, stabilizing the algorithm. Second, the use of $\boldsymbol{Q}_t$-metric and weighted ℓ_1 regularizer guides the update direction towards the true (sparse) solution. Assembling them together, the proposed algorithm enjoys a notable sparsity-seeking property. Third, making the regularization parameter constant prevents our estimates from being too sparse. Extensive simulations, which include regression and classification problems with several real data show the advantages of the proposed algorithm.

Chapter 2

Preliminaries

We denote by $\mathbb{R}^{a \times b}$ the set of real $a \times b$ matrices, by $\mathbb{N}$ the set of all nonnegative integers, and $\mathbb{N}^* := \mathbb{N} \setminus \{0\}$. We denote by $\langle \boldsymbol{w}, \boldsymbol{z} \rangle_{\boldsymbol{Q}} := \boldsymbol{w}^\mathsf{T} \boldsymbol{Q} \boldsymbol{z}$ the inner product for $\boldsymbol{w}, \boldsymbol{z} \in \mathbb{R}^n$ with a positive definite matrix $\boldsymbol{Q}$ and $\|\boldsymbol{w}\|_{\boldsymbol{Q}} := \sqrt{\langle \boldsymbol{w}, \boldsymbol{w} \rangle_{\boldsymbol{Q}}}$ the $\boldsymbol{Q}$-norm induced by $\langle \boldsymbol{w}, \boldsymbol{z} \rangle_{\boldsymbol{Q}}$. Also, we denote by $\boldsymbol{w}^\mathsf{T}$ the transpose of a vector $\boldsymbol{w} := [w_1, w_2, \cdots, w_n]^\mathsf{T} \in \mathbb{R}^n$.

2.1 Problem Formulation

We consider the following stochastic regularized optimization problem:

$$\min_{\boldsymbol{w} \in \mathbb{R}^n} \mathbb{E}_z \left[\varphi(\boldsymbol{w}, z) \right] + \psi(\boldsymbol{w}), \tag{2.1}$$

where $\mathbb{E}_z$ stands for expectation of $z := (\boldsymbol{x}, y)$, which is an input-output pair from an unknown underlying distribution, φ is a possibly nonsmooth loss function , and ψ is a possibly nonsmooth regularizer. We denote $z_\tau := (\boldsymbol{x}_\tau, y_\tau)$ an observation of z at the time index $\tau = 1, 2, \cdots, t$. We consider the following problem to minimize the empirical loss:

$$\min_{\boldsymbol{w} \in \mathbb{R}^n} \frac{1}{t} \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w}) \right] + \psi(\boldsymbol{w}), \tag{2.2}$$

where $\varphi_\tau(\boldsymbol{w}) = \varphi(\boldsymbol{w}, z_\tau)$.

2.2 Regularized Dual Averaging

To solve (2.1) in the case of $\psi = 0$, (i.e., the case of unstochastic regularized optimization problems) Nesterov has proposed the dual averaging method in [28], which aims to minimize

$$l_t(\boldsymbol{w}) := \frac{1}{t} \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w}_\tau) + \langle \boldsymbol{g}_\tau, \boldsymbol{w} - \boldsymbol{w}_\tau \rangle_{\boldsymbol{I}} \right]. \tag{2.3}$$

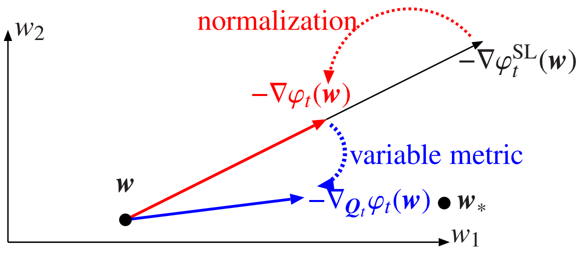

w_2
normalization
$-\nabla\varphi_t^{\mathrm{SL}}(\boldsymbol{w})$
$-\nabla\varphi_t(\boldsymbol{w})$
variable metric
$\boldsymbol{w}$
$-\nabla_{\boldsymbol{Q}_t}\varphi_t(\boldsymbol{w})$
$\boldsymbol{w}_*$
w_1

Figure 2.1: The anti-gradients for a large input vector.

The lower linear model (2.3) is an average of affine minorants of $\varphi_\tau(\boldsymbol{w})$. The simple dual averaging update is given by

$$\begin{aligned}\boldsymbol{w}_t :&= \mathop{\arg\min}_{\boldsymbol{w}\in\mathbb{R}^n}\left(l_t(\boldsymbol{w}) + \frac{\beta_t}{t}h(\boldsymbol{w})\right)\\ &= \mathop{\arg\min}_{\boldsymbol{w}\in\mathbb{R}^n}\left(\left\langle\frac{\boldsymbol{s}_t}{t},\boldsymbol{w}\right\rangle_{\boldsymbol{I}} + \frac{\beta_t}{t}h(\boldsymbol{w})\right),\end{aligned} \tag{2.4}$$

where $(\beta_\tau)_{\tau\in\mathbb{N}^*}$ is a nonnegative and non-decreasing sequence, $\boldsymbol{s}_t := \sum_{\tau=1}^t \boldsymbol{g}_\tau$, and $h(\boldsymbol{w})$ is the so-called prox-function, which determines the distance. In [27], Xiao has proposed RDA, which is an extension of dual averaging to the stochastic regularized optimization with a time-invariant regularizer ψ. The update equation of RDA is defined as

$$\boldsymbol{w}_t := \mathop{\arg\min}_{\boldsymbol{w}\in\mathbb{R}^n}\left(\left\langle\frac{\boldsymbol{s}_t}{\beta_t},\boldsymbol{w}\right\rangle_{\boldsymbol{I}} + h(\boldsymbol{w}) + \frac{t}{\beta_t}\psi(\boldsymbol{w})\right). \tag{2.5}$$

2.3 Projection-based Method

The loss function $\varphi_t(\boldsymbol{w})$ is, in many cases, chosen depending on problems. For instance, squared loss is usually used for the regression problem and hinge loss or logistic loss for the classification problem. However, instead of choosing the loss function directly based on the problem, there is more sophisticated methods to define loss functions, which is called projection-based method [37].

With a time variant positive definite matrix $\boldsymbol{Q}_t$, the $\boldsymbol{Q}_t$-metric distances can be defined as $d_{\boldsymbol{Q}_t}(\boldsymbol{w}, C_t) := \min_{z\in C_t}\|\boldsymbol{w}-\boldsymbol{z}\|_{\boldsymbol{Q}_t}$ between an arbitrary point $\boldsymbol{w}\in\mathbb{R}^n$ and a closed convex set $C_t\subset\mathbb{R}^n$. Then, projection-based method use the loss function defined as

$$\varphi_t(\boldsymbol{w}) := \frac{1}{2}d_{\boldsymbol{Q}_t}^2(\boldsymbol{w}, C_t). \tag{2.6}$$

The $\boldsymbol{Q}_t$-gradient of φ_t at the previous estimate $\boldsymbol{w}_{t-1}\in\mathbb{R}^n$ is given by

$$\boldsymbol{g}_t := \nabla_{\boldsymbol{Q}_t}\varphi_t(\boldsymbol{w}_{t-1}) = \boldsymbol{w}_{t-1} - P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}), \tag{2.7}$$

where $P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}) := \arg\min_{z \in C_t} \|\boldsymbol{w} - \boldsymbol{z}\|_{\boldsymbol{Q}_t}$ is the $\boldsymbol{Q}_t$-projection onto C_t. In the case of $\psi_t = 0$, the SGD update is given by

$$\boldsymbol{w}_t := \boldsymbol{w}_{t-1} - \eta \boldsymbol{g}_t. \tag{2.8}$$

Unlike the case of ordinary loss functions where the step size is bounded depending on measurements, the step size η is simply restricted in the range of $[0, 2]$ in projection-based method. Note here that the projection operator is nonexpansive w.r.t. $\|\cdot\|_{\boldsymbol{Q}_t}$ (i.e., Lipschitz continuous with constant 1), and the gradient operator $\nabla_{\boldsymbol{Q}_t}\varphi_t$ is also nonexpansive. The gradient vector has the following property:

$$\boldsymbol{g}_t = \boldsymbol{0} \Leftrightarrow P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}) = \boldsymbol{w}_{t-1} \Leftrightarrow \boldsymbol{w}_{t-1} \in C_t. \tag{2.9}$$

The convergence of projection-based method (2.8) with $\eta \in [0, 2]$ is ensured by the property (2.9) and the nonexpansivity of $\nabla_{\boldsymbol{Q}_t}\varphi_t$.

We, then, introduce an application for the regression problem and note merits of projection-based method. Let $\boldsymbol{x}_t \in \mathbb{R}^n$ be the input vector, and $y_t := \boldsymbol{w}_*^{\mathsf{T}}\boldsymbol{x}_t + v_t \in \mathbb{R}$ is the output at time instant t with the unknown vector $\boldsymbol{w}_* \in \mathbb{R}^n$ and the additive noise $v_t \in \mathbb{R}$. For the regression problem, projection-based method uses

$$C_t := \arg\min_{\boldsymbol{w} \in \mathbb{R}^n} \left\|\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w} - \boldsymbol{y}_t\right\|_{\boldsymbol{I}}^2 \tag{2.10}$$

where $\boldsymbol{X}_t := [\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-r+1}] \in \mathbb{R}^{n \times r}$ and $\boldsymbol{y}_t := [y_t, y_{t-1} \cdots, y_{t-r+1}]^{\mathsf{T}} \in \mathbb{R}^r$ for some $r \in \mathbb{N}^*$. The $\boldsymbol{Q}_t$-gradient $\boldsymbol{g}_t$ for (2.10) can be seen as an ordinary gradient for a loss function. For instance, in the case of $r = 1$, $\boldsymbol{g}_t$ becomes the gradient of normalized squared loss

$$\varphi_t^{\mathrm{NSL}}(\boldsymbol{w}) = \frac{\left(y_t - \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t\right)^2}{2\,\|\boldsymbol{x}_t\|_{\boldsymbol{Q}_t}^2} \tag{2.11}$$

and (2.8) is reduced to the (improved) proportionate NLMS (PNLMS) algorithms [19, 20, 38]. Figure 2.1 shows the difference among the anti-gradient vectors $-\nabla\varphi_t^{\mathrm{SL}}(\boldsymbol{w})$, with $\varphi_t^{\mathrm{SL}}(\boldsymbol{w}) := \left(y_t - \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t\right)^2/2$, which is squared loss used as ordinary loss for the regression problem, $-\nabla\varphi_t^{\mathrm{NSL}}(\boldsymbol{w})$ with a fixed metric ($\boldsymbol{Q}_t = \boldsymbol{I}$), and $-\nabla_{\boldsymbol{Q}_t}\varphi_t^{\mathrm{NSL}}(\boldsymbol{w})$. The gradient of $\varphi_t^{\mathrm{SL}}(\boldsymbol{w})$ can be disturbed by large inputs, which makes the algorithm unstable. The squared-distance cost (2.6) robustifies the gradient against large inputs. In addition, the metric $\boldsymbol{Q}_t$ guides the update direction towards the optimal point $\boldsymbol{w}_*$, leading to convergence acceleration. When the (improved) proportionate NLMS (PNLMS) algorithms [19, 20, 38].

If $r \geq 2$, the algorithm (2.8) with (2.10) is reduced to the proportionate affine projection algorithm (PAPA) [39, 40]. If the metric is Euclidean, PAPA and PNLMS are further reduced to the affine projection algorithm (APA) [41, 42] and NLMS, respectively. For the classification case, the passive aggressive algorithm [43] uses a half-space where the instantaneous error is zero as C_t to derive algorithm to solve classification problems.

Chapter 3

Projection-based Regularized Dual Averaging

We present the proposed algorithm, called PDA, which is a projection-based stochastic regularized optimization framework based on the dual averaging. We start the derivation of PDA from generalizing RDA framework (2.5) in Section 3.1. Then, we point out problems of RDA (remark 1), which have not been discussed in any previous studies yet. We shows the proposed algorithm, PDA in Section 3.2.

