Lecture 7: Gradient descent continued, and stochastic gradient descent

CS4787/5777 — Principles of Large-Scale Machine Learning Systems

- (continued) Gradient descent
- Stochastic Gradient descent

In [1]:
```
import numpy
import scipy
import matplotlib
from matplotlib import pyplot
import time
```

Recall: from last time, we showed that...

If we assume that for all $x, y \in \mathrm{R}^d$, $\|\nabla f(x) - \nabla f(y)\| \leq L \cdot \|x - y\|$, then

$$f(w_{t+1}) \leq f(w_t) - \alpha\left(1 - \frac{1}{2}\alpha L\right) \cdot \|\nabla f(w_t)\|^2.$$

If we choose our step size α to be *small enough* that $1 \geq \alpha L$, then this simplifies to

$$f(w_{t+1}) \leq f(w_t) - \frac{\alpha}{2} \cdot \|\nabla f(w_t)\|^2.$$

That is, **the objective is guaranteed to decrease at each iteration**! This matches our intuition for why gradient descent should work.

An aside...why do we need sufficiently small step size?

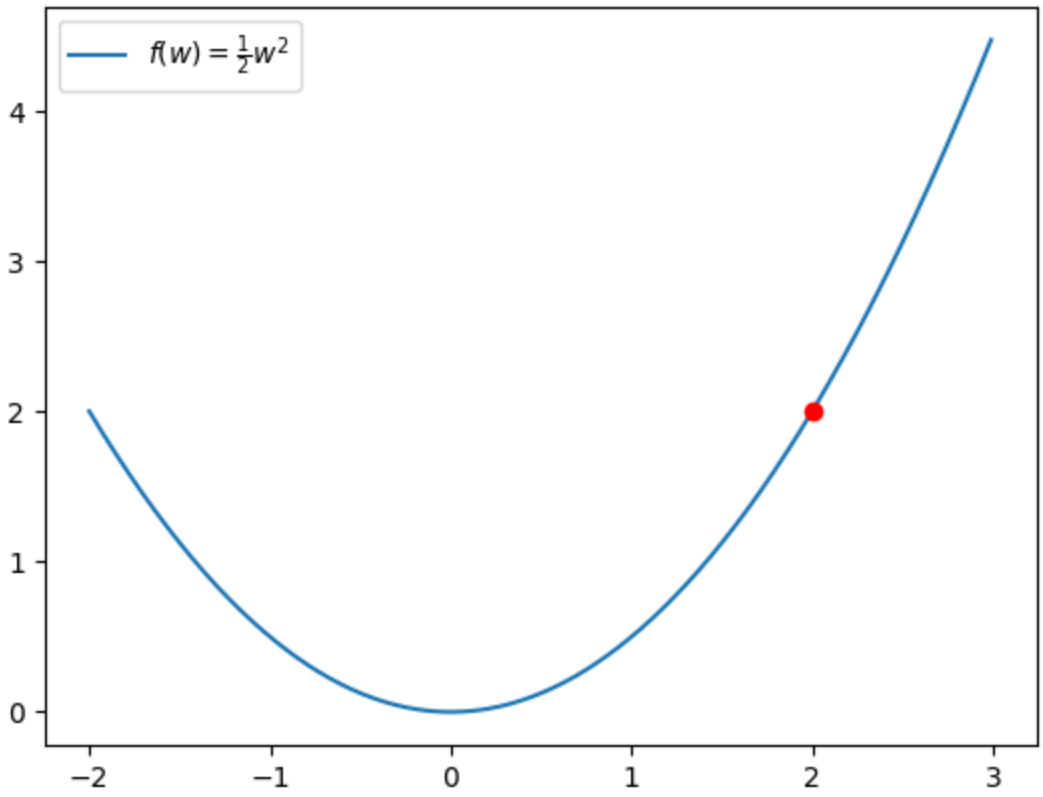
With too large a step size, we can overshoot the optimum!

