A. Appendix

A.1. Convergence for non-Lipschitz PL functions

Karimi et al. (2016) proved that SGD with an appropriate step size will give a $O(1/T)$ convergence for Lipschitz and PL functions. However, it is easy to see that the Lipschitz assumption can be substituted by the smoothness one and obtain a rate that depends on the variance of the noise. Even if this is a straightforward result, we could not find it anywhere so we report here our proof.

Theorem 5. *Assume **(A1)** and **(A3)** and set the step sizes* $\eta_t = \min\left(\frac{1}{L(1+a)}, \frac{2t+1}{\mu(t+1)^2}\right)$. *Then, SGD guarantees*

$$f(\boldsymbol{x}_{T+1}) - f^\star \leq \frac{L^2(1+a)b}{2\mu^3T^2} + \frac{2L}{\mu^2T}b + (f(\boldsymbol{x}_1) - f^\star)\frac{L^2(1+a)^2}{\mu^2T^2}\left(1 - \frac{\mu}{L(1+a)}\right)^{\frac{L(1+a)}{\mu}} .$$

Proof. For simplicity, denote $\mathbb{E}f(\boldsymbol{x}_t) - f^\star$ by Δ_t. With the same analysis as in Theorem 1, we have

$$\Delta_{t+1} \leq (1 - \mu\eta_t)\,\Delta_t + \frac{L}{2}\eta_t^2 b\,.$$

Denote by $t^\star = \min\left\{t : \frac{t^2}{2t+1} \leq \frac{L(1+a)-\mu}{\mu}\right\}$. When $t \leq t^\star$, $\eta_t = \frac{1}{L(1+a)}$ and we obtain

$$\Delta_{t+1} \leq \left(1 - \frac{\mu}{L(1+a)}\right)\Delta_t + \frac{b}{2L(1+a)^2}\,.$$

Thus, by Lemma 2, we get

$$\begin{aligned}\Delta_{t^\star} &\leq \left(1 - \frac{\mu}{L(1+a)}\right)^{t^\star-1}\Delta_1 + \frac{b}{2L(1+a)^2}\sum_{i=0}^{t^\star}\left(1 - \frac{\mu}{L(1+a)}\right)^{t^\star-i} \\ &\leq \left(1 - \frac{\mu}{L(1+a)}\right)^{t^\star}\Delta_1 + \frac{b}{2\mu(1+a)}\,.\end{aligned}$$

Instead, when $t \geq t^\star$, $\eta_t = \frac{2t+1}{\mu(t+1)^2}$, we have

$$\Delta_{t+1} \leq \frac{t^2}{(t+1)^2}\Delta_t + \frac{L(2t+1)^2}{2\mu^2(t+1)^4}b.$$

Multiplying both sides by $(t+1)^2$ and denoting by $\delta_t = t^2\Delta_t$, we get

$$\delta_{t+1} \leq \delta_t + \frac{L(2t+1)^2}{2\mu^2(t+1)^2}b \leq \delta_t + \frac{2L}{\mu^2}b\,.$$

Summing over t from $t^\star$ to T, we have

$$\delta_{T+1} \leq \delta_{t^\star} + \frac{2L(T-t^\star)}{\mu^2}b\,.$$

Then, we finally get

$$\begin{aligned}\Delta_{T+1} &\leq \frac{t^{\star 2}}{T^2}\left(1 - \frac{\mu}{L(1+a)}\right)^{t^\star}\Delta_1 + \frac{t^{\star 2}b}{2\mu(1+a)T^2} + \frac{2L(T-t^\star)}{\mu^2T^2}b \\ &\leq \frac{L^2(1+a)^2}{\mu^2T^2}\left(1 - \frac{\mu}{L(1+a)}\right)^{\frac{L(1+a)}{\mu}}\Delta_1 + \frac{L^2(1+a)b}{2\mu^3T^2} + \frac{2L}{\mu^2T}b\,.\end{aligned}$$