In Section 3.3, we investigate a sparsity-promoting metric $\boldsymbol{Q}_t$ in terms of the relation to the regularizer. Applications for regression and classification problems are shown in Section 3.4, the complexity of PDA is discussed in Section 3.5, and relations to prior works (AdaGrad [16] and APFBS [25, 44]) in Section 3.6. We also give the regret analysis of PDA in Appendix A.1.

3.1 Generalized RDA

In RDA framework (2.5), $1/\beta_t$ behaves as the step size but unlike SGD, in which $\boldsymbol{g}_t$ is scaled by the step size η_t, $\boldsymbol{s}_t$ is scaled by $1/\beta_t$, that means $1/\beta_t$ is uniformly distributed to all (sub)gradients $(\boldsymbol{g}_\tau)_{\tau\in\mathbb{N}^*}$.

Remark 1. *Although β_t in* (2.5) *is not restricted to a particular case, $\sqrt{t}/\eta$ with a constant η is used for practical cases in [27]. In this case, since $\beta_t \sim O(\sqrt{t})$, the strength of the regularizer increases by $t/\beta_t \sim O(\sqrt{t})$. Here we point out the problem that even though the choice $\beta_t \sim O(\sqrt{t})$ works well, since the strength of the regularizer t/β_t increases depending on β_t, it is difficult to control the effect of regularization in general cases. For instance, if β_t is a constant, t/β_t increase by $O(t)$, this interrupts the solution to be updated.*

To see the effect of the issue mentioned in remark 1, we generalize (2.5) as

$$\boldsymbol{w}_t := \arg\min_{\boldsymbol{w}\in\mathbb{R}^n}\left(\left\langle \frac{\eta}{t^b}\boldsymbol{s}_t, \boldsymbol{w}\right\rangle_{\boldsymbol{I}} + h(\boldsymbol{w}) + \eta t^a \psi(\boldsymbol{w})\right). \tag{3.1}$$

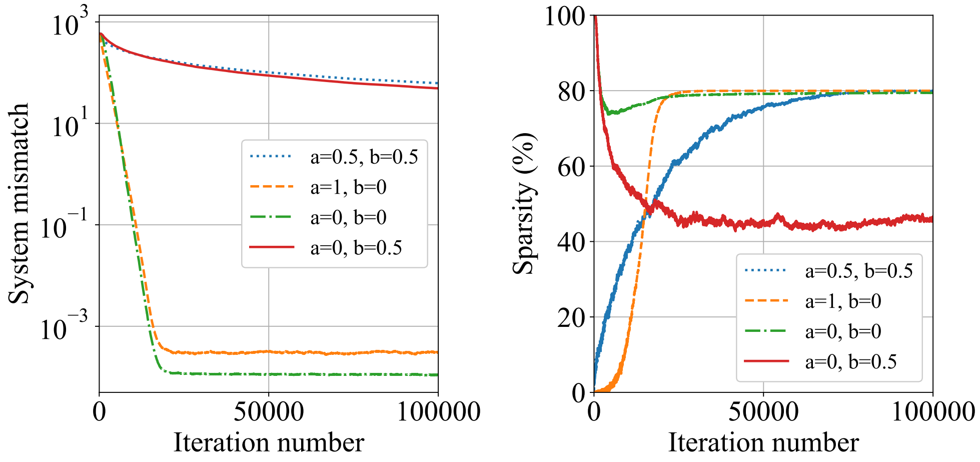


Figure 3.1: System mismatch (left) and proportion of zero components (right) for sparse system identification.

Table 3.1: Parameters for generalized RDA simulation.

	η	λ
$a = 0.5,\ b = 0.5$	10^{-4}	10^4
$a = 1,\ b = 0$	10^{-3}	10^4
$a = 0,\ b = 0$	10^{-3}	5×10^{-1}
$a = 0,\ b = 0.5$	10^{-4}	5×10^{-3}

where η, a, and b are constants. We call this framework (3.2) as generalized RDA. Note that in the case of $a = 1 - b$, generalized RDA (3.2) corresponds to original RDA framework, $a = 0.5$, $b = 0.5$ is RDA algorithm practically used in [27] ($\beta_t = \sqrt{t}/\eta$), and $a = 1$, $b = 0$ is RDA algorithm with a constant $\beta_t = 1/\eta$. Figure 3.1 shows the result for sparse system identification (online regression problem) with squared loss and ℓ_1 regularization $\lambda \sum_{i=1}^{n} w_i$ for $\lambda > 0$ as ψ, comparing four settings of generalized RDA (see Section 4.2.1 for more detail about the experiment setting). Parameters (Table 3.1) are set to achieve the best performance in terms of system mismatch at the end of iteration. All of experiments in this paper uses $h(\boldsymbol{w}) := \|\boldsymbol{w}\|^2/2$, so (3.1) can be reduced to

$$\begin{aligned}\boldsymbol{w}_t &= \arg\min_{\boldsymbol{w}\in\mathbb{R}^n}\left(\frac{1}{2}\left\|\boldsymbol{w}+\frac{\eta}{t^b}\boldsymbol{s}_t\right\|_{\boldsymbol{I}}^2 + \eta t^a\psi(\boldsymbol{w})\right)\\ &= \mathrm{prox}^{\boldsymbol{I}_t}_{\eta t^a\psi}\left(-\frac{\eta}{t^b}\boldsymbol{s}_t\right).\end{aligned} \tag{3.2}$$

In SGD, if the objective loss function φ is twice-differentiable, it is more suitable to use a constant step size, which is at least smaller than $2/\sigma_{max}$ where σ_{max} is the maximum eigenvalue of Hessian matrix of φ than arbitrary scheduling a decreasing

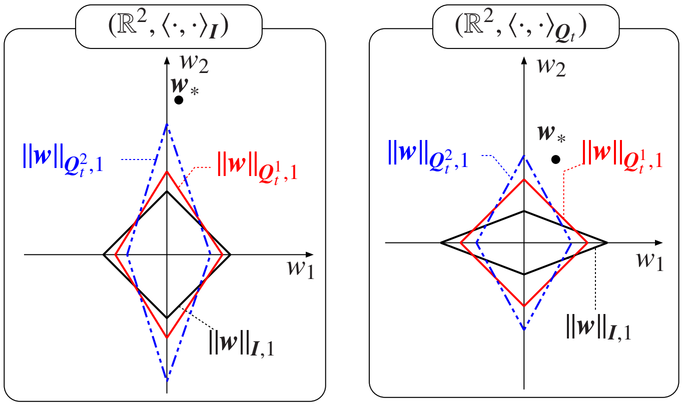

$(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{I}})$
w_2
$\boldsymbol{w}_*$
$\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$
$\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^1,1}$
w_1
$\|\boldsymbol{w}\|_{\boldsymbol{I},1}$
$(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{Q}_t})$
w_2
$\boldsymbol{w}_*$
$\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$
$\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^1,1}$
w_1
$\|\boldsymbol{w}\|_{\boldsymbol{I},1}$

Figure 3.2: Unit balls for different norms in $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{Q}_t})$ and $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{I}})$ in the case of $q_{t,1} < q_{t,2}$.

step size sequence. The result shows that, as in the case of SGD, since we use the squared loss, which is twice-differentiable, a constant β_t ($a = 1,\ b = 0$) effective in RDA that achieves much higher accuracy than an increasing β_t ($a = 0.5,\ b = 0.5$). Moreover, generalized RDA with $a = 0,\ b = 0$ improve the performance, since by letting $b = 0$, we can prevent the increase of regularization effect discussed in remark 1, which becomes non negligible effect when β_t is a constant. Based on the RDA analysis in this section, we propose new framework for the stochastic regularized optimization in next section.

3.2 PDA

We present the proposed framework, projection-based regularized dual averaging (PDA). PDA is based on RDA but there are essential differences between them. First, while original RDA [27] is considered to use ordinary loss, we employ projection-based method (2.6) as the loss function to enhance sparsity by variable metric and achieve stable learning. As far as we know, there are no algorithm, which use projection-based method in the framework of dual averaging, except PDA. Second, we know that RDA can be improved by a constant β_t in the case of twice-differentiable objective loss function from the investigation for RDA in Section 3.1. It is desired to uses a constant step size for PDA, since (2.6) is twice-differentiable. Then, with a constant $1/\eta$ as β_τ for all $\tau \in \mathbb{N}^*$, RDA (2.5) becomes

$$\boldsymbol{w}_t := \arg\min_{\boldsymbol{w}\in\mathbb{R}^n} \left(\langle \eta \boldsymbol{s}_t, \boldsymbol{w} \rangle_{\boldsymbol{Q}_t} + \frac{1}{2} \|\boldsymbol{w}\|^2_{\boldsymbol{Q}_t} + \eta t \psi(\boldsymbol{w}) \right). \tag{3.3}$$

We know that, however, by Section 3.1 RDA has undesirable increase of regularization with a constant step size, which degrades performance. As we mentioned in remark 1, the strength of the regularizer increases depending on β_t, we need to schedule β_t so as not to interrupt the update of coefficients. We thus further modify

(3.3) by removing t multiplied $\eta\psi$ in the third term (3.3) to keep constant strength of ψ regardless of any β_t. Finally, the update equation of PDA is given by

$$\begin{aligned} \boldsymbol{w}_t : &= \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\left(\langle \boldsymbol{s}_t, \boldsymbol{w}\rangle_{\boldsymbol{Q}_t} + \frac{1}{2\eta}\,\|\boldsymbol{w}\|^2_{\boldsymbol{Q}_t} + \psi_t(\boldsymbol{w})\right) \\ &= \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\left(\eta\psi_t(\boldsymbol{w}) + \frac{1}{2}\,\|\boldsymbol{w} + \eta\boldsymbol{s}_t\|^2_{\boldsymbol{Q}_t}\right) \\ &= \mathrm{prox}^{\boldsymbol{Q}_t}_{\eta\psi_t}(-\eta\boldsymbol{s}_t), \end{aligned} \tag{3.4}$$

where $\boldsymbol{g}_t$ is defined as (2.7) and $\boldsymbol{w}_t$ is initialized by arbitrary vector $\boldsymbol{w}_0$. In the generalized RDA framework (3.2), this modified RDA corresponds to the case of $a = 0,\ b = 0$. The key ideas of PDA are summarized as below:

- Projection-based method, which enable high stability of learning, with variable metric to enhance sparsity

- RDA based algorithm with constant β_t and removal of increasing term multiplied regularizer, by which the solution will not be extensively sparse.

Table 3.2 summarizes the PDA algorithm.

