RESCALED PURE GREEDY ALGORITHM FOR CONVEX OPTIMIZATION

A Thesis

by

ZHEMING GAO

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Guergana Petrova
Committee Members,	Peter Howard
	Anirban Bhattacharya
Head of Department,	Emil Straube

May 2016

Major Subject: Mathematics

ABSTRACT

In this thesis, we suggest a new algorithm for solving convex optimization problems in Banach spaces. This algorithm is based on a greedy strategy, and it could be viewed as a nonlinear conjugate gradient type method. We prove its convergent rates under a suitable behavior of the modulus of uniform smoothness of the objective function. We apply the proposed algorithm on several examples such as approximation in Hilbert spaces, solving linear systems, and others. We also perform several numerical tests in the case when the objective function is the opposite of the log-likelihood function under the Logistic Regression model. Our numerical results confirm the fast convergence rate of the proposed algorithm and its potential for solving real life problems.

ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor, Dr. Guergana Petrova at Texas A&M University. It was her that taught me how to do research in Mathematics. Whenever I ran into a trouble spot or had a question about my research or writing, I could always get her help on time. She consistently steered me in the right direction whenever she thought I needed it.

I would also like to thank all professors who were involved in this research project: Dr. Peter Howard, Dr. Anirban Battacharya, and Dr. James Long. Without their kind participation and help, I would have not been able to finish proving lemmas and theorems, and gathering data for numerical tests.

Finally, I must express my very profound gratitude to my parents and to my grilfriend for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of doing research and writing this thesis. This accomplishment would not have been possible without them. I love you all.

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

The main goal in convex optimization is the development and analysis of algorithms for solving the problem

$$\inf_{x\in\Omega} E(x), \tag{1.0.1}$$

where E is a given convex function and Ω is a bounded convex subset of a Banach space X. E is called the *objective* function and satisfies the convexity condition

$$E(\gamma x + \delta y) \leq \gamma E(x) + \delta E(y), \quad x, y \in \Omega, \quad \gamma, \delta \geq 0, \quad \gamma + \delta = 1.$$

While classical convex optimization deals with objective functions E defined on subsets Ω in $\mathbb{R}^n$ for moderate values of n, see [2], some of the new applications require that the dimension n is quite large or even ∞. The design of algorithms for such cases is quite challenging since typical convergence results involve n, and therefore deteriorate severely with the growth of n. This is the so-called curse of dimensionality. Recently, there has been an increased interest, see [4, 9, 10, 13], in developing greedy based strategies for solving (1.0.1) with provable convergence rate depending only on the properties of E and not on the dimension of the underlying space. These algorithms provide approximations $\{E(x_m)\}$, $m = 1, 2, \ldots$ to the solution of (1.0.1), with x_m being a linear combination of m elements from a given dictionary $\mathcal{D} \subset X$. A dictionary is any set $\mathcal{D}$ of norm one elements from X whose span is dense in X. An example of a dictionary is any Schauder basis for X, or a union of several Schauder bases. The current greedy algorithms pick an initial approximation $E(0)$, a set Ω as

$$\Omega := \{x \in X \ : \ E(x) \leq E(0)\},$$

and generate a sequence of successive approximations $E_m := E(x_m)$, $m = 1, 2, \ldots$, recursively, using the dictionary $\mathcal{D}$. Some methods, such as the Weak Chebychev Greedy Algorithm, see [9], provide at Step m an approximant x_m to the point $\bar{x}$ at which E attains its global minimum, determined as

$$x_m := \text{argmin}_{x \in span\{\varphi_{j_1}, \ldots, \varphi_{j_m}\}} E(x),$$

where $\varphi_{j_1}, \ldots, \varphi_{j_m}$ are suitably chosen elements from $\mathcal{D}$. Others choose x_m as

$$x_m := \text{argmin}_{\omega, \lambda \in \mathbb{R}} E(\omega x_{m-1} + \lambda \varphi_m),$$

or

$$x_m := \text{argmin}_{\lambda \in [0,1]} E((1 - \lambda) x_{m-1} + \lambda \varphi_m),$$

for suitably chosen $\varphi_m \in \mathcal{D}$, where x_{m-1} is the previously generated point. Convergence rates for these algorithms are proved to be of order $\mathcal{O}(m^{1-q})$, where q is a parameter related to the smoothness of the objective function E. Note that the last two approaches are more computationally friendly, since they require solving two or one dimensional optimization problems at each step. However, note that some of these algorithms work only if the minimum of E is attained in the convex hull of $\mathcal{D}$, since the approximant x_m is derived as a convex combination of x_{m-1} and φ_m.

In this thesis, we introduce a new greedy algorithm for convex optimization based on one dimensional optimization at each step, which does not require the solution of (1.0.1) to belong to the convex hull of $\mathcal{D}$ and has a rate of convergence $\mathcal{O}(m^{1-q})$. This algorithm can be viewed also as a type of nonlinear conjugate gradient method. The main difference from standard conjugate gradient methods is that rather than

building the next approximation x_m, using the current approximation x_{m-1} as

$$x_m = x_{m-1} + \lambda_m \varphi_m,$$

where λ_m and φ_m are appropriately chosen, see [5], we choose

$$x_m = t_m(x_{m-1} + \lambda_m \varphi_m),$$

for a suitably chosen real number t_m. The presented technique is a generalization of the recently introduced Rescaled Pure Greedy Algorithm (**RPGA**) for approximating functions in Hilbert and Banach spaces, see [8]. We call it **RPGA(co)**.

The thesis is organized as follows. In chapter §2, we list several definitions and known results about convex functions. In chapter §3, we present the **RPGA(co)**, prove its convergence rate, and introduce its weak version. In chapter §4, we perform several numerical tests, and discuss the application of the proposed algorithm to particular choices of objective functions E.

2. PRELIMINARIES

In this chapter, we introduce some notation and state several known facts and definitions.

2.1 The Banach Space X

A Banach space X is a complete vector space with a norm $\|\cdot\|$. A set of functions $\mathcal{D} := \{\varphi\} \subset X$ is called a dictionary for X if $||\varphi|| = 1$ for every $\varphi \in \mathcal{D}$ and the closure of $span(\mathcal{D})$ is X. An example of a dictionary is any Schauder basis for X. However, the main idea behind dictionaries is to cover redundant families such as frames. A common example of dictionaries is the union of several Schauder bases. For a general dictionary $\mathcal{D} \subset X$, we define the class of elements

$$\mathcal{A}_1^o(\mathcal{D}, M) := \{x = \sum_{k\in\Lambda} c_k(x)\varphi_k : \varphi_k \in \mathcal{D}, |\Lambda| < \infty, \sum_{k\in\Lambda} |c_k(x)| \leq M\},$$

and by $\mathcal{A}_1(\mathcal{D}, M)$ its closure in X. Then, $\mathcal{A}_1(\mathcal{D})$ is defined to be the union of the classes $\mathcal{A}_1(\mathcal{D}, M)$ over all $M > 0$. For $x \in \mathcal{A}_1(\mathcal{D})$, we define the norm of x as

$$||x||_{\mathcal{A}_1(\mathcal{D})} := \inf\{M : x \in \mathcal{A}_1(\mathcal{D}, M)\}.$$

In what follows, we assume that the minimizer $\bar{x}$ of the objective function E is such that $\bar{x} \in \mathcal{A}_1(\mathcal{D})$. Our algorithm will generate approximants $x_k \in X$ to $\bar{x}$, where each x_k is a sum of at most k terms from the dictionary $\mathcal{D}$.

2.2 The Objective Function E

Let us first recall several definitions.

- A function E is Fréchet differentiable at $x \in \Omega$ if there exists a bounded linear

functional, denoted by $E'(x) \in X^*$, such that

$$\lim_{\|h\| \to 0} \frac{|E(x+h) - E(x) - \langle E'(x), h \rangle|}{\|h\|} = 0.$$

Here, we use the notation $\langle F, x \rangle := F(x)$ to denote the action of the functional $F \in X^*$ on the element $x \in X$.

- A function $E : X \to \mathbb{R}$ is called convex if $\forall x, y \in X$, and $\forall t \in [0, 1]$,

$$E(tx + (1-t)y) \leq tE(x) + (1-t)E(y).$$

The following lemmas are well known and we simply state them.

Lemma 2.2.1. *Let E be a Fréchet differentiable function at each point in Ω and convex on X. Then, for all $x \in \Omega$ and $x' \in X$,*

$$\langle E'(x), x - x' \rangle \geq E(x) - E(x').$$

Proof. Clearly the inequality holds for $x' = x$. Fix $x \in \Omega$, $x' \in X$, $x' \neq x$. It follows from the definition of Fréchet derivative that for $h = t(x' - x), t \in \mathbb{R}$,

$$\langle E'(x), x' - x \rangle = \lim_{t \to 0} \frac{E((1-t)x + tx') - E(x)}{t}.$$

For $0 < t < 1$, $E((1-t)x + tx') - E(x) \leq (1-t)E(x) + tE(x')$, since E is convex, and therefore

$$\langle E'(x), x' - x \rangle \leq E(x') - E(x),$$

which completes the proof.