□

Algorithm 1 SGD with Cosine Stepsize and Restarts

Input: Initial Step size η_0, time increase factor r, initial point $\boldsymbol{x}_1$.
for $i = 0, \ldots, l$ **do**
  Let $T_i = T_0\, r^i$
  for $t = 0, \ldots, T_i - 1$ **do**
    Run SGD with cosine stepsize $\frac{\eta_0}{2}\left(1 + \cos\frac{t\pi}{T_i}\right)$
  end for
end for

A.2. Cosine stepsize with Restarts

Cosine stepsize is proposed with warm restarts (Loshchilov & Hutter, 2017). We then complete the theory by providing an analysis of SGD with cosine stepsize in this restarting scheme (Algorithm 1) under the PL condition. The proof builds on the fact that the suboptimality gap shrinks after each restarting and the rate depends on the suboptimality gap at the beginning of each restarting.

Theorem 6 (SGD with cosine step size and restart)**.** *Assume (**A1**, **A2**, **A3**). For a given T_0, $r > 1$, $T_i = T_0\, r^i$, $T \triangleq \sum_{i=0}^{l} T_i$, and $\eta_0 = (L(1+a))^{-1}$, Algorithm 1 guarantees (where $\tilde{O}$ hides the log terms)*

$$\mathbb{E}f(\boldsymbol{x}_T) - f^\star \leq \tilde{O}\left(\exp\left(-\frac{\mu(T-l-1)}{2L(1+a)}\right) + b\left(\frac{1}{\mu^{4/3}T^{4/3}} + \frac{1}{\mu^{5/3}T^{2/3}}\right)\right),$$

and for $r = 1$, it guarantees

$$\mathbb{E}f(\boldsymbol{x}_T) - f^\star \leq C_1\left(\frac{1}{\mu^{4/3}T_0^{4/3}} + \frac{1}{\mu^{5/3}T_0^{2/3}}\right)\frac{1-\exp\left(-C_2\mu(T-l-1)\right)}{1-\exp\left(-C_2\mu(T_0-1)\right)} + \exp\left(-\mu C_2(T-l-1)\right)\left(f(x_1) - f^\star\right),$$

where $C_1 \triangleq \frac{6^{5/3}\pi^4 b}{32(1+a)}$ and $C_2 \triangleq \frac{1}{2L(1+a)}$.

Proof of Theorem 6. Denote by $S_i = \sum_{j=0}^{i} T_j$ and $S_{-1} = 1$. Given Theorem 2, it is immediate to have $\forall i = 0, \ldots, l$:

$$\begin{aligned}\mathbb{E}f(\boldsymbol{x}_{S_i}) - f^\star &\leq \frac{\pi^4 b}{32(1+a)T_i^4}\left(\left(\frac{8T_i^2}{\mu}\right)^{4/3} + \left(\frac{6T_i^2}{\mu}\right)^{\frac{5}{3}}\right) + \exp\left(-\frac{\mu(T_i-1)}{2L(1+a)}\right)\left(f(x_{S_{i-1}}) - f^\star\right)\\ &\leq C_1\left(\frac{1}{\mu^{4/3}T_i^{4/3}} + \frac{1}{\mu^{5/3}T_i^{2/3}}\right) + \exp\left(-C_2\mu(T_i-1)\right)\left(f(x_{S_{i-1}}) - f^\star\right).\end{aligned}$$

Repeatedly using the above inequality, we get

$$\mathbb{E}f(\boldsymbol{x}_{S_l}) - f^\star \leq C_1\sum_{i=0}^{l}\prod_{j=i+1}^{l}\exp(-C_2\mu(T_j-1))\left(\frac{1}{\mu^{4/3}T_i^{4/3}} + \frac{1}{\mu^{5/3}T_i^{2/3}}\right) + \exp\left(-C_2\mu(S_l-l-1)\right)\left(f(x_1) - f^\star\right).$$