3.3 Weighted ℓ_1 Regularizer under Sparsity-promoting Metric

We propose weighted ℓ_1 regularizer to perform reasonable regularization under a sparsity-promoting metric. Let $\boldsymbol{Q}_t$ be a diagonal matrix $\boldsymbol{Q}_t := \mathrm{diag}(q_{t,1}, q_{t,2}, \cdots, q_{t,n})$. The metric is designed as follows [45, 46]:

$$\boldsymbol{Q}_t^{-1} := \left(\alpha\boldsymbol{I} + n\frac{1-\alpha}{S_t}\tilde{\boldsymbol{Q}}_t^{-1}\right)^{-1}, \tag{3.5}$$

where $\tilde{\boldsymbol{Q}}_t := \mathrm{diag}(|w_{t-1,1}|, |w_{t-1,2}|, \cdots, |w_{t-1,n}|) + \epsilon\boldsymbol{I}$ for some $\epsilon > 0$, $\alpha \in [0, 1]$, and $S_t := \sum_{i=1}^n (|w_{t-1,i}| + \epsilon)^{-1}$. Figure 3.2 illustrates the unit balls for three norms in the Hilbert spaces $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{I}})$ and $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{Q}_t})$: the ℓ_1 norm $\|\boldsymbol{w}\|_{\boldsymbol{I},1} := \sum_{i=1}^n |w_i|$, a weighted ℓ_1 norm $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^1,1} := \sum_{i=1}^n q_{t,i}|w_i|$, and $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1} := \sum_{i=1}^n q_{t,i}^2|w_i|$. One can see that the ℓ_1 ball has a "fat" shape in $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{Q}_t})$. This actually forces the proximity operator to shrink large components more than small components, yielding undesirable biases. To avoid it and to shrink small components, we design the regularizer as follows:

$$\psi_t(\boldsymbol{w}) = \lambda\,\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1} := \lambda\sum_{i=1}^n q_{t,i}^2|w_i|, \tag{3.6}$$

where $\lambda > 0$ is the regularization parameter. The unit ball of the norm $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$ in (3.6) has a "tall" shape in $(\mathbb{R}^2, \langle\cdot,\cdot\rangle_{\boldsymbol{Q}_t})$. The proximity operator for the ψ_t in (3.6) is

Table 3.2: PDA algorithm.

```
**Requirement**: $\lambda > 0, \eta \in [0, 2], \alpha \in [0, 1]$
                 $r \in \mathbb{N}^*, \delta > 0$
**Initialization**: Initialize $\boldsymbol{s}_0$ and $\boldsymbol{w}_0$.
**Iteration**: For $\tau = 1, 2, \cdots, t$
1. $\boldsymbol{g}_\tau := \boldsymbol{w}_{\tau-1} - P^{\boldsymbol{Q}_\tau}_{C_\tau}(\boldsymbol{w}_{\tau-1})$
2. $\boldsymbol{s}_\tau = \boldsymbol{s}_{\tau-1} + \boldsymbol{g}_\tau$
3. $\boldsymbol{w}_\tau := \mathrm{prox}^{\boldsymbol{Q}_\tau}_{\eta\psi_\tau}(-\eta \boldsymbol{s}_\tau)$
```

given by

$$\mathrm{prox}^{\boldsymbol{Q}_t}_{\eta\psi_t}(\boldsymbol{w}) = \sum_{i=1}^{n} \boldsymbol{e}_i \mathrm{sgn}(w_i) \left[|w_i| - q_{t,i}\lambda\eta\right]_+ , \tag{3.7}$$

where $\{\boldsymbol{e}_i\}_{i=1}^n$ is the standard basis of $\mathbb{R}^n$, $\mathrm{sgn}(\cdot)$ is the signum function, and $[\cdot]_+ := \max\{\cdot, 0\}$ is the hinge function.

3.4 Applications to Regression and Classification

In the case of an online regression problem, we can use (2.10) as C_t and the projection onto C_t is given by

$$P^{\boldsymbol{Q}_t}_{C_t}(\boldsymbol{w}_{t-1}) := \boldsymbol{w}_{t-1} - \boldsymbol{Q}_t^{-1}\boldsymbol{X}_t^{\dagger}(\boldsymbol{X}_t^{\top}\boldsymbol{w}_{t-1} - \boldsymbol{y}_t), \tag{3.8}$$

where $\boldsymbol{X}_t^{\dagger}$ is the Moore-Penrose pseudo-inverse. In practice, $\boldsymbol{X}_t^{\dagger}$ is replaced by $\boldsymbol{X}_t(\boldsymbol{X}_t^{\top}\boldsymbol{Q}_t^{-1}\boldsymbol{X}_t + \delta\boldsymbol{I})^{-1}$, where $\delta > 0$ is the regularization parameter for numerical stability. We also apply the PDA algorithm to an online classification problem. Let $y_t \in \{-1, 1\}$ and define

$$C_t := \left\{\boldsymbol{w} \in \mathbb{R}^n \mid y_t\boldsymbol{w}^{\top}\boldsymbol{x}_t \geq 1\right\}. \tag{3.9}$$

The projection onto C_t is given by

$$P^{\boldsymbol{Q}_t}_{C_t}(\boldsymbol{w}_{t-1}) := \boldsymbol{w}_{t-1} - \boldsymbol{Q}_t^{-1}\boldsymbol{x}_t \frac{\left[1 - y_t\boldsymbol{w}_t^{\top}\boldsymbol{x}_t\right]_+}{y_t \|\boldsymbol{x}_t\|^2_{\boldsymbol{Q}_t^{-1}}} \tag{3.10}$$

The $\boldsymbol{Q}_t$-gradient $\boldsymbol{g}_t$ for (3.9) can be seen as an ordinary gradient for

$$\varphi_t(\boldsymbol{w}) = \frac{\left(\left[y_t\boldsymbol{w}^{\top}\boldsymbol{x}_t - 1\right]_+\right)^2}{2\|\boldsymbol{x}_t\|^2_{\boldsymbol{Q}_t^{-1}}}. \tag{3.11}$$

Table 3.3: Computational complexity.

Algorithms	number of multiplication
PDA	$4n + 3nr + nr^2 + r^2$, $O(n)$ (if $r = 1$)
RDA, AdaGrad, and FOBOS	$O(n)$

3.5 Computational Complexity

The computational complexity for PDA is nr for error calculation, nr^2 for normalization of inputs, $r^2 + nr$ for updating $\boldsymbol{w}_t$, $2n$ for proximity operator, and $2n$ for metric derivation. In total, the computational complexity is $4n + 3nr + nr^2 + r^2$ multiplications. In the case of $\boldsymbol{Q}_t = \boldsymbol{I}$, the complexity of PDA can be reduced to $n + 2nr + nr^2 + r^2$. Since r is usually small integer (we use $r = 1$ for all experiments except for echo cancellation experiment, which uses $r = 2$), the complexity can be regarded as $O(n)$. The computational complexity $O(n)$ is a typical choice comparing other stochastic regularized optimizations such as AdaGrad, RDA, and FOBOS. Table 3.3 summarizes the computational complexity for PDA and other algorithms.

3.6 Relation to Prior Work

3.6.1 APFBS

One can apply the iterates $\boldsymbol{w}_t := \mathrm{prox}_{\eta\psi_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1} - \eta\boldsymbol{g}_t)$ to (2.1). This is actually a special case of APFBS [24, 25], which resembles the FOBOS algorithm [26] in the sense of using forward-backward splitting for online tasks. Note however that APFBS explicitly uses (the sum of multiple) squared-distance functions together with variable metrics, whereas FOBOS considers the squared loss φ_t^{SL} for regression. APFBS is a projection-based forward-backward splitting algorithm, while PDA is based on RDA [27] (see Appendix A.2 about the difference between forward-backward splitting and RDA).

3.6.2 AdaGrad

AdaGrad [16] is one of the celebrated online learning methods in machine learning. The idea is to reduce the variance of the (sub)gradient vector by summing up the outer-products of the history of the (sub)gradient vectors to build a metric. The AdaGrad algorithm was applied to two types of algorithms: RDA and the composite mirror descent [47, 48] (which is a generalization of FOBOS [26]). AdaGrad-RDA has some similarities to the proposed method in the sense that both methods are based on RDA and employs variable metrics. However, AdaGrad-RDA uses the ordinary squared loss $\varphi_t^{\mathrm{SL}}(\boldsymbol{w})$ as well as the metric used in AdaGrad is different from that used in PDA.

Chapter 4

Experiments

We show the efficacy of the proposed algorithm PDA in classification and regression tasks. First, we show the result on classification tasks, in which we use MNIST hand written digit dataset [49] and RCV text dataset [50]. Then, we show the result on regression tasks, which include experiments on sparse system identification problem, an acoustic echo cancellation problem, and nonlinear model estimation by multi-kernel adaptive filtering [51]. We use 10^{-5} for numerical stability parameter in all experiments.

4.1 Classification

In classification tasks, we compare the proposed algorithm with RDA [27], AdaGrad-RDA [16], AdaGrad-Fobos [16], Adam [18] , and Adadelta [4]. PDA uses (2.6) with $\boldsymbol{C}_t$ (3.9) and The other algorithms use logistic loss

$$\varphi_t(\boldsymbol{w}) = y_t \log\left(1 + e^{-\hat{y}_t}\right) + (1 - y_t) \log\left(\frac{1 + e^{-\hat{y}_t}}{e^{-\hat{y}_t}}\right) \tag{4.1}$$

and $\psi_t(\boldsymbol{w}) := \lambda \|\boldsymbol{w}\|_{\boldsymbol{I},1}$ for all algorithms. We split the dataset into validation dataset (30%) and training dataset (70%) and evaluate the algorithms by error rate, which is the misclassification ratio for the validation set. Also we show the sparsity, proportion of zero components, of the estimated coefficient. In both experiments, the error rate and sparsity is averaged over 300 independent trials. In each trial, the training dataset is shuffled randomly. We employ one-vs-all method to train multi-class classifier. The parameters for each algorithm are chosen so that the speeds of initial convergence of error rate are nearly the same, and are shown in Table 4.1 and 4.2 for handwritten digit classification and text classification. In classification experiments, we use $r = 1$ and $\boldsymbol{Q}_t = \boldsymbol{I}$.

4.1.1 Handwritten Digit Classification

MNIST is handwritten digit dataset by [49]. We have 28×28 pixel data with gray scale value normalized in range of $[0, 1]$ and each data is labeled by a digit from 0 to

Table 4.1: Handwritten digit classification: parameters.

Algorithms	λ	η
AdaDelta	-	-
AdaGrad-Fobos	5×10^{-3}	0.1
AdaGrad-RDA	10^{-4}	3.4
Adam	-	5×10^{-5}
RDA	5×10^{-4}	1.5
PDA	10^{-4}	0.15

Table 4.2: Text classification: parameters.

Algorithms	λ	η
AdaDelta	0.1	10^{-5}
AdaGrad-Fobos	4×10^{-6}	0.1
AdaGrad-RDA	10^{-7}	0.1
Adam	0.1	0.1
RDA	10^{-8}	1.4
PDA	0.05	0.3

9. The purpose is to learn a linear classifier, which can recognize the number from the handwritten image. Figure 4.1a shows the learning curve of error rate for each algorithm and you can see that PDA, the proposed algorithm, achieves the lowest validation error for almost entire iteration. Although Adam also achieves much lower error than other algorithms except PDA, the sparsity of Adam is 9.2%, while PDA is 29.9% (sparsities for compared algorithms are shown by Figure 4.1b). To see the effect of the sparsity, we visualize the normalized magnitude of the estimated coefficient by PDA and Adam by Figure 4.2. The coefficient estimated by Adam is seemed vaguer than that by PDA, and it can be said that the coefficient by PDA focuses on wholes and edges of digits more than Adam. Also there are contrasts among the elements of coefficient. By those features, assumed to be achieved by PDA's effectiveness of regularization, PDA can learn more accurate classifier, which can improve the error rate, than existing algorithms.

4.1.2 Text Classification

RCV is text dataset by [50] based on news data with its category label. Each data has bag of words represented text data with four label (Economics, Industrial, Social, and Markets) and multiple labels could be attached. The purpose is to learn a linear classifier to estimate which category the given news text belongs to by the feature of bag-of-words representation, where the frequency of occurrence of each word is used as a feature vector. Figure 4.3a shows the learning curve of error rate for each algorithm and you can see that PDA achieves the lowest validation error.

Table 4.3: sparse system identification: parameters.