□

Lemma 2.2.2. *Let E be a Fréchet differentiable convex function, defined on a convex domain Ω. Then E has a global minimum at $\bar{x} \in \Omega$ if and only if $E'(\bar{x}) = 0$.*

Lemma 2.2.3. *Let F be a Fréchet differentiable function, $\varphi \in X$ be a fixed element in X, and x^* be such that $x^* = \text{argmin}\{F(x) : x = t\varphi, t \in I\!R\}$. Then, $\langle F'(x^*), x^* \rangle = 0$.*

In this thesis, we consider objective functions E that satisfy the following two assumptions.

- **Condition 0:** E has Fréchet derivative $E'(x) \in X^*$ at each point in $\Omega := \{x \in X \ : \ E(x) \leq E(0)\}$, Ω is bounded, and$\|E'(x)\| \leq M_0$, for $x \in \Omega$.
- **Uniform Smoothness (US):** There are constants $0 < \alpha$, $M > 0$, and $1 < q \leq 2$, such that for all x, x' with $\|x - x'\| \leq M$, $x \in \Omega$, $x' \in X$,

$$E(x') - E(x) - \langle E'(x), x' - x \rangle \leq \alpha \|x' - x\|^q.$$

The **US** condition on E is closely related to a condition on the modulus of smoothness ρ of E. We refer the reader to [7], where these relations are discussed.

Next, we point out that when looking for the global minimizer $\bar{x}$ of E, we can restrict ourselves to the set

$$\Omega := \{x : E(x) \leq E(0)\},$$

since $\bar{x} \in \Omega$. In what follows, we will consider the minimization problem (1.0.1) over this set. Note that this is a convex set as a level set of a convex function.

Further in this thesis, we will use the folowing lemma, proved in [7]. Other versions of this lemma have been discussed in [11].

Lemma 2.2.4. *Let $\ell > 0$, $r > 0$, $B > 0$, and $\{a_m\}_{m=1}^{\infty}$ and $\{r_m\}_{m=2}^{\infty}$ be finite or infinite sequences of non-negative numbers satisfying the inequalities*

$$a_J \leq B, \quad a_m \leq a_{m-1}(1 - \frac{r_m}{r} a_{m-1}^{\ell}), \quad m = J+1, J+2, \ldots.$$

Then, we have

$$a_m \leq \max\{1, \ell^{-1/\ell}\} r^{1/\ell} (rB^{-\ell} + \Sigma_{k=J+1}^{m} r_k)^{-1/\ell}, \quad m = J+1, J+2, \ldots.$$

3. RESCALED PURE GREEDY STRATEGIES FOR CONVEX OPTIMIZATION

3.1 Rescaled Pure Greedy Algorithm (**RPGA(co)**)

In this section, we describe our new algorithm for finding the minimum of a convex function with parameter μ and a dictionary $\mathcal{D}$.

RPGA(co)$(\mu, \mathcal{D})$:

- **Step** 0: Define $x_0 = 0$. If $E'(x_0) = 0$, stop the algorithm and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assuming x_{m-1} has been defined and $E'(x_{m-1}) \neq 0$.

  – Choose a direction $\varphi_{j_m} \in \mathcal{D}$ such that

  $$|\langle E'(x_{m-1}), \varphi_{j_m}\rangle| = \sup_{\varphi\in\mathcal{D}} |\langle E'(x_{m-1}), \varphi\rangle|.$$

  – With

  $$\lambda_m := \operatorname{sgn}\{\langle E'(x_{m-1}), \varphi_{j_m}\rangle\} (\alpha\mu)^{-\frac{1}{q-1}} |\langle E'(x_{m-1}), \varphi_{j_m}\rangle|^{\frac{1}{q-1}},$$

  compute $\hat{x}_m := x_{m-1} - \lambda_m\varphi_{j_m}$, $t_m := \operatorname{argmin}_{t\in\mathbb{R}} E(t\hat{x}_m)$

  – Define the next point to be

  $$x_m = t_m\hat{x}_m.$$

- If $E'(x_m) = 0$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.

- If $E'(x_m) \neq 0$, proceed to Step $m+1$.

Let us observe that, because of Lemma 2.2.2, if $E'(x_m) = 0$ at Step m, the output x_m of the algorithm is the minimizer $\bar{x}$. Note that the algorithm requires a minimization of the objective function along the one dimensional space $span\{\hat{x}_m\}$. This univariate optimization problem is called line search and is well studied in optimization theory, see [6]. If at Step m we were to use $\hat{x}_m$ as next approximant and not x_m, which is the minimizer of E along the line generated by $\hat{x}_m$, then the algorithm would be very similar to the **EGA**($\mathcal{C}$) from [9] . The author there proves a convergence rate of $\mathcal{O}(m^{-r})$, for any $r \in (0, \frac{q-1}{q+1})$ under suitable conditions on the parameters. Note that our algorithm, which simply adds a one dimensional optimization at each step, makes it possible to achieve an optimal convergence rate of $\mathcal{O}(m^{1-q})$. Also, in contrast to the other greedy algorithms from [9] that rely on one dimensional minimization at each step, this algorithm provides convergent results for all $\bar{x}$, and not only for $\bar{x}$ in the convex hull of the dictionary $\mathcal{D}$.

Notice that all outputs $\{x_k\}_{k=1}^{\infty}$ generated by the **RPGA(co)**$(\mu, \mathcal{D})$ are in Ω, since $E(x_k) \leq E(0)$. The following theorem is our main convergence result.

Theorem 3.1.1. *Let the convex function E satisfy* **Condition 0** *and the* **US** *condition, and the minimizer $\bar{x} \in \mathcal{A}_1(\mathcal{D})$. Then, the* **RPGA(co)***$(\mu, \mathcal{D})$ with parameter μ, $\mu > \max\{1, \alpha^{-1} M_0 M^{1-q}\}$, applied to E and a dictionary $\mathcal{D} = \{\varphi\}$ outputs the sequence $\{x_k\}_{k=0}^{\infty}$, where the error $e_k := E(x_k) - E(\bar{x})$ satisfies the inequality*

$$e_k \leq C_1 k^{1-q}, \quad k \geq 2,$$

with $C_1 = C_1(q, \alpha, E, \mu)$.

Proof. If at Step k_0 the **RPGA(co)**$(\mu, \mathcal{D})$ had stopped, this means that we had

recovered the point of minimum $\bar{x}$, namely that $\bar{x} = x_{k_0}$. Since we set $x_k = x_{k_0} = \bar{x}$, for $k > k_0$, then the error $e_k = 0$ for $k \geq k_0$. For values of $k < k_0$, or when the algorithm had not stopped, we have the following. We consider Step k, $k = 1, 2, 3, \ldots$ of the algorithm. The definition of λ_k and the choice of parameter μ assures that

$$\|(x_{k-1} - \lambda_k \varphi_{j_k}) - x_{k-1}\| = \left(\frac{|\langle E'(x_{k-1}), \varphi_{j_k} \rangle|}{\alpha \mu} \right)^{\frac{1}{q-1}} \leq M,$$

and therefore, applying the **US** condition to $(x_{k-1} - \lambda_k \varphi_{j_k})$ and x_{k-1} gives

$$\begin{aligned} E(\hat{x}_k) &= E(x_{k-1} - \lambda_k \varphi_{j_k}) \leq E(x_{k-1}) - \lambda_k \langle E'(x_{k-1}), \varphi_{j_k} \rangle + \alpha |\lambda_k|^q \\ &= E(x_{k-1}) - \frac{\mu - 1}{\mu} (\alpha \mu)^{-\frac{1}{q-1}} |\langle E'(x_{k-1}), \varphi_{j_k} \rangle|^{\frac{q}{q-1}}, \end{aligned}$$

where we use the fact that $\|\varphi_{j_k}\| = 1$. Since $E(x_k) \leq E(\hat{x}_k)$, we derive that

$$E(x_k) \leq E(x_{k-1}) - \frac{\mu - 1}{\mu} (\alpha \mu)^{-\frac{1}{q-1}} |\langle E'(x_{k-1}), \varphi_{j_k} \rangle|^{\frac{q}{q-1}}. \tag{3.1.1}$$

In particular, $E(x_k) \leq E(x_{k-1})$, and therefore, by induction

$$E(x_k) \leq E(x_0) = E(0),$$

which means that the generated approximations $x_k \in \Omega$. Next, we provide a lower bound for $|\langle E'(x_{k-1}), \varphi_{j_k} \rangle|$. Let us fix $\varepsilon > 0$ and choose a representation for $\bar{x} = \sum_{\varphi \in \mathcal{D}} c_\varphi^\varepsilon \varphi$, such that

$$\sum_{\varphi \in \mathcal{D}} |c_\varphi^\varepsilon| < \|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})} + \varepsilon.$$