In the case of $r = 1$, $T_i = T_0$, we have for any i that

$$\begin{aligned}\sum_{i=0}^{l}\prod_{j=i+1}^{l}\exp(-C_2\mu(T_j-1))\left(\frac{1}{\mu^{4/3}T_i^{4/3}} + \frac{1}{\mu^{5/3}T_i^{2/3}}\right) &= \left(\frac{1}{\mu^{4/3}T_0^{4/3}} + \frac{1}{\mu^{5/3}T_0^{2/3}}\right)\sum_{i=0}^{l}\exp\left(-C_2\mu(T_0-1)(l-i)\right)\\ &= \left(\frac{1}{\mu^{4/3}T_0^{4/3}} + \frac{1}{\mu^{5/3}T_0^{2/3}}\right)\frac{1-\exp\left(-C_2\mu(T_0-1)(l+1)\right)}{1-\exp\left(-C_2\mu(T_0-1)\right)}.\end{aligned}$$

In the case of $r > 1$, denote by $A_i = \prod_{j=i+1}^{l}\exp(-C_2\mu(T_j-1))\frac{1}{\mu^p T_i^q}, p, q > 0$. For any $i = 0, ..., l$, $\frac{A_i}{A_{i+1}} =$

$\exp(-C_2\mu(T_{i+1}-1))r^q$ is decreasing over i. Denote by $i^\star = \min\{i : \frac{A_i}{A_{i+1}} \le 1\}$. We have

$$\sum_{i=0}^{l} A_i \le A_0 \cdot i^\star + (l - i^\star + 1) \cdot A_l \le (l+1) \cdot (A_0 + A_l) = \frac{l+1}{\mu^p} \cdot \left(\frac{1}{T_l^q} + \frac{1}{T_0^q} \exp(-C_2\mu(T - T_0 - l)) \right), \quad p, q > 0.$$

Note that $T_l = T \cdot \frac{r^l(r-1)}{r^{l+1}-1} \ge \frac{r-1}{r} T$ and $l = O(\ln T)$. Then, the stated bound follows. □

A.3. Proofs in Section 4

Proof of Lemma 1. By (1), we have

$$f(\boldsymbol{x}_{t+1}) \le f(\boldsymbol{x}_t) - \langle \nabla f(\boldsymbol{x}_t), \eta_t \boldsymbol{g}_t \rangle + \frac{L}{2} \eta_t^2 \|\boldsymbol{g}_t\|^2 .$$

Taking expectation on both sides, we get

$$\mathbb{E} f(\boldsymbol{x}_{t+1}) - \mathbb{E} f(\boldsymbol{x}_t) \le -\left(\eta_t - \frac{L(a+1)}{2} \eta_t^2 \right) \mathbb{E} \|\nabla f(\boldsymbol{x}_t)\|^2 + \frac{L}{2} \eta_t^2 b \le -\frac{1}{2} \eta_t \mathbb{E} \|\nabla f(\boldsymbol{x}_t)\|^2 + \frac{L}{2} \eta_t^2 b,$$

where in the last inequality we used the fact that $\eta_t \le \frac{1}{L(1+a)}$. □

Proof of Lemma 2. When $k = 1$, $X_2 \le A_1 X_1 + B_1$ satisfies. By induction, assume $X_k \le \prod_{i=1}^{k-1} A_i X_1 + \sum_{i=1}^{k-1} \prod_{j=i+1}^{k-1} A_j B_i$, and we have

$$\begin{aligned} X_{k+1} &\le A_k \left(\prod_{i=1}^{k-1} A_i X_1 + \sum_{i=1}^{k-1} \prod_{j=i+1}^{k-1} A_j B_i \right) + B_k = \prod_{i=1}^{k} A_i X_1 + \sum_{i=1}^{k-1} \prod_{j=i+1}^{k} A_j B_i + A_k B_k \\ &= \prod_{i=1}^{k} A_i X_1 + \sum_{i=1}^{k} \prod_{j=i+1}^{k} A_j B_i . \end{aligned}$$

