Assignment 1

Ankit Kumar
170121

Chaitanya Pathare
19111061

Hemant Kumar
170297

Rahul Ninaji Pakhare
19111068

Raunak Kumar
19111069

Abstract

Given The Squared SVM Solver we need to add dual constraint to it and define the Lagrangian function to it.And then eliminate the primal variables using first order optimality.Solve the problem with any 3 methods and plot it in graph

1 Question

Given equation

=> $argmin_{w \epsilon R^d} 1/2\mathbf{w}^2 + C\Sigma_{i=1}^{n}([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)^2 \quad (P1)$

rewritten as new optimization problem

=> $argmin_{w \epsilon R^d} 1/2\mathbf{w}^2 + C\Sigma_{i=1}^{n}\xi^2$ for all i $\in [1, n]$

$$s.t. \quad y^i < \mathbf{w}, \mathbf{x}^i > \geq 1 - \xi_i, \xi_i \geq 0 \quad for \quad all \quad i \in [1, n] \quad (P2)$$

Introducing dual variable α_i for each of the n constraints in (P2)

$$L(w, \xi, \alpha) = 1/2\mathbf{w}^2 + C\Sigma_{i=1}^{n}\xi_i^2 + \Sigma_{i=1}^{n}\alpha_i(1 - \xi_i - y^i < \mathbf{w}, \mathbf{x}^i >) \quad , \alpha_i \geq 0 \qquad (P3)$$

Preprint. Under review.

2 Question

To get the dual problem from the above equation (P3). We have appended one extra dimension to our data i.e if X = $\begin{bmatrix} x_{11} & x_{12} & x_{13} & . & x_{1n} \\ x_{21} & \vdots & \vdots & \vdots & x_{2n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{n1} & \vdots & \vdots & \vdots & x_{nn} \end{bmatrix}$ and then transformed X $\rightarrow$ X^0 = $\begin{bmatrix} x_{11} & x_{12} & x_{13} & . & x_{1n} & 1 \\ x_{21} & . & . & . & x_{2n} & 1 \\ . & . & \ddots & \ddots & . & 1 \\ . & . & \ddots & \ddots & . & 1 \\ . & . & \ddots & \ddots & . & 1 \\ x_{n1} & . & . & . & x_{nn} & 1 \end{bmatrix}$ so that w[d-1] =b , { d is dimension of each data point in X^0 }

put ${}^{\delta L}/_{\delta}\mathbf{w} = \mathbf{0} and {}^{\delta L}/_{\delta \xi_i} = \mathbf{0} \quad for \quad all \quad i \in [1, n]$

$$ {}^{\delta L}/_{\delta}\mathbf{w} = \mathbf{w} - \Sigma_{i=1}^{n} \alpha_i \mathbf{x}^i = \mathbf{0} $$

$$ or \quad \mathbf{w} = \Sigma_{i=1}^{n} \alpha_i \mathbf{x}^i $$

and

$$ {}^{\delta L}/_{\delta \xi_i} = 2C\xi_i - \Sigma_{i=1}^{n} \alpha_i = \mathbf{0} $$
$$ or \quad \xi_i = {}^{1}/_{2C} \Sigma_{i=1}^{n} \alpha_i $$

Optimization problem comes down to

$$ => {}^{1}/_{2} \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} (\alpha_i \alpha_j y^i y^j < \mathbf{x}^i \mathbf{x}^j >) + {}^{1}/_{4C} \Sigma_{i=1}^{n} \alpha_i^2 + \Sigma_{i=1}^{n} \alpha_i - {}^{1}/_{2C} \Sigma_{i=1}^{n} \alpha_i^2 $$

$$ - \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} (\alpha_i \alpha_j y^i y^j < \mathbf{x}^i \mathbf{x}^j >) $$

$$ => argmin(\Sigma_{i=1}^{n} \alpha_i - {}^{1}/_{4C} (\Sigma_{i=1}^{n} \alpha_i^2) - {}^{1}/_{2} \Sigma_{i=1}^{n} \Sigma_{j=1}^{n} (\alpha_i \alpha_j y^i y^j < \mathbf{x}^i \mathbf{x}^j >)) \quad (D2) $$

$$ All \quad above \quad \mathbf{w} \epsilon R^{d+1}, \quad \mathbf{x} \epsilon R^{d+1} $$

3 Question

a. Mini Batch Stochastic Gradient descent on P1

Given,

$$ P1 \quad f = \quad argmin_{w \epsilon R^d} {}^{1}/_{2} \mathbf{w}^2 + C \Sigma_{i=1}^{n} ([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)^2 $$

now get gradient for P1

$$ \nabla f = \mathbf{w} + 2C \Sigma_{i=1}^{n} ([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)(-y^i \mathbf{x}^i) $$

