Speeding up Gradient Descent

1 Introduction and Background

Gradient descent has been the optimization algorithm of choice in a large amount of optimization applications, especially those which operate at large scales (where Hessians are too difficult to repeatedly compute). Yet gradient descent is not the fastest converging algorithm, even compared to other algorithms which have just zeroth and first-order information (i.e., knowledge of the function and its gradient). Indeed, Nesterov showed that it is possible to design a sped-up version of gradient descent, called *accelerated gradient descent*, which has asymptotically faster convergence rate than gradient descent [1].[1] This accelerated method is, at least at first, seemingly complicated and unmotivated; the original proof of convergence rate given by Nesterov amounted to a set of mysterious algebraic manipulations. Even one of the most well-known optimization theorists, Sebastian Bubeck, posted on his blog about how understanding Nesterov's acceleration is difficult to understand.

Recently, there have been several lines of research attempting to understand and extend Nesterov's acceleration. One of these attempts uses a generalization of gradient descent called *mirror descent* combined with gradient descent to improve the convergence rate of the algorithm [3], giving an interpretable modification of gradient descent which achieves optimal convergence rates. In this project, we introduce mirror descent and apply it to accelerate gradient descent.

[1] In fact, it achieves asymptotically optimal convergence rates; a proof of this is contained in [2].

2 Overview of Relevant Literature

In the first section of this project, you will read several research papers which are relevant to the topic of this project, and summarize and synthesize them into a related work section. The aim is to gain a better understanding of the topics discussed in this project and to get insights into the state-of-the-art development in our understanding of accelerating gradient descent.

In the related works section, you will summarize the main results and findings for at least three papers. It is especially useful and interesting to write about any common threads you find across multiple papers. We will assign two below, along with some questions you may think about while reading. The remaining paper(s) you choose to read can be found via Google or other sources.[2]

The papers we assign are the following:

1. "A Method of Solving a Convex Programming Problem with Convergence Rate $O(1/k^2)$" by Nesterov [1]. While writing your summary, please mention the following points.
   (a) Describe the main assumptions made in the paper on the function f to optimize. What kinds of functions are shown to be optimized via accelerated gradient descent? (*HINT:* In this paper, Nesterov uses the notation f' to denote the gradient ∇f.)
   (b) What is the update rule at the k^{th} step?
   (c) After T iterations, what is an upper-bound on $f(\vec{x}_T) - \min_{\vec{x}\in\mathbb{R}^n} f(\vec{x})$?
2. "Linear Coupling: An Ultimate Unification of Gradient and Mirror Descent" by Allen-Zhu and Orecchia [3]. While writing your summary, please mention the following points.
   (a) What do the authors claim is the kind of progress made by gradient descent? What about mirror descent?
   (b) Describe the main assumptions made in the paper on the function f to optimize. What kinds of functions are shown to be optimized via the algorithm `AGM` proposed by Allen-Zhu and Orecchia?
   (c) After T iterations of `AGM`, what is an upper-bound on $f(\vec{y}_T) - \min_{\vec{x}\in Q} f(\vec{x})$?
   (d) What is the relationship between Allen-Zhu and Orecchia's linear coupling paper [3] and Nesterov's original accelerated gradient descent paper [1]?
3. The third (and beyond) papers you choose to read yourself must relate to the topic of accelerated optimization algorithms, but the exact paper(s) are your choice. For any additional paper(s), please add a summary of the findings of the paper, and describe how the paper relates to [1] and [3]. What is the novel contribution of the new paper you have read? Why is it important/relevant to the field? What is an open question that remains after reading the paper?

[2] Given a research paper written recently, one way to see papers related to it is to go to Google Scholar and type in the paper title, then look at all papers which cite the paper you started with (via "Cited by `$NUM`"); these tend to continue the same research threads or demonstrate applications. Another way is to look up the papers cited by the papers you have already read, focusing on those which have relevant titles. Each paper collects a list of sources cited near the end of the paper.