Consider the following case of $f(w) = \frac{1}{2}w^2$. Here $f'(w) = \nabla f(w) = w$ (and $f''(w) = 1$), so it's L-smooth with $L = 1$. Suppose we're at $w_t = 2$ as shown here:

In [2]:
```
%matplotlib inline

x = numpy.arange(-2,3,0.01)
y = x**2 / 2
pyplot.plot(x,y, label=r"$f(w) = \frac{1}{2}w^2$");
pyplot.scatter([2.0],[2.0**2/2], c="r", zorder=10)
pyplot.legend()
```

Out[2]: <matplotlib.legend.Legend at 0x12f9e4fd0>



Here our GD update will be $w_{t+1} = w_t - \alpha f(w_t) = 2 - \alpha \cdot 2$. If we step with $\alpha = 1$, we go directly to the optimum at $w = 0$. But! If we step with a larger α, we overshoot, and **for $\alpha > 2$, our loss $f(w)$ actually increases**. This illustrates why having *sufficiently small steps* is necessary for our proof.

Why is our L-smoothness assumption necessary?

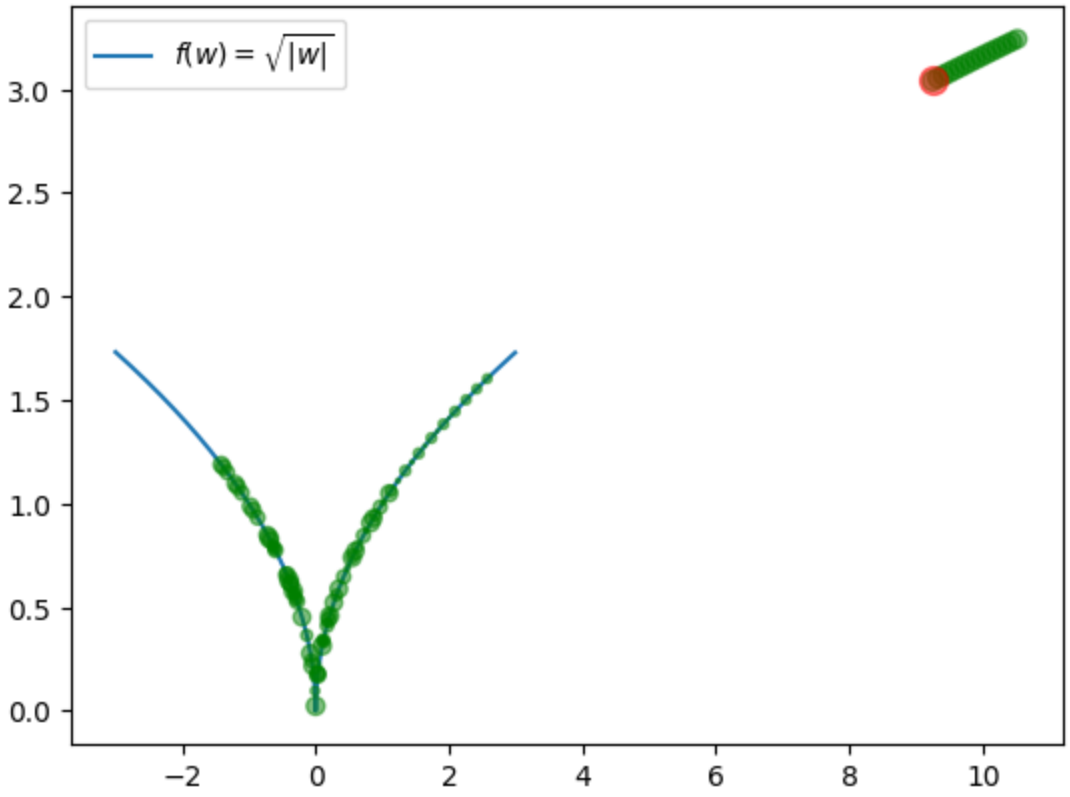
We can also use a simple example to illustrate why assuming some sort of smoothness is necessary to show this result. Consider the (dumb) example of $f(w) = \sqrt{|w|}$. It is differentiable everywhere except at $w = 0$.

In [3]:
```
x = numpy.arange(-3,3,0.01)
y = numpy.sqrt(numpy.abs(x))
w = 2;
prev_w = numpy.array(w)
alpha = 0.5;

for it in range(100):
    w = w - alpha * numpy.sign(w)/(2*numpy.sqrt(numpy.abs(w)))
    prev_w = numpy.append(prev_w,w)
```

