Solution (More Detailed Sketches)
Math 389L
Problem Set 2
Tuesday, February 12, 2019

Note: This problem set is about linear regression. In particular, we hope to show why traditional methods for solving linear regression problems don't work well as the number of data points n in a data set becomes very large. Two methods (traditional gradient descent and stochastic gradient descent) are proposed to solve these issues and you will analyze their issues. We will talk about more recent developments in the literature on this extremely common problem (matrix sketching and leverage score sampling) in class.

1 Given a data matrix $\boldsymbol{A} \in \mathbb{R}^{n\times p}$ and an output vector $\boldsymbol{b}$, the least squares problem asks us to find the vector $\boldsymbol{x}$ which minimizes $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2$. Here we will ask you to analyze the performance of gradient descent applied to solving this problem. For reference, note that exact, stable solutions to the least squares problem take $2np^2 - \frac{2}{3}p^3$ floating point operations [GolubVanLoan, Algorithm 5.3.2], though these solutions in general don't take advantage of sparsity in the data matrix $\boldsymbol{A}$.

(a) Show that the gradient $\nabla f(\boldsymbol{x}) = \boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})$ and the Hessian $\nabla^2 f(\boldsymbol{x}) = \boldsymbol{A}^*\boldsymbol{A}$. Conclude that f is $\sigma_{\min}(\boldsymbol{A})$-strongly convex.

(b) How many floating point operations does it take to compute $\nabla f(\boldsymbol{x})$ for an arbitrary input $\boldsymbol{x} \in \mathbb{R}^p$ if we multiply matrices in the traditional way? Write a bound both in terms of the dimensions n, p, as well as a different bound in terms of the number of non-zero entries in $\boldsymbol{A}$ (which we denote $\text{nnz}(\boldsymbol{A})$)

(c) Give an upper bound on the number of floating point operations it takes to compute an approximate least-squares solution $\tilde{\boldsymbol{x}}$ with $\|\tilde{\boldsymbol{x}} - \boldsymbol{x}\| \leq \epsilon$ via gradient descent which depends on the actual data $\boldsymbol{A}$ and $\boldsymbol{b}$ only through the condition number $\kappa = \sigma_{\max}(\boldsymbol{A})/\sigma_{\min}(\boldsymbol{A})$, $\sigma_{\min}(\boldsymbol{A})$, and $\|\boldsymbol{A}^*\boldsymbol{b}\|_2$. Give another bound in terms of $\text{nnz}(\boldsymbol{A})$ which is tighter for sparse data matrices. Assume that we start the algorithm at the zero vector $\boldsymbol{x}_0 = 0$, and utilize the 'optimal' learning rate to minimize your bounds.

(d) Detail when you would, and would not, want to use this gradient-descent based approach to solving least squares problems over the traditional exact approach.

(c) Use Problem 3 from the previous problem set. Write $k = \frac{\kappa-1}{\kappa+1}$ and $\ell = \sigma_{\min}(\boldsymbol{A})$. You should get something like

$$\text{flops} \sim 4np \log_{1/q}\left(\frac{\|\boldsymbol{A}^*\boldsymbol{b}\|_2}{\ell\epsilon}\right).$$

■

2 Observe that we can find the least squares solution $\boldsymbol{x}$ from Problem 1 by minimizing the rescaled function

$$f(\boldsymbol{x}) = \frac{1}{2n}\sum_{i=1}^{n}(\boldsymbol{a}_i^*\boldsymbol{x} - b_i)^2, \qquad \nabla f(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{a}_i(\boldsymbol{a}_i^*\boldsymbol{x} - b_i)$$

where $\boldsymbol{a}_i$ are the rows of $\boldsymbol{A}$. To mitigate the linear dependence on n in the previous example we might note that randomly sampling i from $\{1, 2, \ldots, n\}$ and giving an approximate gradient descent update

$$\boldsymbol{x}_t = \boldsymbol{x}_{t-1} - \alpha_{t-1}\underbrace{\boldsymbol{a}_i(\boldsymbol{a}_i^*\boldsymbol{x}_{t-1} - b_i)}_{g_k}$$

gives the original gradient descent update in expectation while ignoring completely the number of samples n. Thus it is natural to think that the algorithm given by replacing the original gradient descent update with this *stochastic gradient* update should perform well, and you will prove that this is true. To do so, we will assume that the ℓ^2-norm $\sqrt{\mathbb{E}\|g_k\|_2^2} \leq G$ and f is ℓ-strongly convex (i.e. $\sigma_{\min}(\boldsymbol{A}) \geq \ell$, or $f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \nabla f(\boldsymbol{y})^*(\boldsymbol{x} - \boldsymbol{y}) + \frac{\ell}{2}\|\boldsymbol{x} - \boldsymbol{y}\|_2^2$).

(a) Prove via strong convexity that $\langle \nabla f(\boldsymbol{x}_t), \boldsymbol{x}_t - \boldsymbol{x}\rangle \geq \ell\|\boldsymbol{x}_t - \boldsymbol{x}\|_2^2$.

(b) Prove $\mathbb{E}\|\boldsymbol{x}_{t+1} - \boldsymbol{x}\|_2^2 \leq (1 - 2\ell\alpha_t)\mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{x}\|_2^2 + \alpha_t^2 G^2$.

(c) Prove that $\mathbb{E}\|\boldsymbol{x}_t - \boldsymbol{x}\|_2^2 \leq \frac{\max\{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|_2^2, G^2/\ell^2\}}{t}$ when we set $\alpha_t = \frac{1}{\ell(t+1)}$.

(d) Give an upper bound on the number of floating point operations it takes to compute an approximate least-squares solution $\tilde{\boldsymbol{x}}$ with $\mathbb{E}\|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_2 \leq \epsilon$ via stochastic gradient descent like in Problem 1. Can sparsity of $\boldsymbol{A}$ help?

(e) In which situations might you prefer to use this method over, say, gradient descent or a traditional exact solver?

(a) Play with the definition of strong convexity for the points $\boldsymbol{x}$ and $\boldsymbol{x}_t$.

(b) Expand $\|\boldsymbol{x}_{t+1} - \boldsymbol{x}\|_2^2$ deterministically by conditioning on the previous $\boldsymbol{x}_t$ before applying the expectation to reduce further.

(c) Induction.

(d) Reduce to part (c) using Jensen's inequality. You should get something like

$$\text{flops} \sim 4p\frac{\max\{\|\boldsymbol{x}\|_2^2, G^2/\ell^2\}}{\epsilon^2}.$$

■