3 Problems

1. Mirror Descent

In this problem, we will go through a proof of mirror descent in the case of entropy and ℓ^2 regularization.

Consider the problem

$$f^\star = \min_{\vec{x}\in\mathcal{X}} f(\vec{x}). \tag{1}$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is a convex subset of $\mathbb{R}^n$. Often, running (projected) gradient descent is not the right thing to do and can lead to slow convergence. This is usually because the convergence rates of Euclidean gradient descent are of the form $f(\vec{x}_k) - f^\star \leq O(L\|\vec{x}_0 - \vec{x}^\star\|_2^2/k)$ and it may be the case for $\mathcal{X}$ that the ℓ^2 distance of the initial point from the optimal and/or the smoothness parameter L that's measured according to the ℓ^2 may be quite large. But taking the "right geometry" into account when *defining the update step* may lead to much faster convergence. For the sake of simplicity, however, for all except one subpart, we will only work for the ℓ^2 case in this problem.

For any convex (doubly)-differentiable function $h : \mathcal{X} \to \mathbb{R}$, define the Bregman divergence of h as

$$D_h(\vec{y};\vec{x}) = h(\vec{y}) - h(\vec{x}) - \langle \nabla h(\vec{x}), \vec{y} - \vec{x}\rangle \tag{2}$$

where $\langle \vec{u}, \vec{v}\rangle$ is the dot product between two vectors $\vec{u}$ and $\vec{v}$.

Algorithm 1 Mirror Descent Algorithm

$\vec{x}_0$ is a uniformly random point in $\mathcal{X}$
$k = 0$
while $k \leq T$ **do**
  $\eta_k > 0$ a step size.
  $\vec{g}_k \leftarrow \nabla f(\vec{x}_k)$
  $\vec{x}_{k+1} = \text{Mirr}(\eta_k \vec{g}_k; \vec{x}_k)$ where $\text{Mirr}(\vec{g};\vec{x}) = \operatorname{argmin}_{\vec{z}\in\mathcal{X}} \{\langle \vec{g}, \vec{z}\rangle + D_h(\vec{z};\vec{x})\}$ is the Mirror Descent step.
  $k \leftarrow k + 1$
end while
return $\overline{x}_T = \frac{1}{T}\sum_{k=0}^{T} \vec{x}_k$

(a) Prove that for any convex set $\mathcal{X}$ and α-strongly convex function $h : \mathcal{X} \to \mathbb{R}$, for any fixed $\vec{x} \in \mathcal{X}$, the Bregman divergence $D_h(\vec{y};\vec{x})$ is a α-strongly convex function of $\vec{y}$. Note that by taking $\alpha = 0$, this proves that if h is convex, the Bregman divergence is convex as well.

(b) Something that's going to be useful in a convergence proof of mirror descent is going to be the so-called Bregman three-point inequality. Formally, prove that

$$\langle \nabla h(\vec{x}) - \nabla h(\vec{y}), \vec{y} - \vec{u}\rangle = D_h(\vec{u};\vec{x}) - D_h(\vec{u};\vec{y}) - D_h(\vec{y};\vec{x}) \tag{3}$$

(c) Let's try to understand the mirror descent update in some special cases. For this part, assume $\mathcal{X} = \{\vec{x} \in \mathbb{R}^n | x_i \geq 0\ \forall i \in [n]$ and $\sum_{i=1}^{n} x_i = 1\}$ is the n-dimensional probability simplex. We will take $h(\vec{x}) = \sum_{i=1}^{n} (x_i \log(x_i) - x_i)$, the entropy function. Given $\vec{x} \in \mathcal{X}$ and given some $\vec{g} \in \mathbb{R}^n$ and $\eta > 0$, compute

Mirr$(\eta\vec{g};\vec{x})$. Since $\mathcal{X}$ is constrained to be the simplex, you will have to use a Lagrange multiplier for the $\sum_{i=1}^{n} x_i = 1$ constraint and eliminate the Lagrange multiplier from the final solution. In this setting, this algorithm goes by a more popular name which is "multiplicative weights update method".