□

Proof of Lemma 4. We have

$$\frac{\alpha^{T+1}}{(1-\alpha)} = \frac{\alpha\beta}{T(1-\alpha)} = \frac{\beta}{T\left(1 - \exp\left(-\frac{1}{T}\ln\frac{T}{\beta}\right)\right)} \le \frac{2\beta}{\ln\frac{T}{\beta}},$$

where in the last inequality we used $\exp(-x) \le 1 - \frac{x}{2}$ for $0 < x < \frac{1}{e}$ and the fact that $\frac{1}{T}\ln\left(\frac{T}{\beta}\right) \le \frac{\ln T}{T} \le \frac{1}{e}$. □

Proof of Lemma 3. If T is odd, we have

$$\sum_{t=1}^{T} \cos\frac{t\pi}{T} = \cos\frac{T\pi}{T} + \sum_{t=1}^{(T-1)/2} \cos\frac{t\pi}{T} + \cos\frac{(T-t)\pi}{T} = \cos\pi = -1,$$

where in the second inequality we used the fact that $\cos(\pi - x) = -\cos(x)$ for any x. If T is even, we have

$$\sum_{t=1}^{T} \cos\frac{t\pi}{T} = \cos\frac{T\pi}{T} + \cos\frac{T\pi}{2T} + \sum_{t=1}^{T/2-1} \cos\frac{t\pi}{T} + \cos\frac{(T-t)\pi}{T} = \cos\pi = -1 .$$

□

Proof of Lemma 5. It is enough to prove that $f(x) := x - 1 - \ln x \ge 0$. Observe that $f'(x)$ is increasing and $f'(1) = 0$, hence, we have $f(x) \ge f(1) = 0$. □

Proof of Lemma 6. Note that $f(t) = \exp(-bt)t^a$ is increasing for $t \in [0, a/b]$ and decreasing for $t \geq a/b$. Hence, we have

$$\begin{aligned}
\sum_{t=0}^{T} \exp(-bt)t^a &\leq \sum_{t=0}^{\lfloor a/b \rfloor - 1} \exp(-bt)t^a + \exp(-b\lfloor a/b \rfloor)\lfloor a/b \rfloor^a + \exp(-b\lceil a/b \rceil)\lceil a/b \rceil^a + \sum_{\lceil a/b \rceil + 1}^{T} \exp(-bt)t^a \\
&\leq 2\exp(-a)(a/b)^a + \int_0^{\lfloor a/b \rfloor} \exp(-bt)t^a dt + \int_{\lceil a/b \rceil}^{T} \exp(-bt)t^a dt \\
&\leq 2\exp(-a)(a/b)^a + \int_0^{T} \exp(-bt)t^a dt \\
&\leq 2\exp(-a)(a/b)^a + \int_0^{\infty} \exp(-bt)t^a dt \\
&= 2\exp(-a)(a/b)^a + \frac{1}{b^{a+1}}\Gamma(a+1)\,.
\end{aligned}$$

□

Proof of Theorem 3 and Theorem 4. We observe that for exponential step sizes,

$$\sum_{t=1}^{T} \eta_t^2 \leq \frac{\alpha^2}{L^2 c^2 (a+1)^2 (1-\alpha^2)}\,.$$

and for cosine step sizes,

$$\sum_{t=1}^{T} \eta_t^2 = \frac{\eta_0^2}{4} \sum_{t=1}^{T} \left(1 + \cos\frac{t\pi}{T}\right)^2 = \frac{\eta_0^2}{4} \sum_{t=0}^{T-1} \left(1 - \cos\frac{t\pi}{T}\right)^2 = \eta_0^2 \sum_{t=1}^{T} \sin^4\frac{t\pi}{2T} \leq \eta_0^2 \sum_{t=1}^{T} \frac{t^4\pi^4}{16T^4} \leq \frac{21\eta_0^2 T}{8\pi^4}\,.$$

Summing (4) over $t = 1, \ldots, T$ and dividing both sides by $\sum_{t=1}^{T} \eta_t$, we get the stated bound. □

A.4. Experiments details

A.4.1. Image classification experiments.