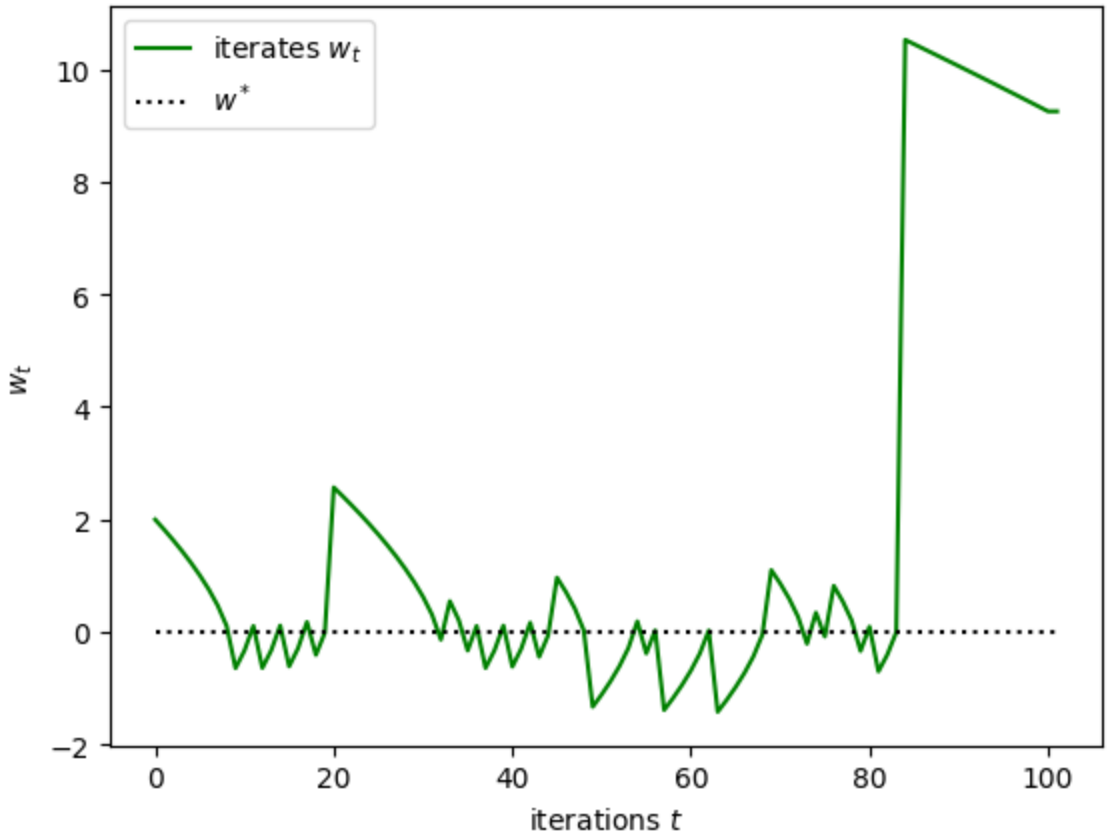
In [4]:
```
pyplot.plot(x,y, label=r"$f(w) = \sqrt{|w|}$");
pyplot.scatter(prev_w, numpy.sqrt(numpy.abs(prev_w)), s=list(numpy.arange(1,10
pyplot.scatter([w], [numpy.sqrt(numpy.abs(w))], [100], c="r", zorder=100, alpha
pyplot.legend()
```

Out[4]: <matplotlib.legend.Legend at 0x12fb31090>



In [5]:
```
prev_w = numpy.append(prev_w,w)
pyplot.plot(prev_w, c="g", label=r"iterates $w_t$")
pyplot.plot(numpy.arange(len(prev_w)), numpy.zeros(len(prev_w)), c="black", li
pyplot.xlabel(r"iterations $t$")
pyplot.ylabel(r"$w_t$")
pyplot.legend()
```

Out[5]:
```
<matplotlib.legend.Legend at 0x12fd8a950>
```



- Gradients around $w = 0$ changes too fast! For $w > 0, f(w) = \sqrt{w}$ with $f'(w) = \frac{1}{2\sqrt{w}}$, $|f''(w)| = \frac{1}{4w^{3/2}}$.
- As $w \downarrow 0$, $|f''(w)| \to \infty$ so there is no global upper bound for the second derivative, which means we do not have Lipschitz continuous gradients:

$$\|f'(y) - f'(x)\| \leq L\|y - x\|$$

for all x, y

- GD diverged once we got close to 0

Questions?