(d) Now, for the rest of the problem, for simplicity, we will assume $\mathcal{X} = \mathbb{R}^n$ and $h(\vec{x}) = \frac{1}{2}\|\vec{x}\|_2^2$. In this case, given a $\vec{g} \in \mathbb{R}^n$, $\eta > 0$ and $\vec{x} \in \mathcal{X}$, compute Mirr$(\eta\vec{g};\vec{x})$. Also compute $D_h(\vec{y},\vec{x})$ in this case. Do these look familiar to something you have already seen?

(e) To prove convergence of mirror descent, it's convenient to introduce a term from online learning, called regret. For any feasible solution $\vec{u} \in \mathcal{X}$, we define regret in the k^{th} iteration as $\text{Reg}_k(\vec{u}) = \langle \eta_k\vec{g}_k, \vec{x}_k - \vec{u}\rangle$. We will first prove an upper bound on the regret of the k^{th} iteration. Formally, prove that

$$\text{Reg}_k(\vec{u}) = \langle \eta_k\vec{g}_k, \vec{x}_k - \vec{u}\rangle = \langle \eta_k\vec{g}_k, \vec{x}_k - \vec{x}_{k+1}\rangle + D_h(\vec{u};\vec{x}_k) - D_h(\vec{u};\vec{x}_{k+1}) - D_h(\vec{x}_{k+1};\vec{x}_k) \tag{4}$$

$$= \frac{\eta_k^2\|\vec{g}_k\|_2^2}{2} + D_h(\vec{u};\vec{x}_k) - D_h(\vec{u};\vec{x}_{k+1}) \tag{5}$$

This inequality will show up again in the proof of accelerated gradient descent in another question in the project.

HINT: Think of the first term on the equality that has to be proven. Looking at that, maybe adding and subtracting from the regret may help?

(f) Now we will consider the total regret over T iterations, i.e., $\text{TotalReg}_T(\vec{u}) = \sum_{i=0}^{T} \langle \eta_k\vec{g}_k, \vec{x}_k - \vec{u}\rangle$. Prove that

$$\text{TotalReg}_T(\vec{u}) \leq \sum_{k=0}^{T} \eta_k^2\|\vec{g}_k\|_2^2 + D_h(\vec{u};\vec{x}_0) \tag{6}$$

(g) Now, we will prove a lower bound on the regret in terms of the function value at $\overline{x}_T$ and at $\vec{u}$. Taking $\eta_k = \eta$ for all k, prove that

$$\text{TotalReg}_T(\vec{u}) \geq T\eta(f(\overline{x}_T) - f(\vec{u})) \tag{7}$$

Using this, conclude that

$$f(\overline{x}_T) \leq f(\vec{x}^\star) + \frac{1}{T}\left[\eta\sum_{i=0}^{T}\|\vec{g}_k\|_2^2 + D_h(\vec{x}^\star;\vec{x}_0)/\eta\right] \tag{8}$$

(h) Now, assume that the function is L-Lipschitz (note that this is asking for the function to be Lipschitz and not the gradient of the function to be L-Lipschitz). It can be easily proven (and you may assume so without proof) that $\|\nabla f(\vec{x})\|_2 \leq L$ for all $\vec{x} \in \mathcal{X}$. Conclude that

$$f(\overline{x}_T) \leq f(\vec{x}^\star) + \eta L^2 + \frac{1}{2\eta T}\|\vec{x}_0 - \vec{x}^\star\|_2^2 \tag{9}$$

Show that there exists an $\eta > 0$ such that

$$f(\overline{x}_T) \leq f(\vec{x}^\star) + \frac{\sqrt{2}L\|\vec{x}_0 - \vec{x}^\star\|_2}{\sqrt{T}} \tag{10}$$

HINT: Can you try to optimize the total regret upper bound by optimizing it as a function of η? Hence the convergence is at a rate of $1/\sqrt{T}$.