Data Normalization and Augmentation. Images are normalized per channel using the means and standard deviations computed from all training images. For CIFAR-10/100, we adopt the data augmentation technique following Lee et al. (2015) (for training only): 4 pixels are padded on each side of an image and a 32×32 crop is randomly sampled from the padded image or its horizontal flip.

Hyperparameter tuning. We tune the hyperparameters on the validation set using the following two-stage grid searching strategy. First, search over a coarse grid, and select the one yielding the best validation results. Next, continue searching in a fine grid centering at the best performing hyperparameters found in the coarse stage, and in turn, take the best one as the final choice.

For the starting step size η_0, the coarse searching grid is {0.00001, 0.0001, 0.001, 0.01, 0.1, 1}, and the fine grid is like {0.006, 0.008, 0.01, 0.02, 0.04} if the best one in the coarse stage is 0.01.

For the α value, we set its searching grid so that the ratio η_T/η_0, where η_T is the step size in the last iteration, is first searched over the coarse grid of {0.00001, 0.0001, 0.001, 0.01, 0.1, 1}, and then over a fine grid centered at the best one of the coarse stage. Note that we try all pairs of (η_0, α) from their respective searching grids.

For the stagewise step decay, to make the tuning process more thorough, we modify as follows the one employed in Section 6.1 (specifically on tuning SGD V1) of Yuan et al. (2019), where they first set two milestones and then tune the starting step size. Put it explicitly and take the experiment on CIFAR-10 as an example, we first run vanilla SGD with a constant step size to search for a good range of starting step size on the grid {0.00001, 0.0001, 0.001, 0.01, 0.1, 1}, and find 0.01 and 0.1 work well. Based on this, we set the fine searching grid of starting step sizes as {0.007, 0.01, 0.04, 0.07, 0.1, 0.4}. For each of them, we run three settings with an increasing number of milestones: vanilla SGD (with no milestone), SGD with 1 milestone, and SGD with 2 milestones. The searching grid for milestones is {16k, 24k, 32k, 40k, 48k, 56k} (number of iterations). For the 1 milestone setting, the milestone can be any of them. For the 2 milestones, they can be any combination

of two different elements from the searching grid, like (16k, 32k) or (32k, 48k). The grid search strategy for FashionMNIST and CIFAR-100 is similar but with the searching grid for milestones over {3k, 6k, 9k, 12k, 15k, 18k}.

The PyTorch ReduceLROnPlateau scheduler takes multiple arguments, among which we tune the starting learning rate, the factor argument which decides by which the learning rate will be reduced, the patience argument which controls the number of epochs with no improvement after which learning rate will be reduced, and the threshold argument which measures the new optimum to only focus on significant changes. We choose the searching grid for the starting step size using the same strategy for stagewise step decay above, i.e., first running SGD with a constant step size to search for a good starting step size, then search over a grid centering on the found value, which results in the grid {0.004, 0.007, 0.01, 0.04, 0.07} (FashionMNIST) and {0.01, 0.04, 0.07, 0.1, 0.4} (CIFAR10/100). We also explore the searching grid of the factor argument over {0.1, 0.5}, the patience argument over {5, 10} (CIFAR10) or {3, 6} (FashionMNIST/CIFAR100), and the threshold argument over {0.0001, 0.001, 0.01, 0.1}.

For each setting, we choose the combination of hyperparameters that gives the best final validation loss to be used in testing. Also, whenever the best performing hyperparameters lie in the boundary of the searching grid, we always extend the grid to make the final best-performing hyperparameters fall into the interior of the grid.