Algorithms	λ	η	α	r
PAPA	-	0.14	0.8	1
APFBS	10^{-6}	0.14	0.8	1
RDA	10^{-3}	0.01	-	-
AdaGrad	10^{-3}	0.17	-	-
PDA	3	0.13	0.8	1

Sparsity for each algorithm is shown by Figure 4.3b and sparsity for each algorithm at the end of the iteration is RDA (14.2%), AdaGrad-RDA (19.7%), AdaGrad-Fobos (0.4%), AdaDelta (0.3%), Adam (0.3%), and PDA (35.9%). So, as is the case of MNIST classification, PDA achieves the sparsest solution and this seems to contribute the high performance of PDA, since bag-of-words representation can be very sparse since the size of coefficient is the number of vocabulary in all text data, and each text data contains a few of the vocabulary. We can say that PDA successfully extract the sparse structure of the text dataset and utilize it to gain the accuracy in this text classification setting

4.2 Regression

We compare the proposed algorithm with PAPA [39, 40], APFBS [25, 44], RDA [27], and AdaGrad-RDA [16]. The RDA, AdaGrad-RDA algorithms use $\varphi_t^{\mathrm{LS}}(\boldsymbol{w})$ and $\psi_t(\boldsymbol{w}) := \lambda \|\boldsymbol{w}\|_{\boldsymbol{I},1}$. The other algorithms use $\varphi_t(\boldsymbol{w})$ in (2.6) and the weighted ℓ_1 norm in (3.6). We use the system mismatch $\|\boldsymbol{w}_* - \boldsymbol{w}_t\|_{\boldsymbol{I}}^2 / \|\boldsymbol{w}_*\|_{\boldsymbol{I}}^2$ as a performance measure for all regression experiments except nonlinear model estimation, which is evaluated by mean squared error (MSE). Although there are some possible choice for $\boldsymbol{Q}_t$ such as in [19, 20, 38, 45], the metric in (3.5) are used for PAPA, APFBS, and PDA for fairness. In both experiments, the system mismatch (MSE for nonlinear model estimation) is averaged over 300 independent trials. The parameters for each algorithm are chosen so that the speeds of initial convergence of system mismatch are nearly the same, and are shown in Table 4.3 and 4.4 for sparse system identification and echo cancellation problem.

4.2.1 Sparse System Identification

We let the proportion of the zero components of the true coefficient vector $\boldsymbol{w}_* \in \mathbb{R}^{1000}$ be 80%, and the nonzero components are selected randomly from $[-4, 4]$. The noise ν_t is zero-mean i.i.d. Gaussian with variance 0.01. The input vector $\boldsymbol{x}_t \in \mathbb{R}^{1000}$ is randomly drawn from the i.i.d. uniform distribution over $[-2, 2]$.

Figure 4.4a depicts the learning curves. One can see that the almost entire performance of PDA outperforms the other algorithms. The proportion of the zero components of the estimated coefficient vector is given as follows: PAPA (0%), APFBS (0%), RDA (11.6%), AdaGrad-RDA (89%), and PDA (90%). PDA and

Table 4.4: Echo cancellation: parameters.

Algorithms	λ	η	α	r
PAPA	-	0.3	0.2	2
APFBS	10^{-3}	0.2	0.3	2
RDA	10^{-4}	1	-	-
AdaGrad	10^{-4}	0.3	-	-
PDA	0.026	0.2	0.3	2

AdaGrad-RDA estimates the zero components accurately (see Section 3.6). Figure 4.4b depicts the sparsity. One can see that PDA achieves an accurate sparsity-level remarkably faster than the other algorithms.

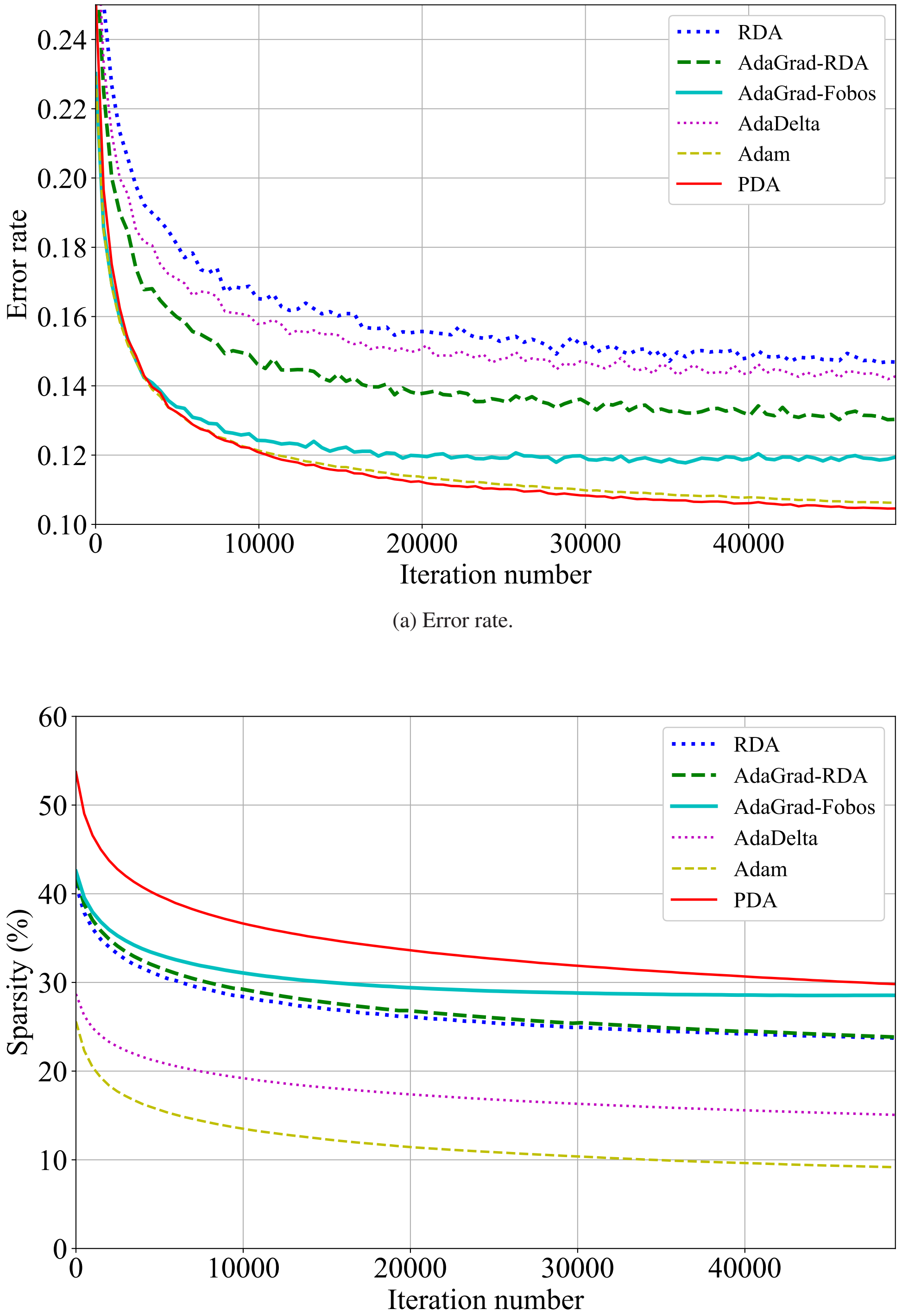
4.2.2 Echo Cancellation

Figure 4.5 shows the amplitude of speech signal and the echo path used in the experiments. The sampling frequency of speech signal and echo path is 8000 Hz. The learning is stopped whenever the amplitude of input signal is below 10^{-4}. The noise is zero-mean i.i.d. Gaussian with the signal noise ratio (SNR) 20 dB. Figure 4.6a shows the learning curves. Figure 4.6b shows the proportion of the zero components of the estimated coefficient vector, which is given as follows: PAPA (0%), APFBS (4.1%), RDA (51.4%), AdaGrad-RDA (90.0%), and PDA (64.8%). Note here that the regularization parameter for each algorithm is chosen to give the best convergence behaviors. In this experiment, even though the echo path has not any zero components, there are several large scale components, which are important to estimate. PDA estimates those large important elements by reducing the estimation variance by its regularization capability, and yields the best estimation with the best convergence behavior. Also the use of the metric $\boldsymbol{Q}_t$ allows PAPA, APFBS, and PDA to attain fast initial convergence. In addition, PDA achieves the lowest system mismatch due to the strong regularization.

4.2.3 Nonlinear model estimation

Finally, we apply PDA to multikernel adaptive filtering [51, 52], which aims to estimate nonlinear system. Figure 4.7a shows the nonlinear system, what we want to estimate in this experiment. To see derivations of the PDA algorithm for multikernel, we need to step into the multikernel adaptive filtering theory [51]. However, the theory of multikernel is out of scope of this paper, so we place more detail about the algorithm's derivation and experiments setting in Appendix A.3.

Figure 4.7b shows the learning curve, where we compare APFBS application for multikernel adaptive filtering [53] with PDA application. One can see that the entire performance is better than APFBS. In terms of sparsity, Figure 4.7c depicts the dictionary size, which is the number of functions used for the estimation, and PDA uses less dictionary functions than APFBS. So, it can be said that PDA is less redundant than APFBS and it increases MSE.



(a) Error rate.

(b) Sparsity.

Figure 4.1: Handwritten digit classification.

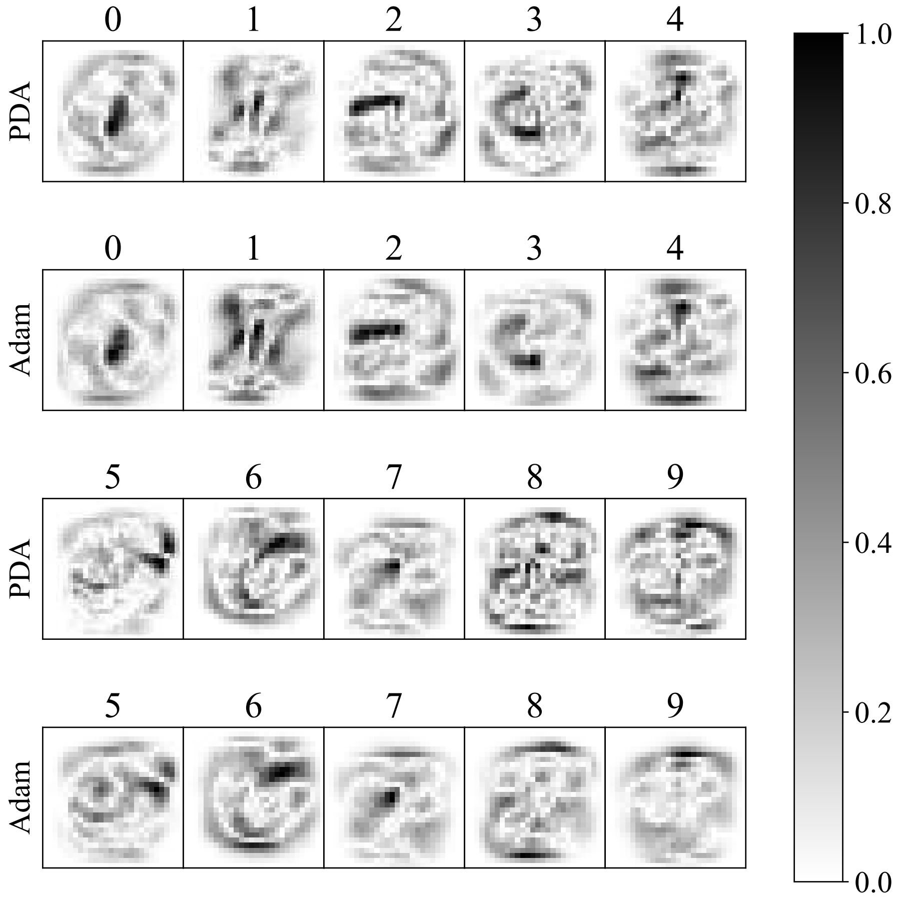
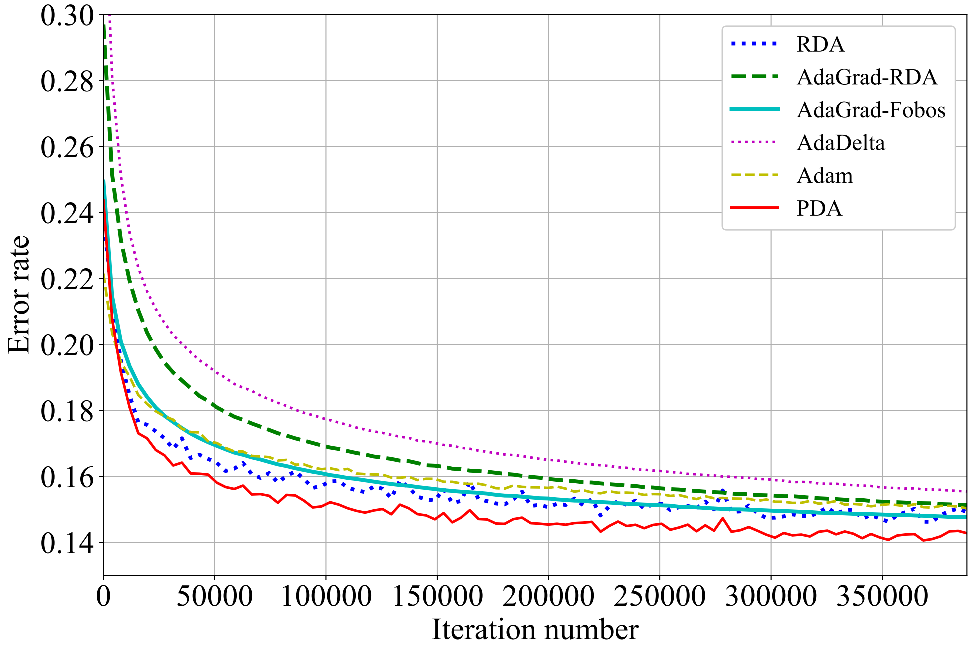
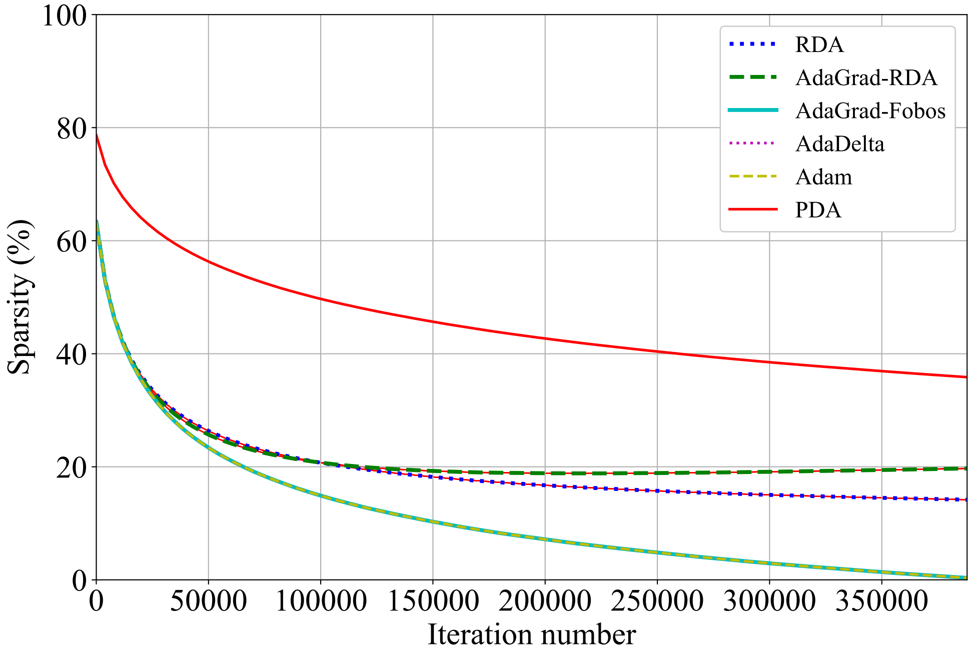