(i) In the accompanying Jupyter notebook, you will implement the mirror descent update for the entropy regularizer and for the ℓ^2 regularizer. The input to the function will be step size η_k, a vector $\vec{g}$ which is meant to represent the gradient, and the current point $\vec{x}_k$.

Hence the convergence is at a rate of $1/\sqrt{T}$. While we did the proof for the unconstrained setting and with the ℓ^2 geometry, this proof with very few changes can be used to prove similar results in constrained settings and with other geometries, which show up, for example, in the probability simplex case corresponding to the multiplicative weights update method, whose mirror update you calculated above.

2. Accelerated Gradient Descent

In this problem, we will go through a proof of accelerated gradient descent by combining a gradient descent and a mirror descent step. You will also implement this algorithm in an accompanying Jupyter notebook and use it to optimize some specific functions.

Recall that in lecture, in the proof of gradient descent for L-smooth functions, we proved the following inequality:

$$f(\vec{x}_+) \leq f(\vec{x}) - \frac{1}{2L}\|\nabla f(\vec{x})\|_2^2 \tag{11}$$

where $\vec{x}_+ = \vec{x} - \frac{1}{L}\nabla f(\vec{x})$ is the gradient descent step. We will need this inequality as well as the inequality you proved in part (e) in the Mirror Descent problem which bounds the per iteration regret.

Let's now describe an accelerated gradient descent algorithm. We will work in the unconstrained optimization setting so that $\mathcal{X} = \mathbb{R}^n$ and will use the ℓ^2 geometry and we assume f is convex and L-smooth.

Algorithm 2 Acceleration via Combining Gradient and Mirror Descent

$\vec{x}_0 = \vec{y}_0 = \vec{z}_0$ is a uniformly random point in $\mathcal{X}$
$k = 0$
while $k \leq T$ **do**
  $\vec{x}_{k+1} = \tau_k \vec{z}_k + (1-\tau_k)\vec{y}_k$ for $\tau_k = 2/(k+2)$
  $\vec{y}_{k+1} \leftarrow x_{k+1} - \frac{1}{L}\nabla f(\vec{x}_{k+1})$
  $\vec{z}_{k+1} = \text{Mirr}(\eta_{k+1}\nabla f(\vec{x}_{k+1}); \vec{z}_k)$ where $\eta_{k+1} = (k+2)/2L = 1/(\tau_k L)$
  $k \leftarrow k+1$
end while
  return y_T

Here the h function defining the Bregman divergence for the mirror descent step is just $h(\vec{x}) = \frac{1}{2}\|x\|_2^2$.

(a) We first understand the regret on the mirror update. Formally prove that,

$$\langle \eta_{k+1}\nabla f(\vec{x}_{k+1}), \vec{z}_k - \vec{u}\rangle = \frac{\eta_{k+1}^2}{2}\|\nabla f(\vec{x}_{k+1})\|_2^2 + D(\vec{u};\vec{z}_k) - D(\vec{u};\vec{z}_{k+1}) \tag{12}$$

HINT: In part (e) of the mirror descent question, would the proof still work if $\vec{g}_k$ was something other than $\nabla f(\vec{z}_k)$?