A.4.2. Natural language inference

Dataset We conduct this experiment on the Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) which contains 570k pairs of human-generated English sentences. Each pair of sentences is manually labeled with one of three categories: entailment, contradiction, and neutral, and thus forms a three-way classification problem. It captures the task of natural language inference, a.k.a. Recognizing Textual Entailment (RTE).

Model We employ the bi-directional LSTM of about 47M parameters proposed by Conneau et al. (2017). Except for replacing the cross-entropy loss with an SVM loss following Berrada et al. (2019), we leave all other components unchanged (codes can be found here[6]). Like them, we also use the open-source GloVe vectors (Pennington et al., 2014) trained on Common Crawl 840B with 300 dimensions as fixed word embeddings.

Training During the validation stage, we tune each method using the grid search. The initial learning rate of each method is grid searched over $\{0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10\}$. And the α of the exponential step size is searched over a grid such that the ratio η_T/η_0, where η_T is the step size in the last iteration, is over {0.0001, 0.001, 0.01, 0.1, 1}. Following (Berrada et al., 2019), for each hyperparameter setting, we record the best validation accuracy obtained during training and select the setting that performs the best according to this metric to do the test. The testing stage is repeated with different random seeds for 5 times to eliminate the influence of stochasticity.

We employ the Nesterov momentum (Nesterov, 1983) of 0.9 without dampening (if having this option), but do not use weight decay. The mini-batch size is 64 and we run for 10 epochs.

Results We compare the exponential and the cosine step sizes with Adagrad, Adam, AMSGrad (Reddi et al., 2018), BPGrad (Zhang et al., 2018), and DFW (Berrada et al., 2019). From Figure 4 and Table 2, we can see that cosine step size remains the best among all methods, with exponential step size following closely next.

Table 2. The best test accuracy achieved by each method. The $\pm$ shows 95% confidence intervals of the mean accuracy value over 5 runs starting from different random seeds.

Methods	Test Accuracy
Adam	0.8479 ± 0.0043
AdaGrad	0.8446 ± 0.0027
AMSGrad	0.8475 ± 0.0029
DFW	0.8412 ± 0.0045
BPGrad	0.8459 ± 0.0030
Exp. Step Size	0.8502 ± 0.0028
Cosine Step Size	$\mathbf{0.8509 \pm 0.0033}$