Since $\langle E'(x_{k-1}), x_{k-1} \rangle = 0$, because of the choice of x_{k-1} and Lemma 2.2.3, we have

that

$$\begin{aligned}\langle E'(x_{k-1}), x_{k-1} - \bar{x}\rangle &= -\langle E'(x_{k-1}), \bar{x}\rangle = -\sum_{\varphi} c_{\varphi}^{\varepsilon}\langle E'(x_{k-1}), \varphi\rangle \\ &\leq |\langle E'(x_{k-1}), \varphi_{j_k}\rangle| \sum_{\varphi} |c_{\varphi}^{\varepsilon}| \\ &< |\langle E'(x_{k-1}), \varphi_{j_k}\rangle(\|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})} + \varepsilon),\end{aligned}$$

where we have used the choice of φ_{j_k}. We let $\varepsilon \to 0$ and obtain the inequality

$$\langle E'(x_{k-1}), x_{k-1} - \bar{x}\rangle \leq |\langle E'(x_{k-1}), \varphi_{j_k}\rangle| \|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})}.$$

It follows from Lemma 2.2.1, with $x = x_{k-1}$ and $x' = \bar{x}$ and the above inequality that

$$\|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})}^{-1} e_{k-1} \leq |\langle E'(x_{k-1}), \varphi_{j_k}\rangle|, \tag{3.1.2}$$

which is the desired estimate from below for $|\langle E'(x_{k-1}), \varphi_{j_k}\rangle|$ In particular,

$$E(x_k) \leq E(x_{k-1}) - \frac{\mu - 1}{\mu} (\alpha\mu)^{-\frac{1}{q-1}} \|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})}^{-\frac{q}{q-1}} e_{k-1}^{\frac{q}{q-1}}. \tag{3.1.3}$$

Subtracting $E(\bar{x})$ from both sides gives

$$e_k \leq e_{k-1}\left(1 - \frac{\mu - 1}{\mu} (\alpha\mu)^{-\frac{1}{q-1}} \|\bar{x}\|_{\mathcal{A}_1(\mathcal{D})}^{-\frac{q}{q-1}} e_{k-1}^{\frac{1}{q-1}}\right).$$

Now we apply Lemma 2.2.4 for the sequence $\{a_k\}$, with

$$a_k := \frac{e_k}{\alpha\mu||\bar{x}||_{\mathcal{A}_1(\mathcal{D})}},$$

and

$$r_k = 1, \quad \ell = \frac{1}{q-1} > 0, \quad B = \frac{||E'(0)||}{\alpha\mu}, \quad r = \frac{\mu}{\mu-1}||\bar{x}||_{\mathcal{A}_1(\mathcal{D})},$$

and derive that

$$e_k \leq \frac{\alpha\mu^q||\bar{x}||^q_{\mathcal{A}_1(\mathcal{D})}}{(\mu-1)^{q-1}}\left(k-1+\frac{\mu}{\mu-1}||\bar{x}||_{\mathcal{A}_1(\mathcal{D})}\left(\frac{\alpha\mu}{||E'(0)||}\right)^{\frac{1}{q-1}}\right)^{1-q},$$

and the proof is completed.

□

Notice that we can optimize with respect to the parameter μ and select a specific value for $\mu > \max\{1, \alpha^{-1}M_0M^{1-q}\}$ that will guarantee the best convergence rate in terms of best constants.

Careful analysis of the above proof shows that a similar theorem holds in the following case.

Theorem 3.1.2. *Let the convex function E be Frechet differentiable and there are constants $0 < \alpha$ and $1 < q \leq 2$, such that for all $x, x' \in X$,*

$$E(x') - E(x) - \langle E'(x), x' - x\rangle \leq \alpha||x' - x||^q.$$

Let the minimizer $\bar{x}$ of E be such that $\bar{x} \in \mathcal{A}_1(\mathcal{D})$. Then, the application of ***RPGA(co)****$(\mu, \mathcal{D})$ with parameter $\mu > 1$ and a dictionary $\mathcal{D} = \{\varphi\}$ outputs a sequence $\{x_k\}_{k=0}^{\infty}$, such that the error $e_k := E(x_k) - E(\bar{x})$ satisfies the inequality*

$$e_k \leq C_1 k^{1-q},$$

for $k \geq 1$ with $C_1 = C_1(q, \alpha, E, \mu)$.

3.2 The Weak Rescaled Pure Greedy Algorithm (**WRPGA(co)**)

In this section, we describe the weak version of our algorithm with weakness sequence $\{\ell_k\}$, $\ell_k \in (0,1]$ $k = 1, 2, \ldots$, and parameter sequence $\{\mu_k\}$, where $\mu_k > \max\{1, \alpha^{-1} M_0 M^{1-q}\}$, $k = 1, 2, \ldots$. In the case when $\ell_k = 1$ and $\mu_k = \mu$, $k = 1, 2, \ldots$, the **WRPGA(co)**$(\{\ell_k\}, \{\mu_k\}, \mathcal{D})$ is the **RPGA(co)**$(\mu, \mathcal{D})$. The weakness sequence allows us to have some freedom in the selection of the next direction φ_{j_k}, while the parameter sequence $\{\mu_k\}$ gives more choices in how much to advance along the selected direction φ_{j_k}.

WRPGA(co)$(\{\ell_k\}, \{\mu_k\}, \mathcal{D})$:

- **Step** 0: Define $x_0 = 0$. If $E'(x_0) = 0$, stop the algorithm and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assuming x_{m-1} has been defined and $E'(x_{m-1}) \neq 0$. Choose a direction $\varphi_{j_m} \in \mathcal{D}$ such that

$$|\langle E'(x_{m-1}), \varphi_{j_m} \rangle| \geq \ell_m \sup_{\varphi \in \mathcal{D}} |\langle E'(x_{m-1}), \varphi \rangle|.$$

  With $\hat{x}_m := x_{m-1} - \lambda_m \varphi_{j_m}$, where

$$\lambda_m := \operatorname{sgn}\{\langle E'(x_{m-1}), \varphi_{j_m} \rangle\} \, (\alpha \mu_m)^{-\frac{1}{q-1}} \, |\langle E'(x_{m-1}), \varphi_{j_m} \rangle|^{\frac{1}{q-1}},$$

$$t_m := \operatorname{argmin}_{t \in \mathbb{R}} E(t \hat{x}_m),$$

  define the next point to be

$$x_m = t_m \hat{x}_m.$$

- If $E'(x_m) = 0$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.

- If $E'(x_m) \neq 0$, proceed to Step $m+1$.

The next theorem is the main result about the convergence rate of the **WRPGA(co)**$(\{\ell_k\}, \{\mu_k\}, \mathcal{D})$.

Theorem 3.2.1. *Let the convex function E satisfy* **Condition 0** *and the* **US** *condition, and its minimizer $\bar{x} \in \mathcal{A}_1(\mathcal{D})$. Then, the application of the* **WRPGA(co)***$(\{\ell_k\}, \{\mu_k\}, \mathcal{D})$ with a weakness sequence $\{\ell_k\}$, parameter sequence $\{\mu_k\}$, $\mu_k > \max\{1, \alpha^{-1} M_0 M^{1-q}\}$, and a dictionary $\mathcal{D} = \{\varphi\}$ outputs the sequence $\{x_k\}_{k=0}^{\infty}$, such that the following inequality holds*

$$e_k := E(x_k) - e(\bar{x}) \leq \alpha \|\bar{x}\|^q_{\mathcal{A}_1(\mathcal{D})} \left(C_1 + \sum_{j=2}^{k} (\mu_j - 1) \left(\frac{\ell_j}{\mu_j} \right)^{\frac{q}{q-1}} \right)^{1-q}, \quad k \geq 1,$$

with $C_1 = C_1(q, \alpha, E)$.

Proof. Similarly to the proof of Theorem 3.1.1, we have for $k \geq 2$,

$$E(x_k) \leq E(x_{k-1}) - \frac{\mu_k - 1}{\mu_k} (\alpha \mu_k)^{-\frac{1}{q-1}} |\langle E'(x_{k-1}), \varphi_{j_k} \rangle|^{\frac{q}{q-1}}. \tag{3.2.4}$$

The same way one can easily derive the lower estimate

$$\|\bar{x}\|^{-1}_{\mathcal{A}_1(\mathcal{D})} \ell_k e_{k-1} \leq |\langle E'(x_{k-1}), \varphi_{j_k} \rangle|,$$

We use (3.2.4) and the lower estimate for $|\langle E'(x_{k-1}), \varphi_{j_k} \rangle|$ to obtain

$$e_k \leq e_{k-1} \left(1 - \frac{\mu_k - 1}{\mu_k} (\alpha \mu_k)^{-\frac{1}{q-1}} \ell_k^{\frac{q}{q-1}} \|\bar{x}\|^{-\frac{q}{q-1}}_{\mathcal{A}_1(\mathcal{D})} e_{k-1}^{\frac{1}{q-1}} \right).$$

It follows from the monotonicity that $e_1 \leq e_0 = E(0) - E(\bar{x})$. Now we apply