(b) Now, we try to understand the regret of $\vec{x}_{k+1}$. Formally, prove that,

$$\langle \eta_{k+1}\nabla f(\vec{x}_{k+1}), \vec{x}_{k+1} - \vec{u}\rangle \tag{13}$$

$$= \frac{(1-\tau_k)\eta_{k+1}}{\tau_k}\langle \nabla f(\vec{x}_{k+1}), \vec{y}_k - \vec{x}_{k+1}\rangle + \frac{\eta_{k+1}^2}{2}\|\nabla f(\vec{x}_{k+1})\|_2^2 + D(\vec{u};\vec{z}_k) - D(\vec{u};\vec{z}_{k+1}). \tag{14}$$

Furthermore, show that one can upper bound the RHS above by

$$\frac{(1-\tau_k)\eta_{k+1}}{\tau_k}(f(\vec{y}_k) - f(\vec{x}_{k+1})) + \eta_{k+1}^2 L(f(\vec{x}_{k+1}) - f(\vec{y}_{k+1})) + D(\vec{u};\vec{z}_k) - D(\vec{u};\vec{z}_{k+1}). \tag{15}$$

HINT: In the $\vec{x}_{k+1} - \vec{u}$ term, add and subtract $\vec{z}_k$ and use the previous part along with the definition of $\vec{x}_{k+1}$.

(c) Now deduce that

$$\eta_{k+1}^2 L f(\vec{y}_{k+1}) - (\eta_{k+1}^2 L - \eta_{k+1}) f(\vec{y}_k) - D(\vec{u}; \vec{z}_k) + D(\vec{u}; \vec{z}_{k+1}) \leq \eta_{k+1} f(\vec{u}) \tag{16}$$

HINT: You will need to use the specific values of η_k and τ_k as defined in the algorithm definition to observe some cancellations

(d) Now, summing up the inequality in the previous part and plugging in values for η_{k+1}, conclude that

$$f(\vec{y}_T) \leq f(\vec{x}^\star) + \frac{2L\|\vec{x}^\star - \vec{x}_0\|_2^2}{(T+1)^2} \tag{17}$$

(e) In the accompanying Jupyter notebook, implement the above acceleration via gradient plus mirror descent step. Run the algorithm on a given low rank quadratic optimization problem. Report how the algorithm performs as compared to Gradient Descent, Adam, Adagrad algorithms. In the logistic regression `fval` function, inside the logarithm terms, you should add a small epsilon like `1e-10` in order to ensure there are no NaNs in the output.

4 Extension

Now that you've seen some of the research landscape and worked through some of the resulting problems, the last part of the project is to complete an *extension* of the project material. An extension is a self-directed addition to the research elucidated in the related work and the problem set. You should include the extension material in your writeup after the problem set.

When selecting a topic for the extension, the following questions may be reasonable to ask:

- Was there a question that arose when reading related work or doing the project that seems like it might be interesting to investigate?
- Do the methods in the papers read for related work or the project itself extend to a particular interesting setting?
- Does a different method solve the same problems considered here? How can the methods be compared (on axes like efficiency, simplicity, etc)?

The extension would ideally answer at least one of these questions constructively, or some other question along the same lines. For example, in the first case, you would write down the question and try to answer it (via theory and/or experiments). In the second case, you would investigate the method in the paper applied to the setting of your interest. In the third case, you would apply the different method to the same paper, or compare it with some baseline methods.

Often, an extension idea will come from reading related work. Some broad ideas for possible extensions which heavily draw upon related work are:

- Try to replicate a piece of related work that's been done and that you have read about. Contrast this with the work done in the project.
- As we have done in this project, present in detail a simple (or complex if you want) case from a related paper that you read.

This is definitely not an extensive list. The overall guideline is that we must be able to understand what *you* have done from your work. It can help if you can crisply formulate a question that you are trying to answer. If you cannot, then it might not be a good extension.

Please note that *to successfully complete the extension, you do not need to have a breakthrough!* It is very difficult to come up with and answer a significant research question in a few weeks, so you should not feel discouraged if this is not possible for you. If you are ever feeling stuck, we encourage you to come to office hours to discuss your extension with us!