Figure 4.2: Visualization of the estimated coefficient for Adam and PDA.

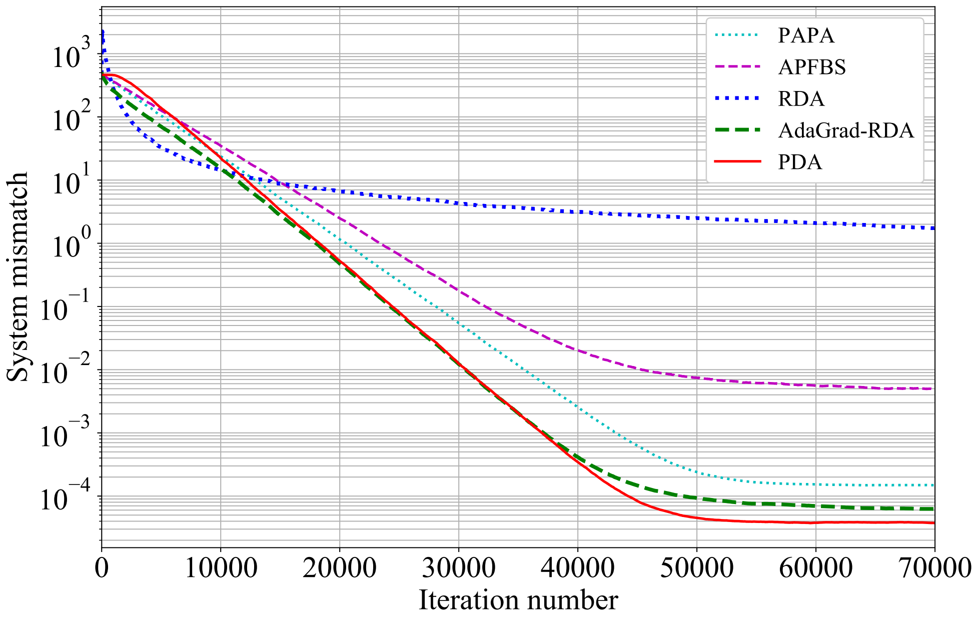


(a) Error rate.

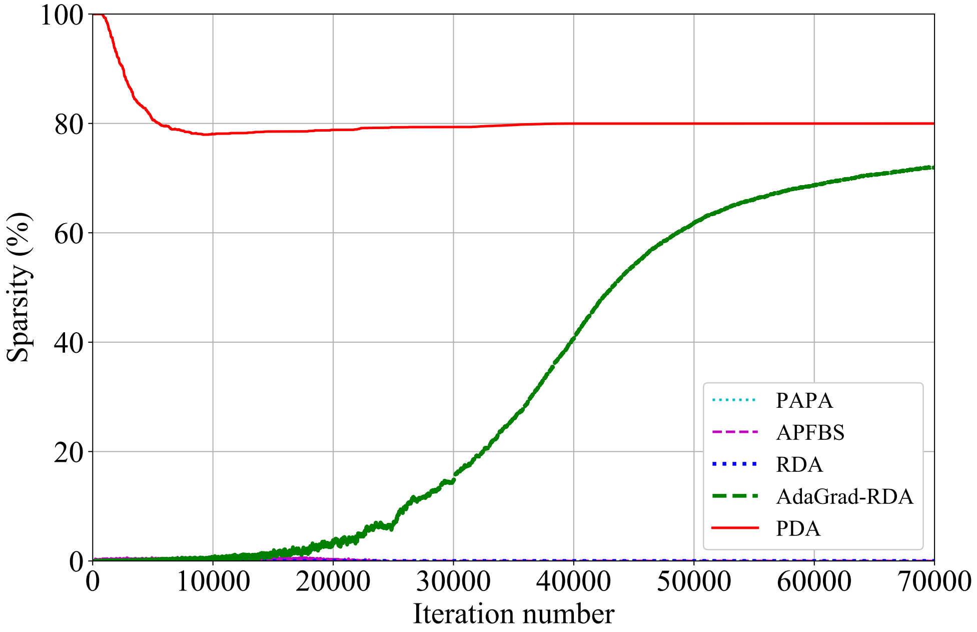


(b) Sparsity.

Figure 4.3: Text classification.



(a) System mismatch.



(b) Sparsity.

Figure 4.4: Sparse system identification.

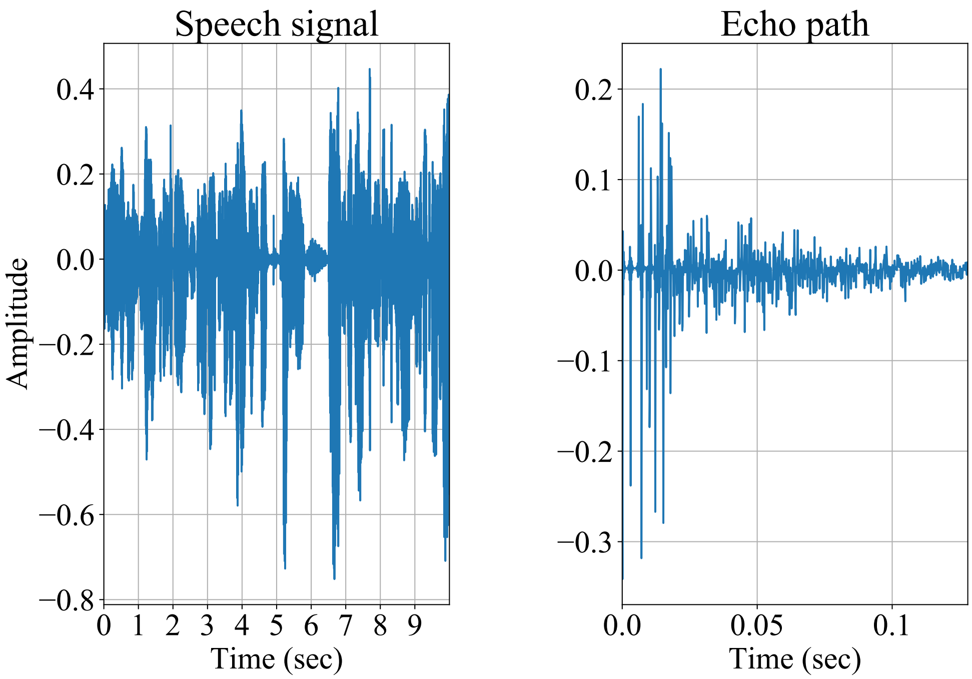
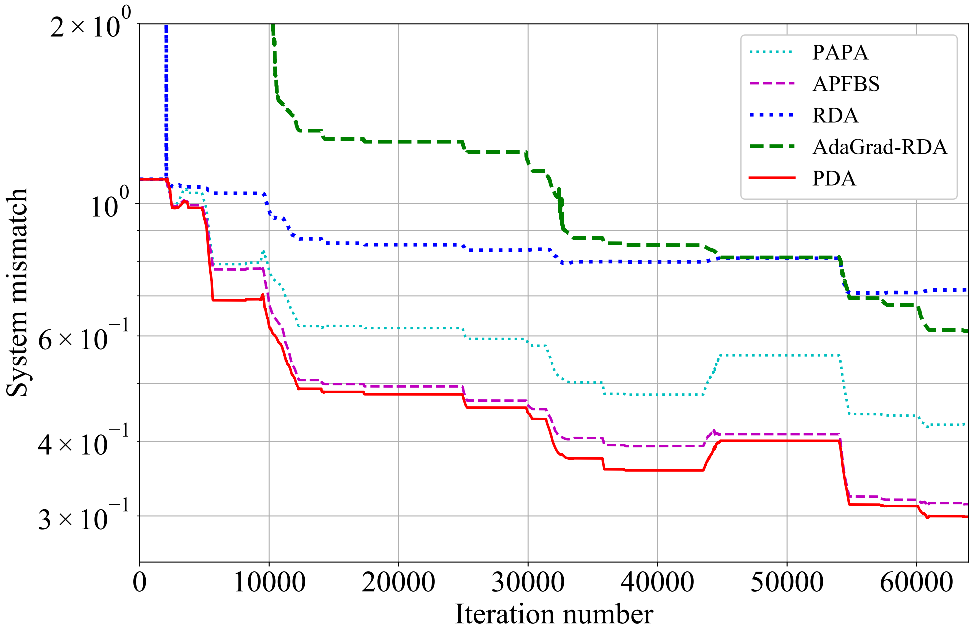
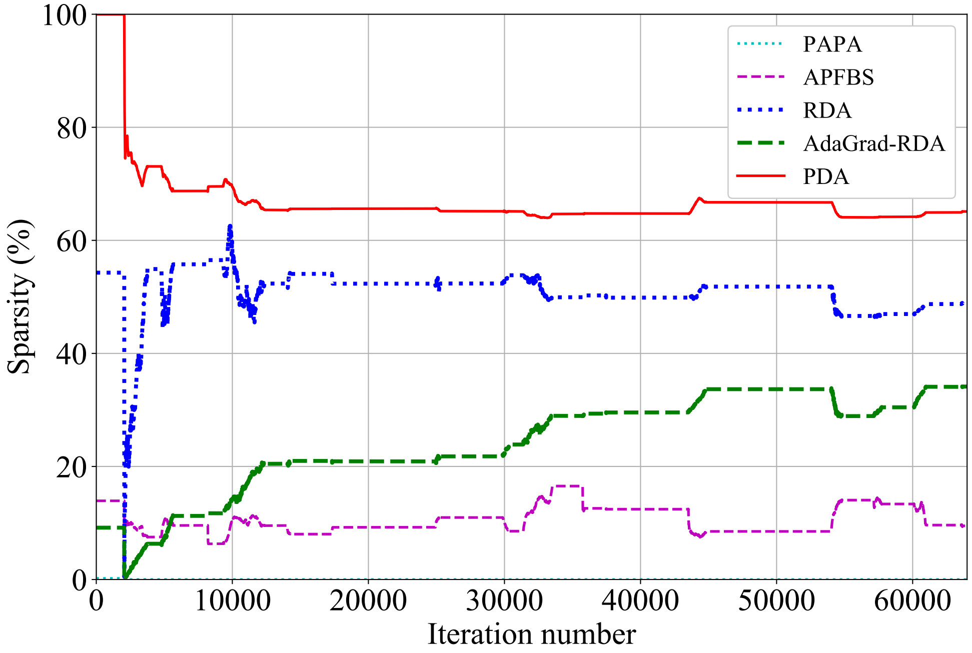


Figure 4.5: Echo path and speech signal.