[6] https://github.com/oval-group/dfw

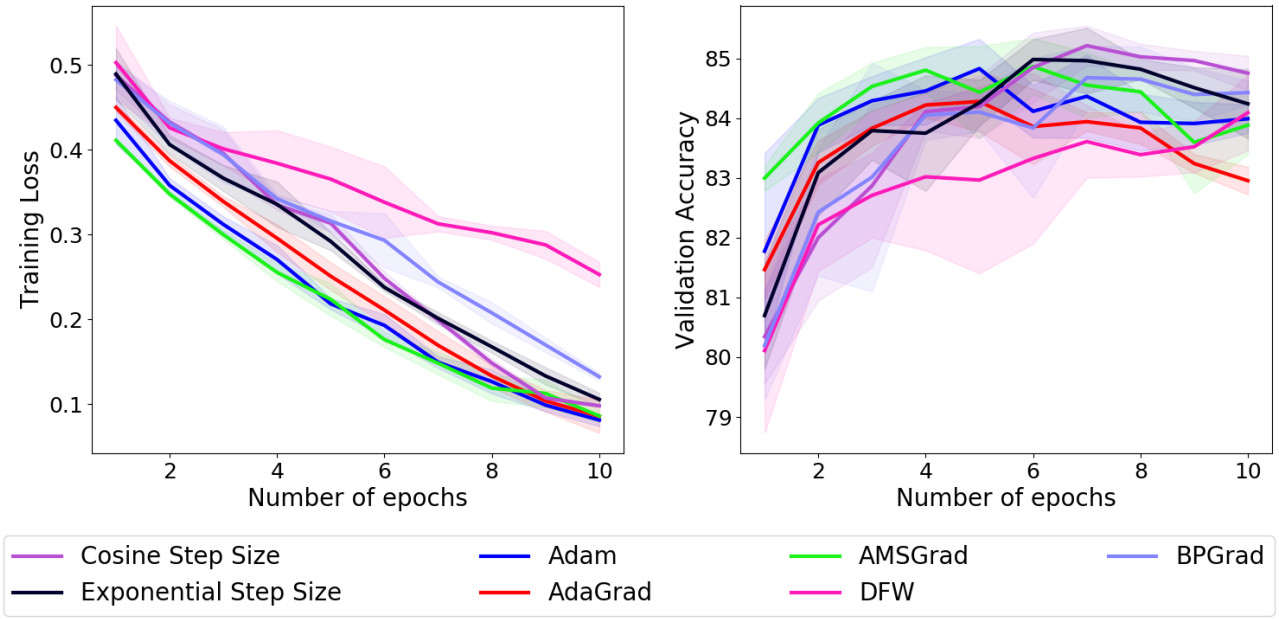

Figure 4. Training loss and validation accuracy curves, averaged over 5 independent runs, on using different methods to optimize a Bi-LSTM to do natural language inference on the SNLI dataset. *(The shading of each curve represents the 95% confidence interval computed across five independent runs from random initial starting points.)*

A.4.3. Synthetic Experiments

We have theoretically proved that exponential and cosine step size can adapt to the level of noise automatically and converge to the optimum without the need to re-tune the hyperparameters. In contrast, for other optimization methods, re-tuning is typically critical for convergence. To validate this empirically, we conduct an experiment on a non-convex function $g(r,\theta) = (2 + \frac{\cos\theta}{2} + \cos 4\theta) r^2 (5/3 - r)$ (Zhou et al., 2017), where r and θ are the polar coordinates, which satisfies the PL condition when $r \leq 1$.

Proof of PL condition. We now prove that $f(x,y) = g(r,\theta) = (2 + \frac{\cos\theta}{2} + \cos 4\theta) r^2 (5/3 - r)$ satisfies the PL condition when $r \leq 1$.

Obviously, $2 + \frac{\cos\theta}{2} + \cos 4\theta \geq \frac{1}{2}$ as $\cos\theta \in [-1, 1]$.

When $r \leq 1$, $\frac{5}{3} - r \geq \frac{2}{3}$, thus $f(x,y) \geq 0$, and $f^\star = f(0,0) = 0$.

We first calculate derivatives in polar coordinates

$$\frac{\partial g}{\partial r} = \left(\frac{10r}{3} - 3r^2\right)\left(2 + \frac{\cos\theta}{2} + \cos 4\theta\right),$$
$$\frac{\partial g}{\partial \theta} = \left(-\frac{\sin\theta}{2} - 4\sin 4\theta\right) r^2 \left(\frac{5}{3} - r\right).$$