Lemma 2.2.4 for the sequence of errors $\{e_k\}_{k=1}^{\infty}$ and

$$r_k = (\mu_k - 1)\left(\frac{\ell_k}{\mu_k}\right)^{\frac{q}{q-1}}, \quad \ell = \frac{1}{q-1} > 0, \quad B = E(0) - E(\bar{x}), \quad r = \left(\alpha \|\bar{x}\|^q_{\mathcal{A}_1(\mathcal{D})}\right)^{\frac{1}{q-1}},$$

and derive that

$$e_k \le \alpha \|\bar{x}\|^q_{\mathcal{A}_1(\mathcal{D})} \left(\left(\frac{\alpha \|\bar{x}\|^q_{\mathcal{A}_1(\mathcal{D})}}{E(0) - E(\bar{x})} \right)^{\frac{1}{q-1}} + \sum_{j=2}^{k} (\mu_j - 1) \left(\frac{\ell_j}{\mu_j} \right)^{\frac{q}{q-1}} \right)^{1-q}.$$

The proof is completed. □

Theorem 3.2.2. *Let the convex function E be Frechet differentiable and there are constants $0 < \alpha$ and $1 < q \le 2$, such that for all $x, x' \in X$,*

$$E(x') - E(x) - \langle E'(x), x' - x \rangle \le \alpha ||x' - x||^q.$$

Let the minimizer $\bar{x}$ of E be in $\mathcal{A}_1(\mathcal{D})$. Then, the application of the **WRPGA(co)** *$(\{\ell_k\}, \{\mu_k\}, \mathcal{D})$ with a weakness sequence $\{\ell_k\}$ and a parameter sequence $\{\mu_k\}$, $\mu_k > 1$, and a dictionary $\mathcal{D} = \{\varphi\}$ outputs a sequence $\{x_k\}_{k=0}^{\infty}$, such that the error $e_k := E(x_k) - E(\bar{x})$ satisfies the inequality*

$$e_k \le \alpha \|\bar{x}\|^q_{\mathcal{A}_1(\mathcal{D})} \left(C_1 + \sum_{j=2}^{k} (\mu_j - 1) \left(\frac{\ell_j}{\mu_j} \right)^{\frac{q}{q-1}} \right)^{1-q},$$

for $k \ge 1$ with $C_1 = C_1(q, \alpha, E)$.

4. EXAMPLES & NUMERICAL EXPERIMENTS

In this chapter, we will apply the **RPGA(co)**$(\mu, \mathcal{D})$ to several convex objective functions E.

4.1 Best Approximation in Hilbert Space

Let us consider the case when X is a Hilbert space H with a norm, induced by scalar product, namely $||\cdot|| = (\cdot,\cdot)^{1/2}$ and an objective function $E : H \to \mathbb{R}$, defined as

$$E(x) := ||x - \bar{x}||^2, \tag{4.1.1}$$

where $\bar{x} \in H$ is a fixed element in the Hilbert space H. Since $||x - \bar{x}||$ is convex because of the properties of a norm, E is a convex function as a composition of the increasing convex function $g(t) = t^2, t \geq 0$ and $||x - \bar{x}||$. It is easy to compute that its Fréchet derivative at $x \in H$ is the linear functional $E'(x)$, which acts on $h \in H$ as

$$\langle E'(x), h\rangle = 2(x - \bar{x}, h).$$

We have that

$$\begin{aligned} E(x') - E(x) - \langle E'(x), x' - x\rangle &= ||x' - \bar{x}||^2 - ||x - \bar{x}||^2 - 2(x - \bar{x}, x' - x) \\ &= ||x - x'||^2, \end{aligned}$$

and therefore the objective function E satisfies the conditions of Theorem 3.1.2 with $\alpha = 1$ and $q = 2$. We call the **RPGA(co)**$(\mu, \mathcal{D})$ in this case the **RPGA**$(\mu, \mathcal{D})$ with parameter $\mu > 1$ and a dictionary $\mathcal{D}$. The algorithm is the following.

RPGA$(\mu, \mathcal{D})$:

- **Step** 0: Define $x_0 = 0$. If $x_0 = \bar{x}$, stop the algorithm and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assuming x_{m-1} has been defined and $x_{m-1} \neq \bar{x}$.

  – Choose $\varphi_{j_m} \in \mathcal{D}$ such that

  $$| (x_{m-1} - \bar{x}, \varphi_{j_m}) | = \sup_{\varphi \in \mathcal{D}} | (x_{m-1} - \bar{x}, \varphi) |.$$

  – Compute

  $$\lambda_m := \frac{2}{\mu}(x_{m-1} - \bar{x}, \varphi_{jm}), \quad \hat{x}_m := x_{m-1} - \lambda_m \varphi_{j_m}, \quad s_m := \frac{(\hat{x}_m, \bar{x})}{||\bar{x}||^2},$$

  – Define

  $$x_m = s_m \hat{x}_m.$$

- If $x_m = \bar{x}$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.

- If $x_m \neq \bar{x}$, proceed to Step $m + 1$.

The next theorem is a direct consequence of Theorem 3.1.2 for the function $E(x) := ||x - \bar{x}||^2$.

Theorem 4.1.1. *If $\bar{x} \in \mathcal{A}_1(\mathcal{D}) \subset H$, then the **RPGA**$(\mu, \mathcal{D})$ with parameter $\mu > 1$ and a dictionary $\mathcal{D} = \{\varphi\}$ outputs the sequence $\{x_k\}_{k=0}^{\infty}$, where x_k is a sum of k terms from the dictionary, such that*

$$||x_k - \bar{x}|| \leq C_1 k^{-1/2}, k \geq 1,$$

with $C_1 = C_1(\mu, \bar{x})$.

When $\mu = 2$, the above algorithm is exactly the $\mathbf{RPGA}(\mathcal{D})$, presented in [8]. The latter was suggested as a modification to the Pure Greedy Algorithm, known also as Matching Pursuit, and an alternative to the Relaxed Greedy Algorithm and the Orthogonal Greedy Algorithm for Hilbert spaces. The theorem in [8] for the convergence rate of this algorithm, see Theorem 3.1 from the latter paper, is actually an improved version of Theorem 4.1.1. The improvement is in the constants since the analysis of the algorithm uses the formula of the objective function rather than some of its properties. We observe here that the $\mathbf{RPGA}(\mathcal{D})$ can be modified so that it depends on a parameter $\mu > 1$, see $\mathbf{RPGA}(\mu, \mathcal{D})$. Similar arguments as in the proof of Theorem 3.1 from [8], with little modifications that account for the parameter μ give the following improved version of Theorem 4.1.1.

Theorem 4.1.2. *Let $\bar{x} \in \mathcal{A}_1(\mathcal{D}) \subset H$. The $\boldsymbol{RPGA}(\mu, \mathcal{D})$, with parameter $\mu > 1$ and a dictionary $\mathcal{D}$ outputs a sequence $\{x_k\}_{k \geq 0}$ of approximations to $\bar{x}$ satisfying the following error estimate*

$$||\bar{x} - x_k|| \leq \frac{\mu}{2\sqrt{\mu - 1}} ||\bar{x}||_{\mathcal{A}_1(\mathcal{D})} k^{-1/2}, \quad k = 1, 2 \dots \tag{4.1.2}$$

Proof. The proof follows the arguments from Theorem 3.1 in [8]. The presence of μ is accounted for in the estimate

$$||\bar{x} - x_k||^2 \leq ||\bar{x} - x_{k-1}||^2 \left(1 - \frac{4(\mu - 1)}{\mu^2} ||\bar{x}||^{-2}_{\mathcal{A}_1(\mathcal{D})} ||\bar{x} - x_{k-1}||^2\right).$$

Application of Lemma 2.2.4 with $a_k = ||\bar{x} - x_k||^2$, $B = ||\bar{x}||^2_{\mathcal{A}_1(\mathcal{D})}$, $r_k := \frac{4(\mu-1)}{\mu^2}$, $r = ||\bar{x}||^2_{\mathcal{A}_1(\mathcal{D})}$, and $\ell = 1$ gives

$$||\bar{x} - x_k|| \leq ||\bar{x}||^2_{\mathcal{A}_1(\mathcal{D})} \left(1 + \frac{4(\mu-1)}{\mu^2}(k-1)\right) \leq ||\bar{x}||^2_{\mathcal{A}_1(\mathcal{D})} \frac{\mu^2}{4(\mu-1)} k^{-1},$$

and the proof is completed. Since the constant $\frac{\mu}{2\sqrt{\mu-1}} \geq 1$, and its minimum value 1 is achieved when $\mu = 2$, the best constant in the above estimate is achieved when the parameter $\mu = 2$.

□

4.2 The Space $\mathbb{R}^n$

In this section, we discuss the case when the Banach space $X = \mathbb{R}^n$, the norm $||\cdot||$ is the Euclidean norm, induced by the standard scalar product $||x||^2 = (x, x), x \in \mathbb{R}^n$. In this case, the dictionary $\mathcal{D}$ is any system $\{\varphi\}$ of vectors in $\mathbb{R}^n$(basis or not), such that $span\{\varphi\} = \mathbb{R}^n$. For any $x \in \mathbb{R}^n$, we define the quantity

$$||x||_{\mathcal{A}_1(\mathcal{D})} = \inf\{M : x = \sum_{\varphi\in\mathcal{D}} x_\varphi \varphi, \sum_{\varphi\in\mathcal{D}} |x_\varphi| \leq M\}.$$

When $\mathcal{D}$ is a basis, $||x||_{\mathcal{A}_1(\mathcal{D})}$ is just the ℓ_1 norm of x with respect to this basis.