Stochastic Gradient Descent

Basic idea: **in gradient descent, just replace the full gradient (which is a sum) with a single gradient example**. Initialize the parameters at some value $w_0 \in \mathrm{R}^d$, and decrease the value of the empirical risk iteratively by sampling a random example $x_{(t)}$ uniformly from the training set and then updating

$$w_{t+1} = w_t - \alpha_t \cdot \nabla f(w_t; x_{(t)})$$

where as usual w_t is the value of the parameter vector at time t, α_t is the *learning rate* or *step size*, and ∇f_i denotes the gradient of the loss function of the ith training example. Compared

with gradient descent and Newton's method, SGD is simple to implement and runs each iteration faster.

A potential objection!

This is not necessarily going to be decreasing the loss at every step!

- Because we're just moving in a direction that will decrease the loss *for one particular example*: this won't necessarily decrease the total loss!
- So we can't demonstrate convergence by using a proof like the one we used for gradient descent, where we showed that the loss decreases at every iteration of the algorithm.
- The fact that SGD doesn't always improve the loss at each iteration motivates the question: **does SGD even work? And if so, why does SGD work?**

Question: Why might it be fine to get an approximate solution to an optimization problem for training?

Question: Why might it be fine to get an approximate solution to an optimization problem for training?

Because we don't always have to minimize the training loss exactly, as long as the performance on the test set is good enough. In machine learning, generalization matters more than optimization, sadly (for an optimizer).

Understanding why SGD converges

Assumption: f is L-smooth, i.e. for any x and y

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|.$$

Assumption: loss f is non-negative. (This is without loss of generality if f is bounded from below, since we can always add a constant to f in that case to make it non-negative.)

New Assumption for SGD: the variance of the gradients is bounded. For some constant $\sigma > 0$, if x is drawn uniformly at random from the training set, for any w

$$\mathbf{E}_x\left[\|\nabla f(w; x) - \nabla f(w)\|^2\right] \leq \sigma^2,$$

or equivalently,

$$\frac{1}{n}\sum_{x \in \mathrm{D}} \|\nabla f(w; x) - \nabla f(w)\|^2 \leq \sigma^2,$$

or also equivalently,

$$\mathbf{E}_x\left[\|\nabla f(w;x)\|^2\right] \leq \|\nabla f(w)\|^2 + \sigma^2.$$

Let's derive the quadratic upper bound again (the "descent" lemma)

that is, for all x, y,

$$f(y) \leq f(x) + \langle \nabla f(x),\, y - x\rangle + \frac{L}{2}\|y - x\|^2$$

To begin, consider the univariate function and its derivative

$$\rho(\tau) = f(x + \tau(y - x))$$
$$\rho'(\tau) = \langle \nabla f(x + \tau(y - x)),\, y - x\rangle$$

By the **Fundamental Theorem of Calculus**,

$$\rho(1) - \rho(0) = \int_0^1 \rho'(\tau)\, d\tau$$

If we substitute back in our definition of ρ, we get

$$f(y) - f(x) = \int_0^1 \langle \nabla f(x + \tau(y - x)),\, y - x > \, d\tau$$
$$f(y) = f(x) + \langle \nabla f(x),\, y - x\rangle + \int_0^1 \langle \nabla f(x + \tau(y - x)) - \nabla f(x),\, y - x > \, d\tau$$

where in the last step we rearranged, followed by an add and subtract of $\langle \nabla f(x),\, y - x\rangle$.

We can now simplify this as follows. Keep in mind: we assume that for all x, y, $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$.

$$\begin{aligned} f(y) &= f(x) + \langle \nabla f(x),\, y - x\rangle + \int_0^1 \langle \nabla f(x + \tau(y - x)) - \nabla f(x),\, y - x > \, d\tau \\ &\leq f(x) + \langle \nabla f(x),\, y - x\rangle + \int_0^1 \|\nabla f(x + \tau(y - x)) - \nabla f(x)\|\|y - x\|\, d\tau \\ &\leq f(x) + \langle \nabla f(x),\, y - x\rangle + \int_0^1 L\|x + \tau(y - x) - x\|\|y - x\|\, d\tau \\ &= f(x) + \langle \nabla f(x),\, y - x\rangle + L\|y - x\|^2 \int_0^1 \tau\, d\tau \end{aligned}$$

where the first inequality comes from Cauchy-Schwarz inequality $a^\top b \leq \|a\|\|b\|$, and the second comes from our L-Lipschitz continuous gradient assumption. which gives us the **"descent" lemma** (although, for SGD, we no longer guarantee descent)

$$f(y) \leq f(x) + \langle \nabla f(x),\, y - x\rangle + \frac{L}{2}\|y - x\|^2.$$