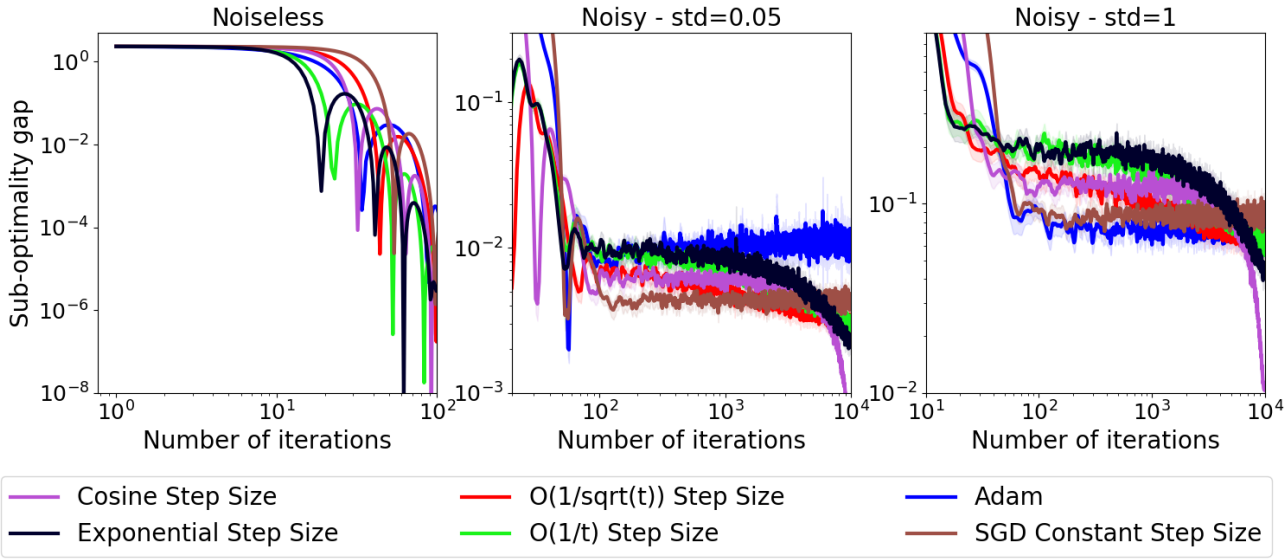

Figure 5. Plots of the sub-optimality gap vs. iterations for optimizing a synthetic function. Both axes in all figures are on the logarithmic scale. The left plot is the noiseless case, the middle one is with the additive Gaussian noise of standard deviation 0.05, while the right plot is with the additive Gaussian noise of standard deviation 1. *(The shading of each curve represents the 95% confidence interval computed across five independent runs from random initial starting points.)*

Then, from the relationship between derivatives in Cartesian and polar coordinates, we have

$$\begin{aligned}\frac{\|\nabla f(x,y)\|^2}{2(f(x,y)-f^\star)} &= \frac{\left(\frac{\partial g}{\partial r}\right)^2 + \frac{1}{r^2}\left(\frac{\partial g}{\partial \theta}\right)^2}{2(2+\frac{\cos\theta}{2}+\cos 4\theta)r^2(\frac{5}{3}-r)} \\ &= \frac{(\frac{10}{3}-3r)^2(2+\frac{\cos\theta}{2}+\cos 4\theta)}{\frac{10}{3}-2r} + \frac{(-\frac{\sin\theta}{2}-4\sin 4\theta)^2(\frac{5}{3}-r)}{2(2+\frac{\cos\theta}{2}+\cos 4\theta)} \\ &\geq \frac{(\frac{10}{3}-3r)^2}{4(\frac{5}{3}-r)} \geq \frac{1}{24}\,.\end{aligned}$$

We compare SGD with decay rules listed in (5), SGD with constant step size, and Adam on optimizing this function. We consider three cases: the noiseless case where we get the exact gradient in each round, the slightly noisy case in which we add independent Gaussian noise with zero mean and standard deviation 0.05 to each dimension of the gradient in each round, and the noisy case with additive Gaussian noise of standard deviation 1.

We tune hyperparameters for all methods on the noiseless case such that they all obtain roughly the same performance after 100 iterations. We then apply those methods directly to the two noisy cases using the same set of hyperparameters obtained in the noiseless case. To reduce the influence of stochasticity, we average on 100 independent runs with different random seeds. Results shown in Figure 5 demonstrate that both exponential and cosine step size behave as the theory predicts and has no problem on converging towards the optimum when the noise level changes. In contrast, after the initial decrease in the sub-optimality gap, other methods all end up oscillating around some value which is method-specific and related to the noise level.