4.2.1 Objective Functions of n Variables

If the objective function E is a sufficiently smooth convex function, defined on $\mathbb{R}^n$, then its Fréchet derivative $E'(x)$ at x is the linear functional $E'(x)$ which acts on $h \in \mathbb{R}^n$ as

$$\langle E'(x), h\rangle = \sum_{j=1}^{n} \frac{\partial E}{\partial x_j}(x) h_j = (\nabla E(x), h), \quad h = (h_1, \ldots, h_n).$$

In this case, we denote our algorithm as $\mathbf{RPGA}(\mathbb{R}^n)(\mu, \mathcal{D})$, and present it below.

- **Step** 0: Define $x_0 = 0$. If $\nabla E(x_0) = 0$, stop the algorithm and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assuming x_{m-1} has been defined and $\nabla E(x_{m-1}) \neq 0$.

  – Choose a direction $\varphi_{j_m} \in \mathcal{D}$ such that

  $$|(\nabla E(x_{m-1}), \varphi_{j_m})| = \sup_{\varphi \in \mathcal{D}} |(\nabla E(x_{m-1}), \varphi)|.$$

  – With

  $$\lambda_m := \operatorname{sgn}\{(\nabla E(x_{m-1}), \varphi_{j_m})\} \, (\alpha\mu)^{-\frac{1}{q-1}} \, |(\nabla E(x_{m-1}), \varphi_{j_m})|^{\frac{1}{q-1}},$$

  compute $\hat{x}_m := x_{m-1} - \lambda_m \varphi_{j_m}, \quad t_m := \operatorname{argmin}_{t \in \mathbb{R}} E(t\hat{x}_m)$.

  – Define the next point to be

  $$x_m = t_m \hat{x}_m.$$

- If $\nabla E(x_m) = 0$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.

- If $\nabla E(x_m) \neq 0$, proceed to Step $m + 1$.

Depending on the properties of the objective function E, versions of Theorem 3.1.1 and Theorem 3.1.2 hold for the $\mathbf{RPGA}(\mathbb{R}^n)(\mu, \mathcal{D})$, applied to E.

4.2.2 Logistic Regression

In this section, we consider the log-likelihood function under the LR model, as our objective function. To be precise, we consider the opposite of this function. We show that it satisfies the conditions of Theorem 3.1.2, and therefore we can apply the **RPGA(co)** to find its minimum.

4.2.2.1 Background

Regression analysis is used to find relationships between data sets. Logistic regression is a regression technique naturally suited to data related to binary outcomes.

Let $(A, \mathbf{b})$ be a dataset with binary outcomes. For each experiment $\mathbf{a}_i$ in A, the outcome is either $b_i = 1$ or $b_i = 0$. Experiments with outcome $b_i = 1$ are said to belong to a *positive class*, while experiments with $b_i = 0$ belong to a *negative class*. We wish to create a regression model which allows classification of an experiment $\mathbf{a}$ as positive or negative, that is, belonging to either the positive or negative class. Though Logistic Regression is applicable to datasets with outcomes in $[0, 1]$, we will restrict our discussion to the binary case.

We can think of an experiment $\mathbf{a}_i$ in A as a Bernoulli trial with mean parameter $\mu(\mathbf{a}_i)$. Thus, b_i is a Bernoulli random variable with mean $\mu(\mathbf{a}_i)$ and variance $\mu(\mathbf{a}_i)(1-\mu(\mathbf{a}_i))$. It is important to note that the variance of b_i depends on the mean, and hence on the experiment $\mathbf{a}_i$. To model the relation between each experiment $\mathbf{a}_i$ and the expected value of its outcome, we will use the logistic function. This function is written as

$$\mu(\mathbf{a}, x) = \frac{\exp(x^T \mathbf{a})}{1 + \exp(x^T \mathbf{a})}, \quad x \in \mathbb{R}^m, \tag{4.2.3}$$

We may interpret the expectation function as the probability that b_i=1, or equiv-

alently, that $\mathbf{a}_i$ belongs to the positive class. Thus, we may compute the probability of the i-th experiment and outcome in the dataset $(A, \mathbf{b})$ as

$$P(\mathbf{a}_i, b_i|x) = \mu(\mathbf{a}_i, x)^{b_i}(1 - \mu(\mathbf{a}_i, x))^{1-b_i}$$

where $b_i \in \{0, 1\}$, $i = 1, 2, \ldots, N$. From this expression, we derive the likelihoood and log-likelihood of the data $(A, \mathbf{b})$ under the LR model with parameters x as

$$L(A, \mathbf{b}, x) = \prod_{i=1}^{N} \mu(\mathbf{a}_i, x)^{b_i}(1 - \mu(\mathbf{a}_i, x))^{1-b_i},$$

and

$$\ln L(A, \mathbf{b}, x) = \sum_{i=1}^{N} \left(b_i \ln(\mu(\mathbf{a}_i, x)) + (1 - b_i) \ln(1 - \mu(\mathbf{a}_i, x))\right).$$

Our goal is to find the Maximum Likelihood Estimator(MLE) of the parameter x.

4.2.2.2 Objective Function

We consider the objective function $E : \mathbb{R}^m \to \mathbb{R}$, defined as

$$E(x) = E(A, \mathbf{b}, x) := -\sum_{i=1}^{N} \left[b_i \ln(\mu(\mathbf{a}_i, x)) + (1 - b_i) \ln(1 - \mu(\mathbf{a}_i, x))\right], \qquad (4.2.4)$$

where $\mathbf{A}$ is an $m \times N$ matrix with i-th column $\mathbf{a}_i$, that is, $A = (\mathbf{a}_1, \ldots, \mathbf{a}_N)$, and $b_i \in \{0, 1\}, i = 1, \ldots, N$. The gradient of $E(x)$ is:

$$\nabla E(x) = -\sum_{i=1}^{N} \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x)),$$

or more precisely,

$$\frac{\partial}{\partial x_j}E(x) = -\sum_{i=1}^{N} a_{ij}(b_i - \mu(\mathbf{a}_i, x)).$$

Next, we obtain that

$$\frac{\partial^2}{\partial x_j \partial x_k}E(x) = \sum_{i=1}^{N} \mu(\mathbf{a}_i, x)(1 - \mu(\mathbf{a}_i, x))a_{ij}a_{ik} = \left(\nabla^2 E(x)\right)_{jk}. \tag{4.2.5}$$

The following lemma shows that E is a convex function on $\mathbb{R}^m$ and satisfies the conditions in Theorem 3.1.2 with $\alpha = \frac{1}{2}\|A\|_2^2$ and $q = 2$.

Lemma 4.2.1. *The function E defined in (4.2.4) is convex. Moreover, for all $x, x' \in \mathbb{R}^m$, we have*

$$E(x) - E(x') - (\nabla E(x), x' - x) \leq \frac{1}{2}||A||_2^2 \cdot ||x' - x||^2. \tag{4.2.6}$$

Proof. Let us fix $x \in \mathbb{R}^m$ and denote by $D = D(x) := \nabla^2 E(x)$, with elements d_{jk}, $j, k = 1, 2, \ldots, m$. Then it follows from (4.2.5) that

$$d_{jk} = \sum_{i=1}^{N} \left[\sqrt{\mu(\mathbf{a}_i, x)(1 - \mu(\mathbf{a}_i, x))}a_{ij}\right]\left[\sqrt{\mu(\mathbf{a}_i, x)(1 - \mu(\mathbf{a}_i, x))}a_{ik}\right], \tag{4.2.7}$$

where we have used the fact that $0 < \mu(\mathbf{a}_i, x) < 1$. Let us denote by $\tilde{D}$ the $N \times m$ matrix with elements
$\tilde{d}_{ij} := \sqrt{\mu(\mathbf{a}_i, x)(1 - \mu(\mathbf{a}_i, x))}a_{ij}$. Then, (4.2.5) can be written as

$$d_{jk} = \sum_{i=1}^{N} \tilde{d}_{ij}\tilde{d}_{ik},$$

which is $D = \tilde{D}^T\tilde{D}$. Therefore, for all $u \in \mathbb{R}^m$, $u^T D u = u^T\tilde{D}^T\tilde{D}u = ||\tilde{D}u||^2 \geq 0$. Hence, D is a positive semi-definite matrix, which proves that E is convex. Next, we

will prove (4.2.6). Since E is smooth, we apply Taylor's theorem and obtain

$$E(x') = E(x) + \frac{\nabla^T E(x)(x'-x)}{1!} + \frac{(x'-x)^T D(\xi)(x'-x)}{2!},$$

where $\xi \in \{tx + (1-t)x' : t \in [0,1]\}$. Since $\sqrt{\mu(\mathbf{a}_i, x)(1-\mu(\mathbf{a}_i, x))} < 1$, we have that

$$\begin{aligned} E(x) - E(x') - (\nabla E(x), x'-x) &= \frac{(x'-x)^T D(\xi)(x'-x)}{2!} \\ &= \frac{1}{2}||\tilde{D}(\xi)(x'-x)||_2^2 \\ &\leq \frac{1}{2}||\tilde{D}(\xi)||_2^2 \cdot ||x'-x||^2 \\ &\leq \frac{1}{2}||A||_2^2 \cdot ||x'-x||^2 \\ &= \frac{1}{2}\lambda_{\max}(A^T A) \cdot ||x'-x||^2, \end{aligned}$$

and the proof is completed.