Questions?

$$f(y) \leq f(x) + \langle \nabla f(x),\ y - x \rangle + \frac{L}{2}\|y - x\|^2.$$

Since it holds for all x, y, we can plug in w_t for x and w_{t+1} for y, and plug in the **SGD update**

$$w_{t+1} = w_t - \alpha_t \nabla f(w_t; x_{(t)})$$

we have

$$\begin{aligned} f(w_{t+1}) &\leq f(w_t) + \langle \nabla f(w_t),\ w_{t+1} - w_t \rangle + \frac{L}{2}\|w_{t+1} - w_t\|^2 \\ &= f(w_t) - \alpha_t \langle \nabla f(w_t), f(w_t; x_{(t)}) \rangle + \frac{\alpha_t^2 L}{2}\|\nabla f(w_t; x_{(t)})\|^2 \end{aligned}$$

But the inner product term can be positive or negative, so we are not guaranteed to decrease the value of f after each step!

$$f(w_{t+1}) \leq f(w_t) - \alpha_t \left\langle \nabla f(w_t), f(w_t; x_{(t)}) \right\rangle + \frac{\alpha_t^2 L}{2}\|\nabla f(w_t; x_{(t)})\|^2.$$

Although this isn't a descent method, but this is going to tend to be decaying in expectation. Let's take the expectation of both sides conditioned on w_t. The randomness here is over the random choice of $x_{(t)}$.

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq \mathbf{E}\left[f(w_t) - \alpha_t \left\langle \nabla f(w_t), f(w_t; x_{(t)}) \right\rangle + \frac{\alpha_t^2 L}{2}\|\nabla f(w_t; x_{(t)})\|^2 \,\middle|\, w_t \right]$$

By linearity of expectation and by factoring out constants that don't depend on the random choice of $x_{(t)}$,

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \left\langle \mathbf{E}[\nabla f(w_t; x_{(t)}) \mid w_t],\ \nabla f(w_t) \right\rangle + \frac{\alpha_t^2 L}{2}\mathbf{E}\left[\|\nabla f(w_t; x_{(t)})\|^2 \,\middle|\, w_t \right].$$

Pause

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \left\langle \mathbf{E}[\nabla f(w_t; x_{(t)}) \mid w_t],\ \nabla f(w_t) \right\rangle + \frac{\alpha_t^2 L}{2}\mathbf{E}\left[\|\nabla f(w_t; x_{(t)})\|^2 \,\middle|\, w_t \right]$$

Now, the first conditional expectation is

$$\mathbf{E}[\nabla f(w_t; x_{(t)}) \mid w_t] = \frac{1}{n} \sum_{x \in \mathrm{D}} \nabla f(w_t; x) = \nabla f(w_t)$$

Althought w_t depends on the randomness in all previous iterations, because of the conditioning on w_t, the only randomness here is the selection of the example $x_{(t)}$, which is uniformly at random from the training set.

So we get

$$\begin{aligned}\mathbf{E}[f(w_{t+1}) \mid w_t] &\leq f(w_t) - \alpha_t \left\langle \nabla f(w_t),\ \nabla f(w_t) \right\rangle + \frac{\alpha_t^2 L}{2} \mathbf{E}\left[\|\nabla f(w_t; x_{(t)})\|^2 \;\middle|\; w_t \right] \\ &= f(w_t) - \alpha_t \|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 L}{2} \mathbf{E}\left[\|\nabla f(w_t; x_{(t)})\|^2 \;\middle|\; w_t \right]\end{aligned}$$

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 L}{2} \mathbf{E}\left[\|\nabla f(w_t; x_{(t)})\|^2 \;\middle|\; w_t \right]$$

Recall our bounded variance assumption of the stochastic gradients: for all w,

$$\mathbf{E}_x\left[\|\nabla f(w; x)\|^2 \right] \leq \|\nabla f(w)\|^2 + \sigma^2.$$

Applying this to the second expecation gives us

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 L}{2} \left(\|\nabla f(w_t)\|^2 + \sigma^2 \right).$$

This simplifies to

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \left(1 - \frac{\alpha_t L}{2} \right) \|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 \sigma^2 L}{2}.$$

Pause

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \alpha_t \left(1 - \frac{\alpha_t L}{2} \right) \|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 \sigma^2 L}{2}.$$