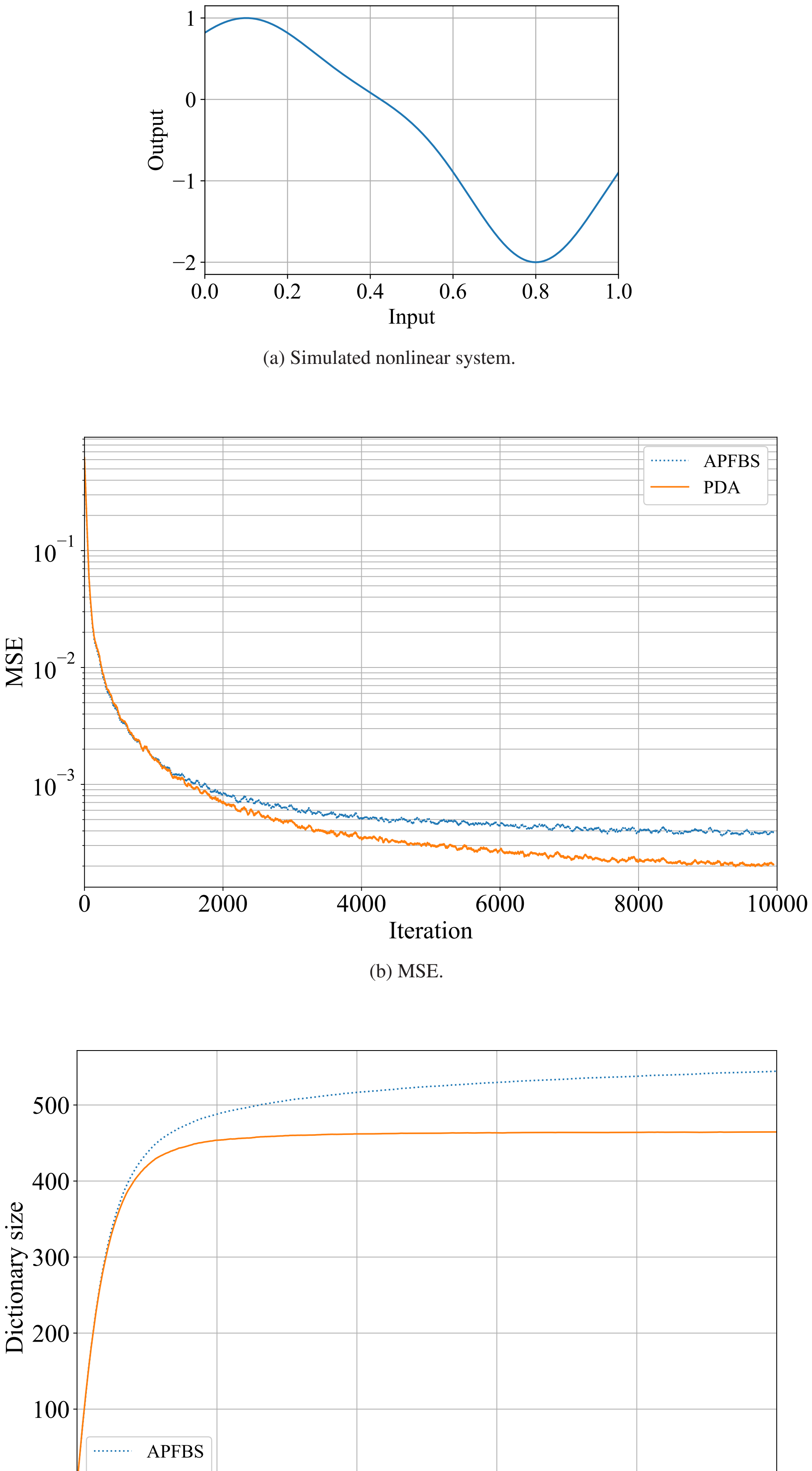


(a) System mismatch.



(b) Sparsity.

Figure 4.6: Echo cancellation.



(a) Simulated nonlinear system.

(b) MSE.

Chapter 5

Conclusion

We presented the projection-based regularized dual averaging (PDA), which features the input-vector normalization that came from the squared-distance function to a closed convex set, the sparsity-seeking variable-metric, and a constant step size without any undesirable biases for regularization. Also a weighted ℓ_1 regularization was proposed to offset the bias, due to the sparsity-promoting metric. variable-metric. Although the squared-distance function has been used in many adaptive filtering algorithms including NLMS, APA, APSM, and APFBS, its application to the dual averaging method has not been studied previously to the best of authors' knowledge. An application of PDA to an online classification and regression problem was presented. The numerical examples, including regression and classification with synthetic/real data demonstrated that PDA performed the best with better sparsity-seeking property compared to the existing methods including AdaGrad and APFBS.

Appendix A

Appendix

A.1 Regret Analysis

In this section, we investigate the regret bound (A.1) of PDA in the case of that $\boldsymbol{Q}_t = \boldsymbol{I}$, which reduces the regularizer ψ_t, defined by (3.6), to time invariant regularizer ψ. We denote by $(\boldsymbol{w}_\tau)_{\tau=0,\cdots,t}$ the sequence of the estimation, by $(\boldsymbol{g}_\tau)_{\tau=1,\cdots,t}$ the sequence of the gradient, and by $(\boldsymbol{s}_\tau)_{\tau=1,\cdots,t}$ the sequence of the sum of gradient, generated by (3.4). In stochastic regularized optimization, the regret with respect to any fixed $\boldsymbol{w} \in \mathbb{R}^n$ is

$$R_t(\boldsymbol{w}) := \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w}_{\tau-1}) + \psi(\boldsymbol{w}_{\tau-1}) - (\varphi_\tau(\boldsymbol{w}) + \psi(\boldsymbol{w}))\right]. \tag{A.1}$$

At first, we define the set for a constant $D > 0$,

$$\mathcal{F}_D := \left\{ \boldsymbol{w} \in \mathbb{R}^n \,\middle|\, \frac{\|\boldsymbol{w}\|_{\boldsymbol{I}}^2}{2} \leq D^2 \right\}, \tag{A.2}$$

and following two type of conjugate functions:

$$U(\boldsymbol{s}) : = \max_{\boldsymbol{w} \in \mathcal{F}_D} \{\langle \boldsymbol{s}, \boldsymbol{w} \rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w})\} \tag{A.3}$$

$$V(\boldsymbol{s}) : = \max_{\boldsymbol{w}} \left\{\langle \boldsymbol{s}, \boldsymbol{w} \rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w}) - \frac{1}{2\eta} \|\boldsymbol{w}\|_{\boldsymbol{I}}^2\right\} \tag{A.4}$$

Then, we prove two lemmas in order to see the upper bound of the regret.

Lemma 1. *For any $\boldsymbol{s} \in \mathbb{R}^n$, we have*

$$U(\boldsymbol{s}) \leq V(\boldsymbol{s}) + \frac{D^2}{\eta}. \tag{A.5}$$

Proof: It can be derived the definition of $U(\boldsymbol{s})$ and $V(\boldsymbol{s})$ as,

$$\begin{aligned}
U(\boldsymbol{s}) &= \max_{\boldsymbol{w}\in\mathcal{F}_D} \{\langle \boldsymbol{s}, \boldsymbol{w}\rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w})\} \\
&\leq \max_{\boldsymbol{w}} \left\{\langle \boldsymbol{s}, \boldsymbol{w}\rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w}) + \frac{D^2 - \|\boldsymbol{w}\|_{\boldsymbol{I}}^2/2}{\eta}\right\} \\
&= V(\boldsymbol{s}) + \frac{D^2}{\eta}.
\end{aligned}$$

□

Lemma 2. *For any $\boldsymbol{s}, \boldsymbol{g} \in \mathbb{R}^n$,*

1. *The function $V(\boldsymbol{s})$ is convex and differentiable.*

2. *The gradient of $V(\boldsymbol{s})$ is given by*

$$\nabla V(\boldsymbol{s}) = \pi(\boldsymbol{s}) \tag{A.6}$$

where

$$\begin{aligned}
\pi(\boldsymbol{s}) :&= \arg\max_{\boldsymbol{w}} \left\{\langle \boldsymbol{s}, \boldsymbol{w}\rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w}) - \frac{1}{2\eta}\|\boldsymbol{w}\|_{\boldsymbol{I}}^2\right\} \\
&= \arg\min_{\boldsymbol{w}} \left\{\langle -\boldsymbol{s}, \boldsymbol{w}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}) + \frac{1}{2\eta}\|\boldsymbol{w}\|_{\boldsymbol{I}}^2\right\}.
\end{aligned} \tag{A.7}$$

3. *The gradient $\nabla V(\boldsymbol{s})$ is Lipschitz continuous with η:*

$$\|\nabla V(\boldsymbol{s}_1) - \nabla V(\boldsymbol{s}_2)\|_{\boldsymbol{I}}^2 \leq \eta \|\boldsymbol{s}_1 - \boldsymbol{s}_2\|_{\boldsymbol{I}}^2 \tag{A.8}$$

and

$$V(\boldsymbol{s} + \boldsymbol{g}) \leq V(\boldsymbol{s}) + \langle \boldsymbol{g}, \nabla V(\boldsymbol{s})\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}\|_{\boldsymbol{I}}^2. \tag{A.9}$$

Proof: The function $\psi(\boldsymbol{w}) + \frac{1}{2\eta}\|\boldsymbol{w}\|_{\boldsymbol{I}}^2$ is $\frac{1}{\eta}$−strongly convex function, so we can apply theorem 1 in [54] to $V(\boldsymbol{s})$ to derive the results. □

Theorem 1. *If there exist $G > 0$ and $L > 0$ such that $\|\boldsymbol{g}_t\|_{\boldsymbol{I}}^2 \leq G$ and $\psi(\boldsymbol{w}_t) \leq L$ for any t,*

$$R_t(\boldsymbol{w}) \leq \frac{\eta t(G + L)}{2} + \frac{D^2}{\eta}, \quad \forall \boldsymbol{w} \in \mathcal{F}_D. \tag{A.10}$$