□

Next, we write the $\mathbf{RPGA}(\mathbb{R}^m)(\mu, \mathcal{D})$ for E, with $\mu = 2$. We call it $\mathbf{RPGA(LR)}(\mathcal{D})$.

RPGA(LR)($\mathcal{D}$):

- **Step** 0: Define $x_0 = 0$. If $\sum_{i=1}^N \mathbf{a}_i(b_i - \frac{1}{2}) = 0$, stop the algorithm and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assume x_{m-1} has been defined and $\sum_{i=1}^N \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_{m-1})) \neq 0$.

  –Choose a direction $\varphi_{j_m} \in \mathcal{D}$ such that

  $$|(\sum_{i=1}^N \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_{m-1})), \varphi_{j_m})| = \sup_{\varphi \in \mathcal{D}} |(\sum_{i=1}^N \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_{m-1})), \varphi)|.$$

  – With

$$\lambda_m := \text{sgn}\{\left(\sum_{i=1}^{N} \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_{m-1})), \varphi_{j_m}\right)\} \frac{|(\sum_{i=1}^{N} \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_{m-1})), \varphi_{j_m})|}{||A||_2^2},$$

compute $\hat{x}_m := x_{m-1} - \lambda_m \varphi_{j_m}$, where t_m satisfies $t_m = \text{argmin}_{t\in\mathbb{R}} E(t\hat{x}_m)$.

–Define the next point to be

$$x_m = t_m \hat{x}_m.$$

- If $\sum_{i=1}^{N} \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_m)) = 0$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.
- If $\sum_{i=1}^{N} \mathbf{a}_i(b_i - \mu(\mathbf{a}_i, x_m)) \neq 0$, proceed to Step $m + 1$.

Corollary 4.2.2. *The **RPGA(LR)($\mathcal{D}$)** generates a sequence of vectors $x_k \in \mathbb{R}^m$, such that*

$$e_k \leq Ck^{-1}, \quad k = 1, 2, \ldots, \tag{4.2.8}$$

where $C = C(A, \boldsymbol{b})$

Proof. The proof is a direct application of Theorem 3.1.2 and Lemma 4.2.1.

□

4.2.2.3 Numerical Tests

Test1: We consitder the example of simple logistic regression from Suzuki et al. (2006), where they measured sand grain size on 28 beaches in Japan and observed the presence or absence of the burrowing wolf spider Lycosa ishikariana on each beach (see [16]). One goal of this study is to determine whether there is a relationship between sand grain size and the presence or absence of the species, in

Table 4.1: Data for Test 1 (see [16])

Grain Size(mm)	Spiders	Grain Size(mm)	Spiders
0.245	absent	0.432	absent
0.247	absent	0.473	present
0.285	present	0.509	present
0.299	present	0.529	present
0.327	present	0.561	absent
0.347	present	0.569	absent
0.356	absent	0.594	present
0.36	present	0.638	present
0.363	absent	0.656	present
0.364	present	0.816	present
0.398	absent	0.853	present
0.400	present	0.938	present
0.409	absent	1.036	present
0.421	present	1.045	present

hopes of understanding more about the biology of the spiders. Because this species is endangered, another goal would be to find an equation that would predict the probability of a wolf spider population surviving on a beach with a particular sand grain size, which would help determine which beaches to reintroduce the spider to.

The data is in the following table:

We construct a parameter matrix A and a vector $\mathbf{b}$ with the data above. Let A be a $m \times N$ matrix, with all elements in the first row being 1. The elements in the second row of A are the grain sizes, that is

$$A = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 0.245 & 0.247 & \cdots & 1.036 & 1.045 \end{bmatrix},$$

where $N = 28, m = 2$. To satisfy the LR model we referred before, we denote $A = (\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N)$, where $\mathbf{a}_i$ is the ith column of A. Let the vector $\mathbf{b} \in \mathbb{R}^{N \times 1}$ describe the absence or presence of the wolf spiders, i.e.

$$\mathbf{b} = [0 \quad 0 \quad 1 \quad \cdots \quad 1 \quad 1]^T,$$

where $b_i = 1$ means the spider is present in the sample $\mathbf{a}_i$, and $b_i = 0$ means the spider is absent. Then we plug our parameter matrix A and parameter vector $\mathbf{b}$ into (4.2.3) and (4.2.4) to obtain the objective function in this case and apply the **RPGA(LR)**$(\mathcal{D})$ to approximately find its minimum. We use a dictionary $\mathcal{D} = \{(1,0),(0,1)\}$. We know from [16] that one approximate solution to this problem is $\bar{x} = [-1.6474, 5.1212]^T$, which means that the probability for presence of the spider on a beach with grain size s is

$$P(b=1) = \frac{\exp(-1.6474 + 5.1212s)}{1 + \exp(-1.6474 + 5.1212s)},$$

and its graph is depicted in figure 4.1. We use the value of $\bar{x}$ derived in [16] as a reference solution for our algorithm. That is, we stop the algorithm if $\|x_k - \bar{x}\| < \epsilon$. We select $\epsilon = 10^{-4}$ and use Linesearch$(0.1, 0.5)$ to perfome the one dimensional optimization at each step.

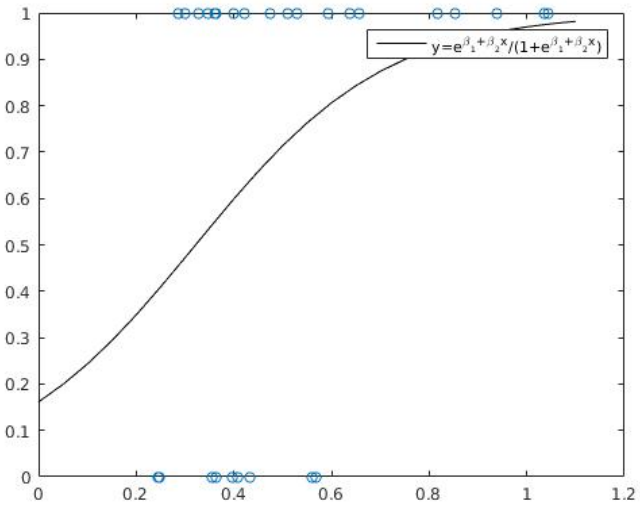

Figure 4.1: Probability Distribution (Test 1)

We plot the error $\|x_k - \bar{x}\|$ for step sizes between $[120, 160]$, see Figure 4.2.

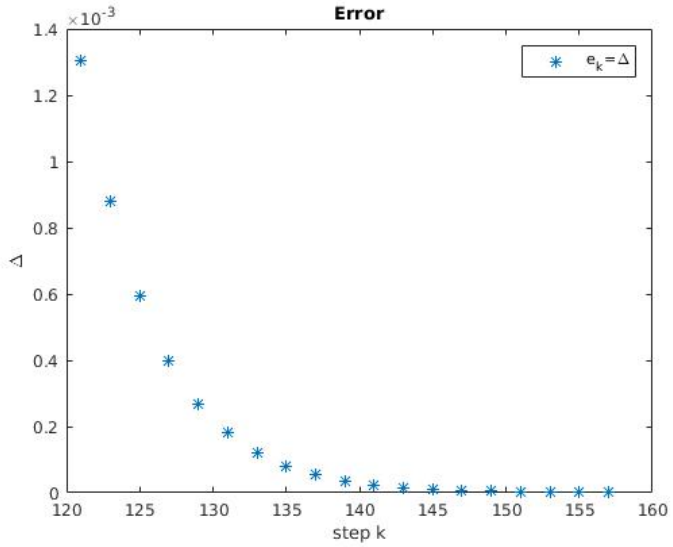

Figure 4.2: **RPGA(LR)**($\mathcal{D}$) Error (Test 1)

At last, we present a table, which shows how many steps k are need, so that the output x_k is ϵ-close to the reference minimum $\bar{x} = [-1.6474, 5.1212]^T$ from [16].

Table 4.2: Error & Steps for Test 1

Error ϵ	Steps
10^{-4}	134
10^{-5}	146
10^{-6}	157
10^{-7}	169
10^{-8}	180
10^{-9}	191
10^{-10}	198

Note that we do not need to know $\bar{x}$ in order to run the algorithm. We use it here simply to demonstrate the convergence of our algorithm, and investigate the number of steps needed to achieve certain prescribed accuracy.