If we constrain the step size to be sufficiently small, $\alpha_t L \leq 1$, then this can be further simplified to

$$\mathbf{E}[f(w_{t+1}) \mid w_t] \leq f(w_t) - \frac{\alpha_t}{2}\|\nabla f(w_t)\|^2 + \frac{\alpha_t^2 \sigma^2 L}{2}.$$

Let's call this the **progress bound** of SGD

Finally, let's take the full expected value over **all** the randomness in the algorithm, not just this conditional expectation. By the law of total expectation, this yields

$$\begin{aligned}\mathbf{E}[f(w_{t+1})] &= \mathbf{E}[\mathbf{E}[f(w_{t+1}) \mid w_t]] \\ &\leq \mathbf{E}[f(w_t)] - \frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] + \frac{\alpha_t^2 \sigma^2 L}{2}\end{aligned}$$

Pause

Rearranging the above expression, we can get

$$\frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] \leq \mathbf{E}[f(w_t)] - \mathbf{E}[f(w_{t+1})] + \frac{\alpha_t^2 \sigma^2 L}{2}.$$

Now, let's imagine that we run K total iterations of SGD, and we sum both sides of this expression going from t from 0 to $K-1$. This yields

$$\sum_{t=0}^{K-1} \frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] \leq \left(\sum_{t=0}^{K-1} \left(\mathbf{E}[f(w_t)] - \mathbf{E}[f(w_{t+1})]\right)\right) + \frac{\sigma^2 L}{2}\sum_{t=0}^{K-1} \alpha_t^2.$$

$$\sum_{t=0}^{K-1} \frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] \leq \left(\sum_{t=0}^{K-1} \left(\mathbf{E}[f(w_t)] - \mathbf{E}[f(w_{t+1})]\right)\right) + \frac{\sigma^2 L}{2}\sum_{t=0}^{K-1} \alpha_t^2.$$

Observe that this sum **telescopes**!

So we get

$$\begin{aligned}\sum_{t=0}^{K-1} \left(\mathbf{E}[f(w_t)] - \mathbf{E}[f(w_{t+1})]\right) &= \mathbf{E}[f(w_0)] - \mathbf{E}[f(w_K)] \\ &= f(w_0) - \mathbf{E}[f(w_K)] \\ &\leq f(w_0),\end{aligned}$$

where this last inequality follows from our assumption that the loss is non-negative. Thus,

$$\sum_{t=0}^{K-1} \frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2}\sum_{t=0}^{K-1} \alpha_t^2.$$

$$\sum_{t=0}^{K-1} \frac{\alpha_t}{2} \mathbf{E}[\|\nabla f(w_t)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2} \sum_{t=0}^{K-1} \alpha_t^2.$$

Now, let

$$\rho_t = \frac{\alpha_t}{\sum_{k=0}^{K-1} \alpha_t}.$$

Multiply by $1 = \frac{\sum_{k=0}^{K-1} \alpha_t}{\sum_{k=0}^{K-1} \alpha_t}$ on the LHS, we have

$$\left(\sum_{t=0}^{K-1} \frac{\alpha_t}{2} \right) \sum_{t=0}^{K-1} \rho_t \mathbf{E}[\|\nabla f(w_t)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2} \sum_{t=0}^{K-1} \alpha_t^2.$$

$$\left(\sum_{t=0}^{K-1} \frac{\alpha_t}{2} \right) \sum_{t=0}^{K-1} \rho_t \mathbf{E}[\|\nabla f(w_t)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2} \sum_{t=0}^{K-1} \alpha_t^2.$$

But if we let τ be a random varible with distribution given by ρ (this is a distribution because it sums to 1), then we can write this left side in expected-value form as

$$\left(\sum_{t=0}^{K-1} \frac{\alpha_t}{2} \right) \cdot \mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2} \sum_{t=0}^{K-1} \alpha_t^2,$$

where now the expected value is taken over both the algorthmic randomness and the choice of τ. We can think of w_τ as the output of SGD run for a random number of steps.

Questions?

Constant Step Size

Here, we have $\alpha_t = \alpha \leq 1/L$

$$\left(\sum_{t=0}^{K-1} \frac{\alpha_t}{2} \right) \cdot \mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq f(w_0) + \frac{\sigma^2 L}{2} \sum_{t=0}^{K-1} \alpha_t^2$$

$$\frac{\alpha K}{2} \cdot \mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq f(w_0) + \frac{\alpha^2 \sigma^2 L K}{2}$$

Multiplying both sides by $\frac{2}{\alpha K}$,

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L$$

How should we interpret this?

