CS60021: Scalable Data Mining
Practice Questions: Large Scale Optimization

State whether following statements are true or false with max 2-sentences of explanations:

a. Even non-differentiable convex loss functions can be minimized using SGD.
b. Mini-batches in SGD are expected to not only reduce the number of epochs till convergence, but also fluctuations in loss function values after each update.
c. SGD is best suited for large scale optimization (large number of examples) where very high accuracy is needed.
d. Dual decomposition converges and results in the same algorithm as ADMM if the loss function is strongly convex.
e. Adagrad is a variation of Nesterov's acceleration with direction specific normalization.
f. For minimization of convex and smooth loss functions, the best convergence rate achieved is $1/\sqrt{T}$, where T is the number of iterations.
g. Dual ascent may not converge for all convex optimization problems.
h. Given an empirical loss minimization problem with N datapoints, and M minibatches, SAG algorithm requires O(N) memory for the updates.
i. SGD with Nesterov momentum achieves linear convergence rates.
j. ADAM optimizer combines adaptive gradients with Nesterov momentum updates.
k. RMSprop updates preserve sparsity of gradients, but momentum updates may not.
l. Distributed SGD is slow for practical problems due to communication and synchronization bottleneck.

Q1. What is a decomposable loss function ? Write $L(x)$ as a decomposable loss. Write the stochastic gradient descent algorithm for this loss function. Show the final output. What should be step-size rule.

Q2. For the above SGD show that the final estimate of x converges to x^* in expectation.

Q3. Write the gradient for least squares loss function. Run 10 updates of SGD for the following dataset, and report the final training mean square error: (x_1, x_2, y): $(0,0,0), (0,1,0.6), (1,1,1), (1,0,0.6), (1,1,1.2)$

Q4. Given a loss function $L(x) = l_1(x) + l_2(x) + l_3(x)$, derive the ADMM updates for relevant variables. State the conditions under which the ADMM is guaranteed to converge. Write the expressions for primal and dual residuals.

Q5. Consider the situation where n noisy sensors trying to determine each others' temperature $t_1, \dots, t_n$ by minimizing discrepancy $D = \sum_{i=1}^{n} d_i$, with $d_i = \left|T_i - \sum_{j=1}^{n} s_{ij} t_j\right|^2$, where s_{ij} is some nearness measure between sensors i and j, and are known to both (i, j), but not other sensors. Devise an ADMM based algorithm that computes the optimal temperatures t_i^* given the temperature readings T_i (fixed) at each sensor.

Q6. Describe the stochastic gradient descent algorithm for optimizing an additive loss function. Formally show that:
For a convex loss function, the expected loss of weighted average of parameter iterates (weighted by the step length) generated by SGD, converges to the minimum loss.

Q7. Write the updates for RMSprop algorithm. Show 4 updates of RMSprop for the least square regression problem $y = w_1x_1 + w_2x_2$, trained on the following dataset: (x_1, x_2, y) = { (1,1,5) (-1,1,2) (1,-1,2) }. Use step length $\eta_t = \frac{1}{t}$, starting point as (0,0), and batch size as 1.

Q8. Derive the ADMM formulation for optimizing from first principles:

$$\min_{x_1,x_2} (x_1 - 5)^2 + (x_2 + 2)^2$$
$$sub.to: x_1 = x_2$$

such that updates to x_1 and x_2 happen in parallel. Clearly state the dual and consensus variables. Write expressions for primal and dual residuals.

Q9. Consider the following optimization problem:

$$\min_{x,y} (1 - x)^2 + 100(y - x)^2$$

Derive updates for the above problem using (i) SGD (ii) SGD with Nesterov momentum.

Q10. For the updates derived above, show 5 epochs of updates starting with the point (5,5) and learning rate 0.1. Assume weightage of momentum term as 0.2.

Q11. Derive an $O(\frac{1}{\sqrt{T}})$ bound on expected sub optimality $E[f(w^T)] - f(w^*)$ after T updates of form $w^{t+1} = w^t - \eta v^t$, where $v^t = \nabla f(w^t)$. You can assume that $|w| < B$ and $|v^t| < \rho$.

Q12. Derive the fully distributed ADMM updates for optimizing:

$$\min_{x_1,x_2} (x_1 - a_1)^2 + (x_2 - a_2)^2$$
$$sub.to: x_1 = x_2$$

such that updates to x_1 and x_2 happen in parallel on computers C_1 and C_2, without any central computer.

Q13. Derive the dual decomposition update for the above problem. Will it converge ?

Q14.