5 Deliverables

Your submission should contain:

1. A PDF of your final report. Your final report should be written in LaTeX. You are recommended to use the template that has been provided on the course website. You may choose not to typeset your equations, but in that case the document should include very high quality, cleanly handwritten scans of your work; we will not try to read illegible content. In particular, we prefer that you typeset your work, since it is an important skill to learn, and a nicely typeset project report (put on your website or CV) can be a strong showcase of the work you have done.

   Your report should include

   (a) An abstract, i.e., a paragraph length summary of what is in your document.

   (b) An introduction, which describes what the research problem studied in the project is and why it is important.

   (c) A literature review, whose guidelines were elaborated in Section 2.

   (d) Solutions to all guided portions of the project in Section 3, including relevant plots, figures, or code snippets that are needed to answer questions.

   (e) A detailed description of the work you did for your extension in Section 4.

   (f) A contributions section, i.e., a description of which members of your group did what work for the project.

2. Uploaded code, including Jupyter notebooks, for your work in the guided portion of the project (Section 3).

3. Uploaded code, including Jupyter notebooks, for your work in the project extension (Section 4).

All of these deliverables (project report PDF, code for the guided part of the project, and code for the extension) will have separate Gradescope assignments.

6 Rubric

To get any grade, you must submit a project report with:

- An abstract summarizing the report;
- An introduction section;
- A literature review section, which contains a literature review as detailed in Section 2;
- A results section, which contains the solutions for the guided portion of the project as detailed in Section 3;
- An extension section, which contains a summary of your extension as detailed in Section 4;
- A contribution section;

as well as upload your code (including Jupyter notebooks) for your work in the guided portion of the project as well as the extension (if applicable). Once these requirements are met, your grade is based on the quality of the report.

- To get a C, your project report must fulfill the following requirements:
  - Your introduction must describe the research problem clearly and in detail;
  - Your literature review must summarize both provided papers and at least one more, and contain mostly-correct answers to the provided questions in Section 2;
  - Your results for the guided portion of the project (Section 3) must contain:
    * A mostly correct implementation for either problem 1(i) or problem 2(e) (or both), up to minor bugs;
    * Correct solutions for any four problems from 1(a) — 1(h) and any two problems from 2(a) — 2(d).
  - Your extension section may be blank (i.e., you do not need an extension to get a C).
- To get a B, your project report must fulfill the following requirements:
  - Your introduction must describe the research problem clearly and in detail;
  - Your literature review must summarize both provided papers and at least one more, and contain mostly-correct answers to the provided questions in Section 2;
  - Your results for the guided portion of the project (Section 3) must contain:
    * A mostly correct implementation for problems 1(i) and 2(e), up to minor bugs;
    * Correct solutions for any six problems from 1(a) — 1(h) and any three problems from 2(a) — 2(d).
  - Your extension section may be blank (i.e., you do not need an extension to get a B).
- To get a A, your project report must fulfill the following requirements:
  - Your introduction must describe the research problem clearly and in detail;
  - Your literature review must summarize both provided papers plus at least one more related work and contain completely correct answers to the provided questions in Section 2;
  - Your results for the guided portion of the project (Section 3) must contain completely correct mathematical solutions and/or code implementations for all parts;
  - Your extension must contain significant, detailed, and organized work towards summarizing or synthesizing existing research, as per the guidelines in Section 4. In particular, it should start with a clear research question and document one or more attempts to answer this question. Note that the question does not need to be conclusively answered — this rubric item just asks for a thoughtful attempt to be made.

References

[1] Y. E. Nesterov, "A method of solving a convex programming problem with convergence rate o\bigl(𝕜2\bigr)," in *Doklady Akademii Nauk*, Russian Academy of Sciences, vol. 269, 1983, pp. 543–547.

[2] Y. Nesterov *et al.*, *Lectures on convex optimization*. Springer, 2018, vol. 137.

[3] Z. Allen-Zhu and L. Orecchia, "Linear coupling: An ultimate unification of gradient and mirror descent," *arXiv preprint arXiv:1407.1537*, 2014.