Proof: By the assumption $\boldsymbol{w} \in \mathcal{F}_D$, the regret can be rewritten as,

$$\begin{aligned}
R_t(\boldsymbol{w}) &= \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w}_{\tau-1}) + \psi(\boldsymbol{w}_{\tau-1}) - (\varphi_\tau(\boldsymbol{w}) + \psi(\boldsymbol{w}))\right] \\
&\leq \sum_{\tau=1}^{t} \left[\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1} - \boldsymbol{w}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right] - t\psi(\boldsymbol{w}) \\
&\leq \sum_{\tau=1}^{t} \left[\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1} - \boldsymbol{w}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right] - \psi(\boldsymbol{w}) \\
&\leq \max_{\boldsymbol{w}\in\mathcal{F}_D} \left\{ \left[\sum_{\tau=1}^{t} \langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1} - \boldsymbol{w}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right] - \psi(\boldsymbol{w}) \right\} \\
&= \sum_{\tau=1}^{t} \left(\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right) + \max_{\boldsymbol{w}\in\mathcal{F}_D} \left\{\langle \boldsymbol{s}_t, -\boldsymbol{w}\rangle_{\boldsymbol{I}} - \psi(\boldsymbol{w})\right\} \\
&=: \delta_t. \qquad \text{(A.11)}
\end{aligned}$$

The first inequality uses $\varphi_\tau(\boldsymbol{w}_\tau) - \varphi_\tau(\boldsymbol{w}) \leq \langle \boldsymbol{g}_\tau, \boldsymbol{w}_\tau - \boldsymbol{w}\rangle$. By lemma 1, the upper bound (A.11) can be bounded by

$$\begin{aligned}
\delta_t &= \sum_{\tau=1}^{t} \left(\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right) + U(-\boldsymbol{s}_t) \\
&\leq \sum_{\tau=1}^{t} \left(\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1}\rangle_{\boldsymbol{I}} + \psi(\boldsymbol{w}_{\tau-1})\right) + V(-\boldsymbol{s}_t) + \frac{D^2}{\eta}. \qquad \text{(A.12)}
\end{aligned}$$

By lemma 2,

$$\begin{aligned}
V(-\boldsymbol{s}_t) &= V(-\boldsymbol{s}_{t-1} - \boldsymbol{g}_t) \\
&\leq V(-\boldsymbol{s}_{t-1}) - \langle \boldsymbol{g}_t, \nabla V(-\boldsymbol{s}_{t-1})\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}_t\|_{\boldsymbol{I}}^2 \\
&\leq V(-\boldsymbol{s}_{t-1}) - \langle \boldsymbol{g}_t, \pi(-\boldsymbol{s}_{t-1})\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}_t\|_{\boldsymbol{I}}^2. \qquad \text{(A.13)}
\end{aligned}$$

Since $\pi(-\boldsymbol{s}_t)$ corresponds to the update of PDA (3.4),

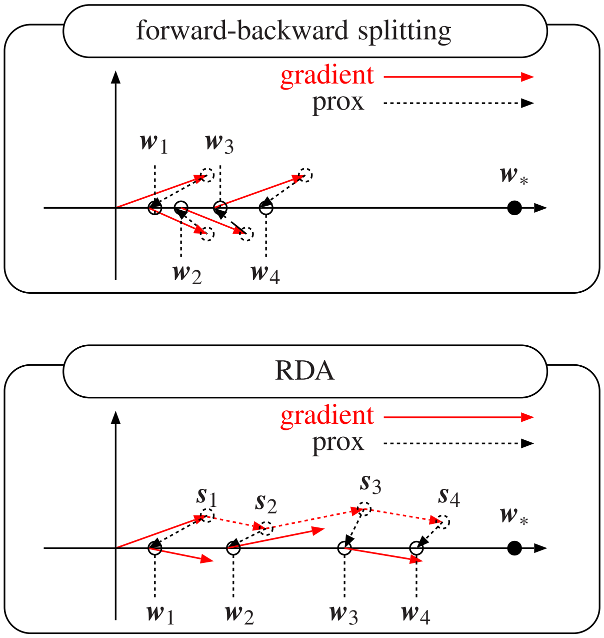
$$V(-\boldsymbol{s}_t) \leq V(-\boldsymbol{s}_{t-1}) - \langle \boldsymbol{g}_t, \boldsymbol{w}_{t-1}\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}_t\|_{\boldsymbol{I}}^2, \qquad \text{(A.14)}$$

so

$$V(-\boldsymbol{s}_t) - V(-\boldsymbol{s}_{t-1}) \leq -\langle \boldsymbol{g}_t, \boldsymbol{w}_{t-1}\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}_t\|_{\boldsymbol{I}}^2. \qquad \text{(A.15)}$$

By summing (A.15) for $t = 1, 2, \cdots, t$, and noting that $\boldsymbol{s}_0 := \boldsymbol{0}$, we arrive at

$$V(-\boldsymbol{s}_t) \leq \sum_{\tau=1}^{t} \left[-\langle \boldsymbol{g}_\tau, \boldsymbol{w}_{\tau-1}\rangle_{\boldsymbol{I}} + \frac{\eta}{2}\|\boldsymbol{g}_\tau\|_{\boldsymbol{I}}^2\right]. \qquad \text{(A.16)}$$


forward-backward splitting

RDA

Figure A.1: Illustrations of forward-backward splitting and RDA.

Finally, by (A.11) and (A.12), we attain,

$$R_t(\boldsymbol{w}) \leq \delta_t \leq tL + \frac{\eta t G}{2} + \frac{D^2}{\eta}. \tag{A.17}$$

□

From theorem 1, we can conclude that the regret of $(\boldsymbol{w}_\tau)_{\tau=0,1,\cdots,t}$ is bounded by $tL + \eta t G/2 + D^2/\eta$ for all $\boldsymbol{w} \in \mathcal{F}_D$ under the assumptions of theorem 1 (the upper bound of the regret for RDA with same setting is $\eta t G/2 + D^2/\eta$).

A.2 Forward Backward Splitting

To solve (2.1) in the case of $\psi_t = 0$, we can use SGD to update the previous estimate $\boldsymbol{w}_{t-1} \in \mathbb{R}^n$ by

$$\boldsymbol{w}_t : = \boldsymbol{w}_{t-1} - \eta_t \boldsymbol{g}_t, \quad \boldsymbol{g}_t \in \nabla \varphi_t(\boldsymbol{w}_{t-1}) \tag{A.18}$$

where $(\eta_\tau)_{\tau=1,2,\cdots,t}$ is the step size sequence.

Then, for $\psi_t \neq 0$, one can apply

$$\boldsymbol{w}_t := \mathrm{prox}^{\boldsymbol{I}_t}_{\eta_t \psi_t}\left(\boldsymbol{w}_{t-1} - \eta_t \boldsymbol{g}_t\right), \tag{A.19}$$

where $\boldsymbol{I}$ is the $n \times n$ identity matrix, and the proximity operator is defined, for $\boldsymbol{w} \in \mathbb{R}^n$, as [55, 56]

$$\mathrm{prox}_{\eta\psi_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}) := \arg\min_{\boldsymbol{z}\in\mathbb{R}^n} \left(\eta\psi_t(\boldsymbol{w}) + \frac{1}{2} \|\boldsymbol{w} - \boldsymbol{z}\|_{\boldsymbol{Q}_t}^2 \right). \tag{A.20}$$

This type of algorithm is called forward-backward splitting [57].

A.2.1 Difference Between Forward Backward Splitting and RDA

Figure A.1 shows the difference between the forward-backward splitting and RDA. One can see that the effects of the proximity operator accumulate over the iteration. This actually increases the estimation biases, and APFBS therefore has a tradeoff between the strength of regularization and the estimation accuracy. RDA is free from the accumulation issue, yielding high estimation accuracy together with a high level of sparsity.

A.3 Application for Multikernel Adaptive Filter

Here we introduce multikernel adaptive filtering [51, 52] and PDA application. In multikernel adaptive filtering, we want to estimate a nonlinear function such as

$$y = f(\boldsymbol{x}), \quad y \in \mathbb{R}, \quad \boldsymbol{x} \in \mathbb{R}^n \tag{A.21}$$

by multikernel modeling defined as

$$f_t(\boldsymbol{x}) := \sum_{m\in\mathcal{M}} \underbrace{\sum_{j\in\mathcal{J}_t} h_{j,t}^{(m)} k_m(\boldsymbol{x}, \boldsymbol{x}_j)}_{m\text{th model}}, \quad h_{j,t}^{(m)} \in \mathbb{R} \tag{A.22}$$

where $k_m : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, $m \in \mathcal{M} := \{1, 2, \cdots, M\}$ is the set of positive definite kernels to be used and $\{k_m(\cdot, \boldsymbol{x}_j)\}_{m\in\mathcal{M}, j\in\mathcal{J}_t}$ is the dictionary indicated by the dictionary index set $\mathcal{J}_t := \{j_1^{(t)}, j_2^{(t)}, , \cdots , , j_{r_t}^{(t)}\} \subset \{1, 2, \cdots, t\}$ where r_t is the size of the dictionary index set $\mathcal{J}_t$. Figure 4.7c shows r_t through the iteration. The model (A.22) can be rewritten as

$$f_t(\boldsymbol{x}_t) = \langle \boldsymbol{H}_t, \boldsymbol{K}_t \rangle \tag{A.23}$$

where $\langle \boldsymbol{A}, \boldsymbol{B} \rangle := \mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{B})$, the norm is $\|\boldsymbol{A}\| := \sqrt{\langle \boldsymbol{A}, \boldsymbol{A} \rangle}$ for any same sized matrix $\boldsymbol{A}, \boldsymbol{B}$ and

$$\begin{aligned}
\boldsymbol{H}_t &: = \left[\boldsymbol{h}_{j_1^t,t}, \boldsymbol{h}_{j_2^t,t}, \cdots, \boldsymbol{h}_{j_{r_t}^t,t}\right] \in \mathbb{R}^{M\times r_t}, \\
\boldsymbol{h}_{j,t} &: = \left[h_{j,t}^{(1)}, h_{j,t}^{(2)}, , \cdots , , h_{j,t}^{(M)}\right] \in \mathbb{R}^{M}, \\
\boldsymbol{K}_t &: = \left[\boldsymbol{k}_{j_1^t,t}, \boldsymbol{k}_{j_2^t,t}, \cdots, \boldsymbol{k}_{j_{r_t}^t,t}\right] \in \mathbb{R}^{M\times r_t}, \\
\boldsymbol{k}_{j,t} &:= \left[k_1(\boldsymbol{x}_t, \boldsymbol{x}_j), k_2(\boldsymbol{x}_t, \boldsymbol{x}_j), \cdots, k_M(\boldsymbol{x}_t, \boldsymbol{x}_j)\right] \in \mathbb{R}^{M}.
\end{aligned}$$

As loss function, we use

$$l_t(\boldsymbol{H}) := \varphi_t(\boldsymbol{H}) + \psi_t(\boldsymbol{H}), \tag{A.24}$$

$$\varphi_t(\boldsymbol{H}) := \frac{1}{2} d^2(\boldsymbol{H}, C_t), \quad \psi_t(\boldsymbol{H}) = \lambda \sum_{j \in \mathcal{J}_t} \left\| \boldsymbol{h}_j \right\| \tag{A.25}$$

where

$$d(\boldsymbol{H}, C_t) := \min_{\boldsymbol{Y} \in C_t} \|\boldsymbol{H} - \boldsymbol{Y}\|, \tag{A.26}$$

$$C_t := \{\boldsymbol{H} \in \mathbb{R}^{M \times r_t} : \langle \boldsymbol{H}, \boldsymbol{K}_t \rangle = y_t\}. \tag{A.27}$$

The projection onto (A.27) is given by

$$\nabla \varphi_t(\boldsymbol{H}) = \boldsymbol{H} - P_{C_t}(\boldsymbol{H}), \tag{A.28}$$

$$\begin{aligned} P_{C_t}(\boldsymbol{H}) &:= \arg\min_{\boldsymbol{Y} \in C_t} \|\boldsymbol{H} - \boldsymbol{Y}\|^2 \\ &= \boldsymbol{H} - \frac{\langle \boldsymbol{H}, \boldsymbol{K}_t \rangle - y_t}{\|\boldsymbol{K}_t\|^2} \boldsymbol{K}_t. \end{aligned} \tag{A.29}$$