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L$$

SGD with constant step size converges to a "noise ball" (a.k.a. "noise floor")! Gradient magnitude doesn't necessarily go to zero

Even if we run for a very large number of iterations,

$$\lim_{K\to\infty} \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L = \alpha\sigma^2 L \neq 0.$$

For many applications this is fine...but it seems somehow lacking.

How should we interpret this?

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L$$

(Figure shamelessly stolen from Mark Schmidt)

Question: how can we make the noise ball smaller?

We can make this arbitrarily small

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L$$

(Figure shamelessly stolen from Mark Schmidt)

We can make this arbitrarily small

By choosing the step size constant as a function of the number of steps K we plan to run. If we minize the RHS of

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{\alpha K} + \alpha\sigma^2 L$$

with respect to α, we will get

$$\alpha = \min\left(\sqrt{\frac{2f(w_0)}{\sigma^2 LK}}, \frac{1}{L}\right)$$

where the $1/L$ came from the earlier requirement of $\alpha L \leq 1$ in order to get to this point (the **progress bound**).

Substitute this in,

$$\mathbf{E}[\|\nabla f(w_\tau)\|^2] \leq \frac{2f(w_0)}{K} \max\left(\sqrt{\frac{\sigma^2 LK}{2f(w_0)}}, L\right) + \sigma^2 L \min\left(\sqrt{\frac{2f(w_0)}{\sigma^2 LK}}, \frac{1}{L}\right)$$

Then since $\min(x, y) \leq x$ and $\max(x, y) \leq x + y$,

$$\begin{aligned}
\mathbf{E}[\|\nabla f(w_\tau)\|^2] &\leq \frac{2f(w_0)}{K}\left(\sqrt{\frac{\sigma^2 LK}{2f(w_0)}} + L\right) + \sigma^2 L \cdot \frac{2f(w_0)}{\sigma^2 LK} \\
&= \sqrt{\frac{2f(w_0)\sigma^2 L}{K}} + \frac{2Lf(w_0)}{K} + \sqrt{\frac{2f(w_0)\sigma^2 L}{K}} \\
&= \sqrt{\frac{8f(w_0)\sigma^2 L}{K}} + \frac{2Lf(w_0)}{K}.
\end{aligned}$$

This *is* decreasing to zero as $K \to \infty$.

Questions?

Under strong convexity, we can do better

Recall the **progress bound** of SGD

$$\mathbf{E}[f(w_{t+1})] \leq \mathbf{E}[f(w_t)] - \frac{\alpha_t}{2}\mathbf{E}[\|\nabla f(w_t)\|^2] + \frac{\alpha_t^2 \sigma^2 L}{2}.$$

If f is μ-strongly convex, we have that $\|\nabla f(x)\|^2 \geq 2\mu\left(f(x) - f^*\right)$, and so

$$\mathbf{E}[f(w_{t+1})] \leq \mathbf{E}[f(w_t)] - \frac{\alpha_t}{2}\mathbf{E}[2\mu\left(f(w_t) - f^*\right)] + \frac{\alpha_t^2 \sigma^2 L}{2}$$

$$\mathbf{E}[f(w_{t+1})] \leq \mathbf{E}[f(w_t)] - \frac{\alpha_t}{2}\mathbf{E}[2\mu\Big(f(w_t) - f^*\Big)] + \frac{\alpha_t^2\sigma^2 L}{2}$$

Subtracting the global minimum f^* from both sides,

$$\mathbf{E}[f(w_{t+1}) - f^*] \leq \mathbf{E}[f(w_t) - f^*] - \alpha_t\mu\mathbf{E}[f(w_t) - f^*] + \frac{\alpha_t^2\sigma^2 L}{2},$$

which simplifies to

$$\mathbf{E}[f(w_{t+1}) - f^*] \leq (1 - \alpha_t\mu)\mathbf{E}[f(w_t) - f^*] + \frac{\alpha_t^2\sigma^2 L}{2}.$$

Again supposing a constant step size,

$$\mathbf{E}[f(w_{t+1}) - f^*] \leq (1 - \alpha\mu) \cdot \mathbf{E}[f(w_t) - f^*] + \frac{\alpha^2\sigma^2 L}{2}.$$