Test 2: This test describes 20 male(denote 1) and female(denote 0) students with aptitude score from 1 to 10 and their admission into a graduate program. (see [17])

Table 4.3: Data for Test 2 (see [17])

Aptitude	Gender	Admission	Aptitude	Gender	Admission
8	1	1	4	0	0
7	1	0	7	0	1
5	1	1	3	0	1
3	1	0	2	0	0
3	1	0	4	0	0
5	1	1	2	0	0
7	1	1	3	0	0
8	1	1	4	0	1
5	1	1	3	0	0
5	1	1	2	0	0

We construct the parameter matrix A and the vector $\mathbf{b}$ with the data above. Let A be the $m \times N$ matrix, with all elements being 1 in the first row. The elements

in the second row reflect the sex of the students, and the elements in third row are their scores of aptitude. The resulting matrix is

$$A = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 0 & 0 \\ 8 & 7 & \cdots & 3 & 2 \end{bmatrix}, \tag{4.2.9}$$

where $N = 20, m = 3$, namely $A = (\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_N)$, where $\mathbf{a}_i$ is the ith column of A. Let the vector $\mathbf{b} \in \mathbb{R}^{N\times 1}$ describe the admission situation namely, $b_i = 1$ means to be admitted and $b_i = 0$ means to be denied. The resulting vector is

$$\mathbf{b} = [1 \quad 0 \quad 1 \quad \cdots \quad 0 \quad 0]^T.$$

We plug the parameter matrix A and the parameter vector $\mathbf{b}$ into (4.2.3) and (4.2.4) to obtain the respective objective function and apply **RPGA(LR)**$(\mathcal{D})$. We use the dictionary $\mathcal{D} = \{(1,0,0),(0,1,0),(0,0,1)\}$. We know from [17] that one approximate solution to this problem is $\bar{x} = [-4.028765, 0.8982803, 0.2671938]^T$. We use this value as a stopping criteria for our algorithm. That is, we stop if $\|x_k - \bar{x}\| < \epsilon$. We select $\epsilon = 10^{-4}$ and use Linesearch$(0.1, 0.5)$ to perfome the one dimensional optimization at each step. We obtain the following table, which shows after how many steps we have derived the solution within the prescribed accuracy ϵ.

Table 4.4: Error & Steps for Test 2

Error ϵ	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}
Steps	3047	4319	5591	6862	8129

Next, we plot the error between 2000 and 3000 steps, see Figure 4.3.

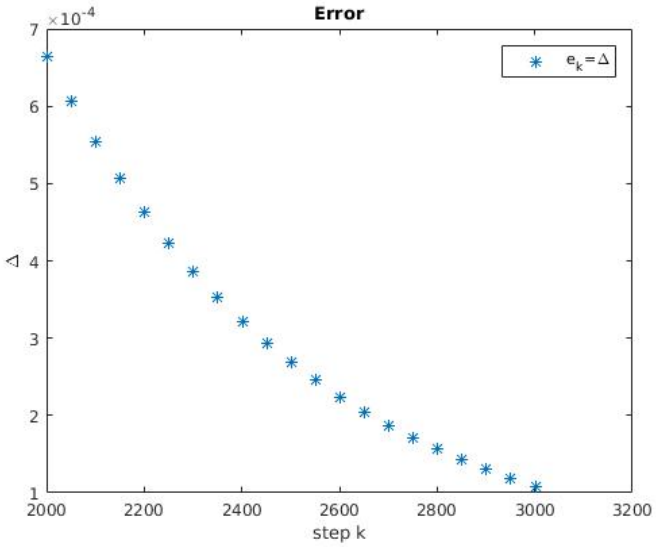

Figure 4.3: **RPGA(LR)**($\mathcal{D}$) Error (Test 2)

We conclude, that both numerical examples confirm the fast convergence rate of our algorithm and its potential to solving real life problems.

4.2.3 Linear Systems

We present a new iterative algorithm for solving linear systems

$$A\bar{x} = b,$$

for any $n \times n$ matrix A with $\det(A) \neq 0$ that generates a sequence of vectors $x_k \in I\!R^n$. This algorithm can be viewed as an application of the suggested rescaled pure greedy algorithm to the function $E(x) = \|Ax - b\|^2$. Before continuing further, we introduce

the notation r_k for the residual

$$r_k := Ax_k - b,$$

and $\kappa(A)$ for the condition number of A, $\kappa(A) := \|A^{-1}\|\|A\|$. All matrix norms, unless specified otherwise, are the matrix norms induced by the Euclidean vector norm $\|x\|^2 = (x, x) = \sum_{i=1}^n x_i^2$.

Let us define for any nonsingular matrix A the function

$$E(x) := \|Ax - b\|^2 = (Ax - b, Ax - b) = (Ax - b, A(x - \bar{x})),$$

and list several properties of E. First, for any vectors $x', x \in \mathbb{R}^n$, we have

$$E(x') - E(x) - 2(Ax - b, A(x' - x)) = \|A(x' - x)\|^2 \leq \|A\|^2\|x' - x\|^2. \quad (4.2.10)$$

Since $\|A(\bar{x} - x^*)\|^2 = E(x^*)$, we have

$$\|\bar{x} - x^*\| \leq \|A^{-1}\|\|A(\bar{x} - x^*)\| = \|A^{-1}\|\|Ax^* - b\| = \|A^{-1}\|E(x^*)^{1/2}. \quad (4.2.11)$$

Next, we present our iterative algorithm, which generates a sequence of vectors $x_k \in \mathbb{R}^n$, and prove the rate of convergence of $x_k \to \bar{x}$ when $k \to \infty$. Let $B = \{\varphi_i\}_{i=1}^n$ be an orthonormal basis for $\mathbb{R}^n$. A canonical example for B is $B = \{e_i\}_{i=1}^n$. In this case the computation of $A\varphi_i$ is simplified since $Ae_i = a^i$, where a^i is the i-th column of A.

RPGA(lin)(B) :

- **Step** 0: Define $x_0 = 0$. Compute $r_0 := Ax_0 - b$. If $r_0 = 0$, stop the algorithm

and define $x_k := x_0 = \bar{x}$, $k \geq 1$.

- **Step** m: Assuming x_{m-1} has been defined and $r_{m-1} \neq 0$.
  - choose a direction $\varphi_{j_m} \in B$ such that

$$|(r_{m-1}, A\varphi_{j_m})| = \sup_{\varphi \in B} |(r_{m-1}, A\varphi)|.$$

  - compute

$$\lambda_m := \frac{1}{\|A\|^2}(r_{m-1}, A\varphi_{j_m}), \quad \hat{x}_m := x_{m-1} - \lambda_m \varphi_{j_m}$$

    and

$$t_m := \frac{(A\hat{x}_m, b)}{(A\hat{x}_m, A\hat{x}_m)}.$$

  - define the next approximation to be

$$x_m = t_m \hat{x}_m.$$

  - compute the residual $r_m := Ax_m - b$.

- If $r_m = 0$, stop the algorithm and define $x_k = x_m = \bar{x}$, for $k > m$.
- If $r_m \neq 0$, proceed to Step $m + 1$.

The next theorem provides the rate of convergence of the proposed algorithm.

Theorem 4.2.3. *Consider the linear system $A\bar{x} = b$ with an $n \times n$ nonsingular matrix A. The* **RPGA(lin)**(B) *generates a sequence of vectors $x_k \in \mathbb{R}^n$ such that*

$$\|x_k - \bar{x}\| \leq \|A^{-1}\|\|b\| \left(1 - \frac{1}{n\kappa(A)^2}\right)^{\frac{k}{2}}, \quad k \geq 1,$$

where $\kappa(A)$ is the condition number of A.

Proof. We first find an estimate for $E(\hat{x}_k)$ by using (4.2.10) with the values

$$x' = \hat{x}_k = x_{k-1} - \lambda_k \varphi_{j_k}, \quad x = x_{k-1}.$$

As above, we get that

$$\begin{aligned} E(\hat{x}_k) &= E(x_{k-1} - \lambda_k \varphi_{j_k}) = E(x_{k-1}) - 2\lambda_k (r_{k-1}, A\varphi_{j_k}) + \|A(\lambda_k \varphi_{j_k})\|^2 \\ &\leq E(x_{k-1}) - 2\lambda_k (r_{k-1}, A\varphi_{j_k}) + \|A\|^2 \lambda_k^2 \\ &= E(x_{k-1}) - \frac{1}{\|A\|^2} (r_{k-1}, A\varphi_{j_k})^2, \end{aligned} \tag{4.2.12}$$

where we have used that $\|\varphi_{j_k}\| = 1$ and the choice of λ_k. Since the quadratic function Φ, defined as

$$\Phi(t) := t^2 (A\hat{x}_k, A\hat{x}_k) - 2t(A\hat{x}_k, b) + \|b\|^2 = \|A(t\hat{x}_k) - b\|^2 \geq 0$$

achieves minimum at $t = t_k$, we have that

$$E(x_k) = \Phi(t_k) \leq \Phi(1) = E(\hat{x}_k),$$

and therefore it follows from (4.2.12) that

$$E(x_k) \leq E(\hat{x}_k) \leq E(x_{k-1}) - \frac{1}{\|A\|^2} (r_{k-1}, A\varphi_{j_k})^2. \tag{4.2.13}$$