We compare PDA with APFBS, that has been applied to multikernel adaptive filtering by [53]:

$$\hat{\boldsymbol{H}}_t = \boldsymbol{H}_{t-1} - \eta \nabla \varphi_t(\boldsymbol{H}_{t-1}), \tag{A.30}$$

$$\begin{aligned} \boldsymbol{H}_t &= \operatorname{prox}_{\eta \psi_t}\left(\hat{\boldsymbol{H}}_t\right) \\ &= \sum_{j \in \mathcal{J}_t} \max\left\{1 - \frac{\lambda \eta}{\left\|\hat{\boldsymbol{h}}_{j,t}\right\|}, 0\right\} \hat{\boldsymbol{h}}_{j,t} \boldsymbol{e}_{j,r_n}^{\mathsf{T}}, \end{aligned} \tag{A.31}$$

where $\boldsymbol{e}_{j,r_n}$ is a length-r_n unit vector that has one at the jth entry and zeros elsewhere. PDA algorithm for multikernel adaptive filtering can be derived as

$$\hat{\boldsymbol{H}}_t = -\eta \sum_{\tau=1}^{t} \nabla \varphi_\tau(\boldsymbol{H}_{\tau-1}), \tag{A.32}$$

$$\boldsymbol{H}_t = \operatorname{prox}_{\eta \psi_t}\left(\hat{\boldsymbol{H}}_t\right). \tag{A.33}$$

In addition, following [53], we use dictionary update such as

```
Dictionary update
Iteration : for time instance $t$
1. Update dictionary $\mathcal{J}_t := \mathcal{J}_{t-1} \cup \{t\}$
2. Update coefficient $\boldsymbol{H}_{t-1}$ to $\boldsymbol{H}_t$
   by APFBS (A.31) or PDA (A.33)
3. [If $r_t \geq r_{\max}$] Refine dictionary
   $\mathcal{J}_t := \{j \in \mathcal{J}_t : \left\|\boldsymbol{h}_{j,t}\right\| \geq \epsilon_{\mathcal{J}}\}, \quad \epsilon_{\mathcal{J}} > 0$
```

Table A.1: Nonlinear model selection: parameters.

Algorithms	η	λ	$\epsilon_{\mathcal{J}}$	$r_{\max}$
APFBS	0.3	10^{-8}	10^{-12}	20
PDA	0.3	10^{-6}	10^{-12}	20

As experiment setting, following [58], we use the normalized gaussian kernel,

$$k_m(\boldsymbol{d}, \boldsymbol{u}) := \frac{1}{\sqrt{2\pi\sigma_m}} \exp\left(-\frac{\|\boldsymbol{d} - \boldsymbol{u}\|^2}{2\sigma_m}\right), \quad \boldsymbol{d}, \boldsymbol{u} \in \mathbb{R}^n \tag{A.34}$$

where σ_m is the variance, which set to $a \times 10^b$, $a \in \{1, 2, , \cdots, , 10\}$, $b \in \{-4, -3, , \cdots, , 1\}$. Other parameters are summarized in Table A.1. As true function f, with input $x_t \in \mathbb{R}$, outputs $y_t \in \mathbb{R}$, and white noise $u_t \sim \mathcal{N}(0, 0.01)$, we use

$$y_t := \exp\left(-20(x_t - 0.1)^2\right) - 2\exp\left(-20(x_t - 0.8)^2\right) + u_t \tag{A.35}$$

where $x_t \sim \mathcal{U}[0, 1]$. The system (A.35) is used in [53] and is visualized by Figure 4.7a.

Bibliography

[1] H. Kushner and G. G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer Science & Business Media, 2003, vol. 35.

[2] K. Slavakis, S.-J. Kim, G. Mateos, and G. B. Giannakis, "Stochastic approximation vis-a-vis online learning for big data analytics [lecture notes]," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 124–129, 2014.

[3] A. Bordes, L. Bottou, and P. Gallinari, "Sgd-qn: Careful quasi-newton stochastic gradient descent," *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1737–1754, 2009.

[4] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[5] T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learning rates," in *International Conference on Machine Learning*, 2013, pp. 343–351.

[6] T. Schaul and Y. LeCun, "Adaptive learning rates and parallelization for stochastic, sparse, non-smooth gradients," in *International Conference on Learning Representations*, Scottsdale, AZ, 2013.

[7] O. Vinyals and D. Povey, "Krylov subspace descent for deep learning," in *Artificial Intelligence and Statistics*, 2012, pp. 1261–1268.

[8] N. N. Schraudolph, "Fast curvature matrix-vector products for second-order gradient descent," *Neural computation*, vol. 14, no. 7, pp. 1723–1738, 2002.

[9] J. Martens, "Deep learning via hessian-free optimization," in *Proceedings of the 27th International Conference on Machine Learning*, 2010, pp. 735–742.

[10] R. H. Byrd, S. L. Hansen, J. Nocedal, and Y. Singer, "A stochastic quasi-newton method for large-scale optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1008–1031, 2016.

[11] N. N. Schraudolph, J. Yu, and S. Günter, "A stochastic quasi-newton method for online convex optimization," in *Artificial Intelligence and Statistics*, 2007, pp. 436–443.

[12] A. Mokhtari and A. Ribeiro, "Res: Regularized stochastic bfgs algorithm," *IEEE Transactions on Signal Processing*, vol. 62, no. 23, pp. 6089–6104, 2014.

[13] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural computation*, vol. 10, no. 2, pp. 251–276, 1998.

[14] N. L. Roux, P.-A. Manzagol, and Y. Bengio, "Topmoumoute online natural gradient algorithm," in *Advances in neural information processing systems*, 2008, pp. 849–856.

[15] N. L. Roux and A. W. Fitzgibbon, "A fast natural newton method," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 623–630.

[16] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[17] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[18] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning and Representations*, 2015, pp. 1–13.

[19] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 5, pp. 508–518, 2000.

[20] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, vol. 1, 1998, pp. 394–398.

[21] S. Makino, Y. Kaneda, and N. Koizumi, "Exponentially weighted stepsize nlms adaptive filter based on the statistics of a room impulse response," *IEEE Trans. Speech and Audio Processing*, vol. 1, no. 1, pp. 101–108, 1993.

[22] M. Yukawa, K. Slavakis, and I. Yamada, "Adaptive parallel quadratic-metric projection algorithms," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1665–1680, 2007.

[23] M. Yukawa and I. Yamada, "A unified view of adaptive variable-metric projection algorithms," *EURASIP J. Advances in Signal Processing*, vol. 2009, Article ID 589260, 13 pages, 2009.

[24] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. IEEE ICASSP*, 2010, pp. 3734–3737.

[25] M. Yamagishi, M. Yukawa, and I. Yamada, “Acceleration of adaptive proximal forward-backward splitting method and its application to sparse system identification,” in *Proc. IEEE ICASSP*, 2011, pp. 4296–4299.

[26] Y. Singer and J. C. Duchi, “Efficient learning using forward-backward splitting,” in *Advances in Neural Information Processing Systems*, 2009, pp. 495–503.

[27] L. Xiao, “Dual averaging method for regularized stochastic learning and online optimization,” in *Advances in Neural Information Processing Systems*, 2009, pp. 2116–2124.

[28] Y. Nesterov, “Primal-dual subgradient methods for convex problems,” *Mathematical Programming*, vol. 120, no. 1, pp. 221–259, 2009.

[29] A. Kalai and S. Vempala, “Efficient algorithms for online decision problems,” *Journal of Computer and System Sciences*, vol. 71, no. 3, pp. 291–307, 2005.

[30] J.-I. Nagumo and A. Noda, “A learning method for system identification,” *IEEE Trans. Automatic Control*, vol. 12, no. 3, pp. 282–287, 1967.

[31] A. E. Albert and L. S. Gardner Jr., *Stochastic Approximation and Nonlinear Regression*. Cambridge MA: MIT Press, 1967.

[32] B. Widrow and M. E. Hoff, “Adaptive switching circuits,” in *IRE WESCON convention record*, vol. 4, no. 1. New York, 1960, pp. 96–104.

[33] B. Widrow and S. D. Stearns, “Adaptive signal processing,” *Englewood Cliffs, NJ, Prentice-Hall, Inc., 1985.*, vol. 1, 1985.

[34] I. Yamada and N. Ogura, “Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions,” *Numer. Funct. Anal. Optim.*, vol. 25, no. 7&8, pp. 593–617, 2004.

[35] K. Slavakis, I. Yamada, and N. Ogura, “The adaptive projected subgradient method over the fixed point set of strongly attracting nonexpansive mappings,” *Numerical Functional Analysis and Optimization*, vol. 27, no. 7-8, pp. 905–930, 2006.

[36] M. Yukawa, K. Slavakis, and I. Yamada, “Multi-domain adaptive learning based on feasibility splitting and adaptive projected subgradient method,” *IEICE Trans. fundamentals of electronics, communications and computer sciences*, vol. 93, no. 2, pp. 456–466, 2010.

[37] S. Theodoridis, K. Slavakis, and I. Yamada, “Adaptive learning in a world of projections: a unifying framework for linear and nonlinear classification and regression tasks,” *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, Jan. 2011.

[38] C. Paleologu, J. Benesty, and S. Ciochin, "An improved proportionate NLMS algorithm based on the ℓ_0 norm," in *Proc. IEEE ICASSP*, 2010, pp. 309–312.

[39] T. Gansler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 6, pp. 656–663, 2000.

[40] J. Benesty, Y. A. Huang, J. Chen, and P. A. Naylor, "Adaptive algorithms for the identification of sparse impulse responses," *Selected Methods for Acoustic Echo and Noise Control*, vol. 5, pp. 125–153, 2006.

[41] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electronics and Communications in Japan (Part I: Communications)*, vol. 67, no. 5, pp. 19–27, 1984.

[42] T. Hinamoto and S. Maekawa, "Extended theory of learning identification," *Electrical Engineering in Japan*, vol. 95, no. 5, pp. 101–107, 1975.

[43] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *Journal of Machine Learning Research*, vol. 7, no. Mar, pp. 551–585, 2006.

[44] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. IEEE ICASSP*, 2010, pp. 3734–3737.

[45] M. Yukawa and I. Yamada, "Two product-space formulations for unifying multiple metrics in set-theoretic adaptive filtering," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2010, pp. 1010–1014.

[46] O. Toda, M. Yukawa, S. Sasaki, and H. Kikuchi, "An efficient adaptive filtering scheme based on combining multiple metrics," *IEICE Trans. Fundamentals*, vol. E97-A, no. 3, pp. 800–808, Mar. 2014.

[47] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization. manuscript submitted to," *SIAM Journal on Optimization*, 2008.

[48] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari, "Composite objective mirror descent." in *Proc. COLT*, 2010, pp. 14–26.

[49] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[50] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "Rcv1: A new benchmark collection for text categorization research," *Journal of machine learning research*, vol. 5, no. Apr, pp. 361–397, 2004.

[51] M. Yukawa, "Multikernel adaptive filtering," *IEEE Trans. Signal Processing*, vol. 60, no. 9, pp. 4672–4682, Sept. 2012.

[52] M. Yukawa and R. Ishii, "On adaptivity of online model selection method based on multikernel adaptive filtering," in *Proc. APSIPA-ASC*, 2013.

[53] ——, "Online model selection and learning by multikernel adaptive filtering," in *Proc. EUSIPCO*, 2013.

[54] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical programming*, vol. 103, no. 1, pp. 127–152, 2005.

[55] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 1st ed. New York: NY: Springer, 2011.

[56] I. Yamada, M. Yukawa, and M. Yamagishi, "Minimizing the Moreau envelope of nonsmooth convex functions over the fixed point set of certain quasi-nonexpansive mappings," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, ser. Optimization and Its Applications, vol. 49. New York: Springer, 2011, pp. 345–390.

[57] P.-L. Lions and B. Mercier, "Splitting algorithms for the sum of two nonlinear operators," *SIAM Journal on Numerical Analysis*, vol. 16, no. 6, pp. 964–979, 1979.

[58] O. Toda and M. Yukawa, "On kernel design for online model selection by gaussian multikernel adaptive filtering," in *Proc. APSIPA Annual Summit and Conference*. IEEE, 2014, pp. 1–5.