Note: $[1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+ = \{(1 - y^i < \mathbf{w}, \mathbf{x}^i >) \quad if \quad y^i < \mathbf{w}, \mathbf{x}^i >< 1$

$$ 0 \quad if \quad y^i < \mathbf{w}, \mathbf{x}^i > \geq 1 \} $$

iterate till the time end for the X

Do

```
1.find gradient for each data point in the given Batch set B and step length eta

2.update w for these
```

i.e, $\mathbf{w} = \mathbf{w} - \eta 2C\Sigma_{i=x}^{x+B}([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)(-y^i\mathbf{x}^i)$
Differengt cobination of (batch B) and step length η we tried are as follows,all works fine

B	η
10	0.01
100	0.05
1000	0.001

b. Coordinate descent on P1
Given,

$$P1 \quad f = \quad argmin_{w \epsilon R^d} 1/2\mathbf{w}^2 + C\Sigma_{i=1}^{n}([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)^2$$

now get gradient for P1

$$\nabla f = \mathbf{w} + 2C\Sigma_{i=1}^{n}([1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+)(-y^i\mathbf{x}^i)$$

Note: $[1 - y^i < \mathbf{w}, \mathbf{x}^i >]_+ = \{(1 - y^i < \mathbf{w}, \mathbf{x}^i >) \quad if \quad y^i < \mathbf{w}, \mathbf{x}^i >< 1$

$$0 \quad if \quad y^i < \mathbf{w}, \mathbf{x}^i >\geq 1\}$$

iterate till the time end for the X

Do

```
1.find gradient for each data point choosing the coordinate cyclically i,e j th= 1,2...d,1,2..

2.update w for these
```

i.e, $\mathbf{w} = \mathbf{w} - \eta 2C\Sigma_{i=1}^{n}([1 - y^i < \mathbf{w}, \mathbf{x}^i_{j^th} >]_+)(-y^i\mathbf{x}^i_{j^th})$
Note :η value was choosen very small or else the curve diverges.

c. Method used for D2 maximization

$$D2 \qquad => argmax(\Sigma_{i=1}^{n}\alpha_i - 1/4C(\Sigma_{i=1}^{n}\alpha_i^2) - 1/2\Sigma_{i=1}^{n}\Sigma_{j=1}^{n}(\alpha_i\alpha_j y^i y^j < \mathbf{x}^i\mathbf{x}^j >))$$

concentrate on α_i

$$argmax(\alpha_i - 1/4C(\alpha_i^2) - (\Sigma_{i\neq j}\alpha_j^2) - 1/2(\alpha_i^2(\mathbf{x}^i)^2) - \alpha_i y^i\Sigma_{i\neq j}(\alpha_j y^j < \mathbf{x}^i\mathbf{x}^j >))$$

$$Let \quad x = \alpha_i, \quad q = (\mathbf{x}^i)^2, \quad p = y^i\Sigma_{i\neq j}(\alpha_j y^j < \mathbf{x}^i\mathbf{x}^j >), \quad r = (\Sigma_{i\neq j}\alpha_j^2)$$

$$argmin(x^2(1/2q + 1/4C) - x(1 - p) + r)$$

$$Let \quad {}^{q^0}/_2 = ({}^{q}/_2 + \frac{1}{4C}) \qquad argmin({}^{q^0}/_2 x^2 - x(1-p) + r)$$

minimum at $x' = {}^{(1-p)}/_{q^0}$

If $x' \in [0, \infty]$ then x' is solution

elseif $x' < 0$ solution is 0

else solution is ∞.

Note:

$$p = y^i \Sigma_{i \neq j} (\alpha_j y^j < x\mathbf{x}^i \mathbf{x}^j >)$$

$$= \mathbf{w}^T \mathbf{x}^i - \alpha_i y^i q^0$$

Algorithm minimize:

$$intialize \quad \alpha^T = \{0, 0, ...0\}_{1Xn} \quad \mathbf{w} = \{0, 0, ...0\}_{1Xn}$$

$$\mathbf{w} = \Sigma_{i=1}^{n} \alpha_i y_i \mathbf{x}^i \quad for \quad i \in [1, n]$$