Now we derive a lower bound for $(r_{k-1}, A\varphi_{j_k})^2$. Since

$$x_{k-1} - \bar{x} = \sum_{i=1}^{n} c_i (x_{k-1} - \bar{x}) \varphi_i,$$

where $c_i(x_{k-1} - \bar{x})$ are the coefficients in the representation of $x_{k-1} - \bar{x}$ with respect to the basis B, we have

$$\begin{aligned} E(x_{k-1}) &= (r_{k-1}, A(x_{k-1} - \bar{x})) = \sum_{i=1}^{n} c_i(x_{k-1} - \bar{x})(r_{k-1}, A\varphi_i) \\ &\leq |(r_{k-1}, A\varphi_{j_k})| \sum_{i=1}^{n} |c_i(x_{k-1} - \bar{x})| \\ &\leq |(r_{k-1}, A\varphi_{j_k})| \sqrt{n} \sqrt{\sum_{i=1}^{n} c_i^2(x_{k-1} - \bar{x})}, \\ &= |(r_{k-1}, A\varphi_{j_k})| \sqrt{n} \|x_{k-1} - \bar{x}\|, \\ &\leq \|A^{-1}\| \sqrt{n} |(r_{k-1}, A\varphi_{j_k})| E(x_{k-1})^{1/2}, \end{aligned}$$

where we have used the definition of φ_{j_k}, Cauchy's inequality and (4.2.11). Therefore, we obtain the lower bound

$$\frac{1}{n\|A^{-1}\|^2} E(x_{k-1}) \leq (r_{k-1}, A\varphi_{j_k})^2, \tag{4.2.14}$$

and (4.2.13) becomes

$$\begin{aligned} E(x_k) &\leq E(x_{k-1}) - \frac{1}{n\|A\|^2\|A^{-1}\|^2} E(x_{k-1}) \\ &= E(x_{k-1}) \left(1 - \frac{1}{n\kappa(A)^2}\right). \end{aligned}$$

Since $E(x_0) = \|b\|^2$, the above inequality gives

$$E(x_k) \leq \|b\|^2 \left(1 - \frac{1}{n\kappa(A)^2}\right)^k,$$

and therefore

$$\|x_k - \bar{x}\| \leq \|A^{-1}\| E(x_k)^{1/2} \leq \|A^{-1}\| \|b\| \left(1 - \frac{1}{n\kappa(A)^2}\right)^{\frac{k}{2}},$$

which completes the proof. □

4.2.4 Stability Analysis

In this section, we present a stability analysis for our algorithm in the sense that we investigate how errors in the computation of λ_m propagate. More precisely, we compute λ_m the following way:

$$\lambda_m := \frac{1 + \epsilon_m}{\|A\|^2}(r_{m-1}, A\varphi_{j_m}), \quad |\epsilon_m| < 1. \tag{4.2.15}$$

The following theorem holds.

Theorem 4.2.4. *Consider the linear system $A\bar{x} = b$ with an $n \times n$ nonsingular matrix A. The* **RPGA(lin)(B)**, *where λ_m is computed according to* (4.2.15) *generates a sequence of vectors $x_k \in \mathbb{R}^n$ such that*

$$\|x_k - \bar{x}\| \leq \|A^{-1}\| \|b\| \prod_{i=1}^{k} \left(1 - \frac{1 - \epsilon_i^2}{n\kappa(A)^2}\right)^{\frac{1}{2}}, \quad k \geq 1,$$

where $\kappa(A)$ is the condition number of A.

Proof. As in the proof of Theorem 4.2.3, we derive that

$$\begin{aligned} E(x_k) \leq E(\hat{x}_k) &\leq E(x_{k-1}) - 2\lambda_k(r_{k-1}, A\varphi_{j_k}) + \|A\|^2\lambda_k^2 \\ &= E(x_{k-1}) - \frac{1 - \epsilon_k^2}{\|A\|^2}(r_{k-1}, A\varphi_{j_k})^2. \end{aligned} \tag{4.2.16}$$

The lower bound for $(r_{k-1}, A\varphi_{j_k})^2$ is as in (4.2.14) and (4.2.16) becomes

$$E(x_k) \;\leq\; E(x_{k-1})\left(1 - \frac{1-\epsilon_k^2}{n\kappa(A)^2}\right).$$

Since $E(x_0) = \|b\|^2$, the above inequality gives

$$E(x_k) \leq \|b\|^2 \prod_{i=1}^{k}\left(1 - \frac{1-\epsilon_i^2}{n\kappa(A)^2}\right),$$

and therefore

$$\|x_k - \bar{x}\| \leq \|A^{-1}\| E(x_k)^{1/2} \leq \|A^{-1}\|\|b\| \prod_{i=1}^{k}\left(1 - \frac{1-\epsilon_i^2}{n\kappa(A)^2}\right)^{\frac{1}{2}},$$

which completes the proof. □

Remark 4.2.5. *If* $|\epsilon_m| \leq \epsilon < 1$ *in* (4.2.15)*, then*

$$\|x_k - \bar{x}\| \leq \|A^{-1}\|\|b\| \left(1 - \frac{1-\epsilon^2}{n\kappa(A)^2}\right)^{\frac{k}{2}},$$

5. SUMMARY

In the thesis, we introduced Rescaled Pure Greedy Algorithm (**RPGA(co)**) for solving convex optimization problems in Banach spaces. In chapter 3 we discussed **RPGA(co)** and analyzed its convergence rate. Moreover, we also introduced and analyzed a weak form of **RPGA(co)**, which is Weak Rescaled Pure Greedy Algorithm (**WRPGA(co)**). Our analysis showed that **RPGA(co)** converges fast under certain given conditions.

We also applied **RPGA(co)** on several examples in chapter 4, such as approximation in Hilbert spaces, solving linear systems, and others. Especially in section 4.2.2, we performed several numerical tests, and compared with those results given in [16, 17]. We concluded that **RPGA(co)** has fast convergence rate as we analyzed in chapter 3 and confirmed its potential for solving real world problems.

We also applied **RPGA(co)** on solving linear systems. We analyzed the algorithm and showed that the convergence rate is as good as that in chapter 3. In order to verify our analysis, we will perform numerical tests for **RPGA(co)** in solving linear systems in our future work.

REFERENCES

[1] J. Borwein, A. Guiro, P. Hajek, and J. Vanderwerff, *Uniformly convex functions on Banach Spaces*, Proc. Amer. Math. Soc., **137**, 1081–1091, 2009.

[2] S. Boyd, L. Vandenberghe, Convex optimization, Cambridge University Press, 2009.

[3] R. DeVore, V. Temlyakov, *Some remarks on greedy algorithms*, Advances in Computational Math., **5**, 173–187, 1996.

[4] R. DeVore, V. Temlyakov, *Convex optimization on Banach spaces*, Foundations of Computational Mathematics, accepted.

[5] W. W. Hager, H. Zhang, *A survey of nonlinear conjugate gradient methods*, Pacific Journal of Optimization, **2** , 35–58, 2006.

[6] A. Nemirovski, *Optimization II: Numerical methods for nonlinear continuous optimization*, Lecture Notes, Israel Institute of Technology, 1999.

[7] H. Nguyen, G. Petrova, *Greedy strategies for convex optimization*, Calcolo, to appear, *arXiv:1401.1754*.

[8] G. Petrova, *Rescaled Pure Greedy Algorithm for Hilbert and Banach Spaces*, Applied and Computational Harmonic Analysis, to appear, *arXiv:1505.03604*.

[9] V. Temlyakov, *Greedy expansions in convex optimization*, Proceedings of the Steklov Institute of Mathematics, **284**(1), 244–262, 2014.

[10] V. Temlyakov, *Greedy approximation in convex optimization*, Constr. Approx., **41**(2), 269–296, 2015.

[11] V. Temlyakov, Greedy approximation, Cambridge monographs on Applied and Computational Mathematics, Cambridge University Press, 2011.

[12] C. Zalinescu, Convex Analysis in General Vector Spaces, World Scientific Publishing Co. Inc., River Edge, NJ, 2002.

[13] T. Zhang, *Sequential greedy approximation for certain convex optimization problems*, IEEE Transactions on Information Theory, **49**(3), 682–691, 2003.

[14] P. Komarek, *Logistic Regression for Data Mining and High-Dimensional Classification*, Chapter 4, 22-28, PhD Dissertation, Dept. of Math Sciences, Carnegie Mellon University, 2004.

[15] G. Casella, R.L.Berger, Statistical Inference, Duxbury Press, 2002.

[16] J.H. McDonald, Handbook of Biological Statistics (3rd ed.). Sparky House Publishing, Baltimore, Maryland, 2014.

[17] Sharyn O'Halloran, 2005, Admission Data, Retrieved from http://www.columbia.edu/∼so33/SusDev/Lecture