Optimization for Machine Learning (Homework #4)

Assignment date: Nov 2
Due date: Nov 16 (noon)

Theoretical Problems (10 points)

1. (4 points) Write down the ADMM algorithms for the following problems

   - (2 points) Let $\hat{\Sigma}$ be a $d \times d$ symmetric positive semidefinite matrix. We want to solve the following problem with $d \times d$ symmetric positive definite matrices X and Z:

   $$\min_{X,Z \succ 0} \left[-\ln \det(X) + \operatorname{trace}(\hat{\Sigma} X) + \lambda \|Z\|_1 \right], \qquad X - Z = 0.$$

   Write down the ADMM algorithm, and derive the closed form solutions for the sub-optimization problems. (**Input**: ρ, $\hat{\Sigma}$, X_0, T; **Output**: X_T)

   - (2 points) Let $x \in \mathbb{R}^d$, $b \in \mathbb{R}^d$, and Σ is a $d \times d$ symmetric positive definite matrix, A is $m \times d$ matrix. Assume that we want to solve the problem

   $$\min_x \left[\frac{1}{2} x^\top \Sigma x - b^\top x \right] \qquad \text{subject to } \|Ax\|_\infty \leq 1$$

   using ADMM by rewriting it as follows:

   $$\min_{x,z} \left[\frac{1}{2} x^\top \Sigma x - b^\top x + \mathbb{1}(\|z\|_\infty \leq 1) \right], \quad Ax - z = 0.$$

   Write down the ADMM algorithm for this decomposition with closed form solutions for the sub problems. (**Input**: ρ, Σ, A, b, x_0, T, m; **Output**: x_T)

2. (4 points) Write down the SGD algorithms for the following problems. Consider the k-class linear structured SVM problem, which has the following loss function:

   $$\frac{1}{n} \sum_{i=1}^{n} \max_{y'} u_{i,y'} + \frac{\lambda}{2} \|w\|_2^2,$$

   with the constraints

   $$\forall i \in \{1, \ldots, n\}, y' \in \{1, \ldots, k\} : u_{i,y'} = \left[\delta(y', y_i) - w^\top \psi(x_i, y_i) + w^\top \psi(x_i, y') \right].$$

   - (2 points) Write down the SGD procedure with batchsize 1 (single step update rule).
   - (2 points) Assume that $\sup_y \|\psi(x_i, y)\|_2 \leq A$ and you have a budget of T total gradient evaluations. How do you set learning rate in your SGD procedure, and what is the convergence rate you expect based on the lecture?

3. (6 points) Consider the L_1-L_2 regularized loss minimization problem:

$$\min_w \left[\underbrace{\frac{1}{n}\sum_{i=1}^{n} \ln(1+\exp(-w^\top x_i y_i)) + \frac{\lambda}{2}\|w\|_2^2}_{f(w)} + \underbrace{\mu\|w\|_1}_{g(w)} \right],$$

where $x_i, w \in \mathbb{R}^d$, $\|x_i\|_2 \le 1$, $y_i \in \{\pm 1\}$,$\lambda < 1$, and $(u)_+ = \max(0, u)$.

- (2 point) Estimate a simple upper bound of smoothness parameter and a simple lower bound of strong convexity parameter. Estimate a simple bound for the SGD variance V.
- (2 points) Write down the minibatch Accelerated Proximal SGD update rule with batch size m. (**Input**: w_0, λ, $\{\eta_t, \theta_t\}$, T, m; **Output**: w_T; Assume $\lambda > 0$) For $\tilde{T} = mT$ total gradient computations, what is the largest batch size you can choose? How do you want to set constant learning rate and momentum parameters for this batch size, and what's the convergence rate? You may use $O(\cdot)$ notation to hide constants.
- (2 points) Assume that $\lambda = 0$, write down minibatch stochastic RDA update rule for this problem with minibatch size m for $\tilde{T} = mT$ total gradients, by setting constant θ and η_0, and calculate the corresponding setting for η_t. (**Input**: w_0, $\eta_0 > 0$, $\theta \in (0, 1)$, T, m; **Output**: w_T)

  What is the largest batch size m, and what's the corresponding θ and η_0 and what is the convergence rate in terms of $\tilde{T}$? You may use $\tilde{O}(\cdot)$ which is up to a $\ln \tilde{T}$ factor.

  By comparing the solution of the proximal operator, can you explain why dual averaging can achieve more sparsity when the weights are near zero?

Programming Problem (6 points)

- Using the mnist class 1 (positive) versus 7 (negative) data.
- Use the python template "progtemplate.py", and implement functions marked with '# implement'.
  - (4 pts) Implement RDA-ACCL and ADMM-ACCL-linear and compare the RDA-ACCL algorithm with different θ schedulings. Submit your code and outputs.
  - (2 pts) Compare your plots to the theoretical convergence rates in class, and discuss your experimental results.