Note: We have appended one extra dimension to our data i.e if X = $\begin{bmatrix} x_{11} & x_{12} & x_{13} & . & x_{1n} \\ x_{21} & \vdots & \vdots & \vdots & x_{2n} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{n1} & \vdots & \vdots & \vdots & x_{nn} \end{bmatrix}$ and then transformed X $\rightarrow$ X^0 = $\begin{bmatrix} x_{11} & x_{12} & x_{13} & . & x_{1n} & 1 \\ x_{21} & . & . & . & x_{2n} & 1 \\ . & . & \ddots & \ddots & . & 1 \\ . & . & \ddots & \ddots & . & 1 \\ . & . & \ddots & \ddots & . & 1 \\ x_{n1} & . & . & . & x_{nn} & 1 \end{bmatrix}$ so that w[d-1] =b , { d is dimension of each data point in X^0 }.Iterate till time does not end, for each data point in X^0

Do-

1. Calculate α_i if $\alpha_i \geq 0$ then $\alpha[i] = \alpha_i$

else $\alpha[i] = 0$

2. Update w, $\mathbf{w} = \Sigma_{i=1}^{n} \alpha_i y_i \mathbf{x}^i$

w will converge after sufficient number of iteration.

4 Question

Hyperparameters used

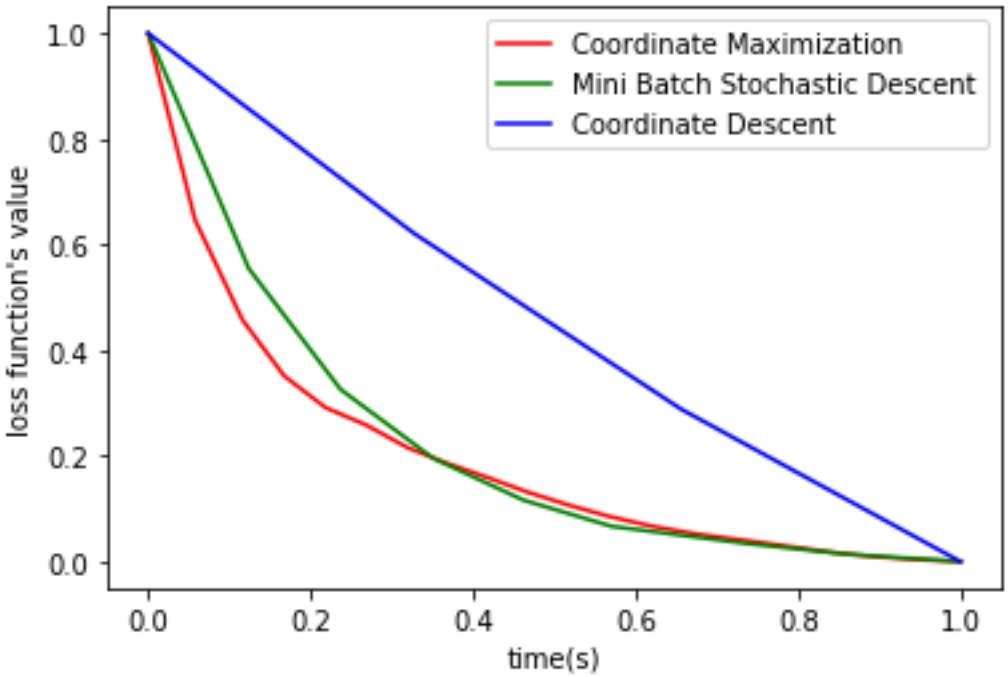
a.Mini Batch SGD on P1: tried different values for Batch size from 10 to 1000 all gives similar results with $\eta = 0.001$ or 0.01.

b.Coordinate descent on P1: here the step length need to be very small or else the curve diverges. We tried with step length 0.1 , 0.01 ,..and it kept on diverging.Thus fixed step length to a very small number 0.000001 that lead to convergence of the curve.

c.Coordinate Maximization on D2: No hyperparameter used.

5 Question

Figure: Plot for convergence curves offered by those 3 method



Following graph was normalized using the minimum and maximum value for both the axis,as given below

a.Mini Batch SGD on P1: X axis (time): Min = 0.3228445053100586 Max = 3.0747430324554443

Y axis (loss function value): Min = 5607.35 Max = 6252.87

b.Coordinate Descent on P1: X axis (time): Min = 7.35390830039978 Max = 29.89153790473938

Y axis (loss function value): Min = 13471.7 Max = 18036.3

c.Coordinate Maximization on D2: X axis (time): Min = 0.4362635612487793 Max = 9.31814169883728

Y axis (loss function value): Min = 5236.12 Max = 9059.85

6 Question

Among the above three curves it is clear that Mini Batch Stochastic gradient descent gives the best convergence for the graph as seen in the plot.