Subtracting $\frac{\alpha\sigma^2 L}{2\mu}$ from both sides,

$$\begin{aligned}\mathbf{E}\left[f(w_{t+1}) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right] &\leq (1 - \alpha\mu) \cdot \mathbf{E}[f(w_t) - f^*] + \frac{\alpha^2\sigma^2 L}{2} - \frac{\alpha\sigma^2 L}{2\mu} \\ &\leq (1 - \alpha\mu) \cdot \mathbf{E}\left[f(w_t) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right].\end{aligned}$$

$$\mathbf{E}\left[f(w_{t+1}) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right] \leq (1 - \alpha\mu) \cdot \mathbf{E}\left[f(w_t) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right].$$

We can now apply this recursively! Over K total iterations,

$$\begin{aligned}\mathbf{E}\left[f(w_K) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right] &\leq (1 - \alpha\mu)^K \cdot \left(f(w_0) - f^* - \frac{\alpha\sigma^2 L}{2\mu}\right) \\ &\leq \exp(-\alpha\mu K) \cdot \Big(f(w_0) - f^*\Big)\end{aligned}$$

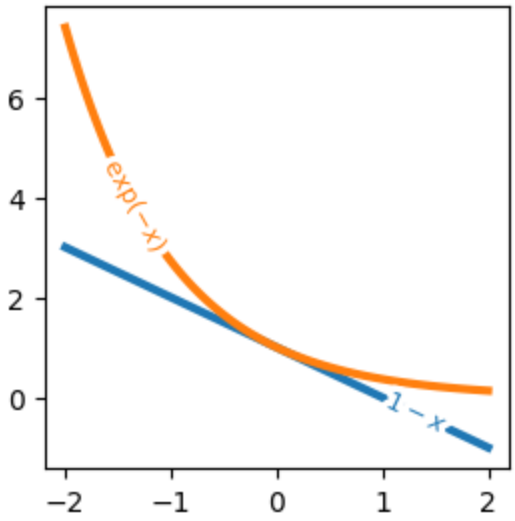
where we were able to drop the expected value on the right because w_0 is not a random variable.

Note that we again used the inequality $(1 - x) \leq \exp(-x)$ for all x,

In [7]:
```
from labellines import labelLine, labelLines
x = numpy.linspace(-2,2,100); y = 1-x; z = numpy.exp(-x)
```

```
fig = pyplot.figure(figsize=(3,3)); ax = fig.add_subplot(111)

ax.plot(x,y, linewidth=3, label=r"$1-x$");ax.plot(x,z, linewidth=3, label=r"$\
```



This yields a final bound of

$$\mathbf{E}\left[f(w_K) - f^*\right] \leq \exp(-\alpha\mu K) \cdot \left(f(w_0) - f^*\right) + \frac{\alpha\sigma^2 L}{2\mu}$$

$$\mathbf{E}\left[f(w_K) - f^*\right] \leq \exp(-\alpha\mu K) \cdot \left(f(w_0) - f^*\right) + \frac{\alpha\sigma^2 L}{2\mu}$$

Again we have the same issue of convergence to a noise ball for constant α. We can minimize this over α to pick a number-of-steps-dependent step size.

$$0 = -\mu K \exp(-\alpha\mu K) \cdot \left(f(w_0) - f^*\right) + \frac{\sigma^2 L}{2\mu} \rightarrow \alpha = \frac{1}{\mu K} \log\left(\frac{2\mu^2\left(f(w_0) - f^*\right)K}{\sigma^2 L}\right).$$

This yields

$$\begin{aligned}\mathbf{E}\left[f(w_K) - f^*\right] &\leq \frac{\sigma^2 L}{2\mu^2\left(f(w_0) - f^*\right)K} \cdot \left(f(w_0) - f^*\right) + \frac{\sigma^2 L}{2\mu} \cdot \frac{1}{\mu K} \log\left(\frac{2\mu^2\left(f(w_0) - f^*\right)K}{\sigma^2 L}\right) \\ &= \frac{\sigma^2 L}{2\mu^2 K} + \frac{\sigma^2 L}{2\mu^2 K} \log\left(\frac{2\mu^2\left(f(w_0) - f^*\right)K}{\sigma^2 L}\right) \\ &= \frac{\sigma^2 L}{2\mu^2 K} \log\left(\frac{2e\mu^2\left(f(w_0) - f^*\right)K}{\sigma^2 L}\right).\end{aligned}$$

This is indeed approaching 0 as K becomes large!

Final questions?