HW1 Intelligent Systems

Linear classification & optimization

Fardin Abbasi 810199456

School of Electrical and Computer Engineering

College of Engineering

University of Tehran

Fall 2023

Questions

Q1: Convex optimization (Theory)

A: Finding optimal point

If *f* is a convex function, it has only one optimal point which is globally optimal.

In order to experiment whether f is convex or non-convex, we construct its **Hessian matrix**.

$$f(x_1, x_2) = 100(x_2 - x_1)^2 + (1 - x_1)^2$$

$$\nabla f(x_1, x_2) = \begin{bmatrix} -200(x_2 - x_1) - 2(1 - x_1) \\ 200(x_2 - x_1) \end{bmatrix}$$

$$\nabla^2 f(x_1, x_2) = \begin{bmatrix} 202 & -200 \\ -200 & 200 \end{bmatrix}$$

Since Hessian matrix is positive definite, we conclude f is a convex function and its global optimal point is where $\nabla f(x_1, x_2) = 0$.

$$f(x_1, x_2) = \begin{bmatrix} -200(x_2 - x_1) - 2(1 - x_1) \\ 200(x_2 - x_1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \to \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

B: Finding optimal point with steepest descent method

1.

$$\nabla f(x_1, x_2) = \begin{bmatrix} -200(x_2 - x_1) - 2(1 - x_1) \\ 200(x_2 - x_1) \end{bmatrix}$$

2.

```
Initial guess x_0 at k = 0
while ||∇f(x_k)|| > accuracy do
    Find the search direction s_k = −∇f(x_k)
    Solve for α_k by decreasing f(x_k + α s_k) significantly
        satisfying the Wolfe conditions
    Update the result x_{k+1} = x_k + α_k s_k
    k ← k + 1
end while
```

$$X_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \alpha = 0.5$$

$\boldsymbol{k = 0}$:

$\nabla f(X_0) = \begin{bmatrix} -2 \\ 0 \end{bmatrix}$

$X_1 = X_0 - \alpha \nabla f(X_0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.5 \times \begin{bmatrix} -2 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$\boldsymbol{k = 1}$:

$$\nabla f(X_1) = \begin{bmatrix} 200 \\ -200 \end{bmatrix}$$

$$X_2 = X_1 - \alpha \nabla f(X_1) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} - 0.5 \times \begin{bmatrix} 200 \\ -200 \end{bmatrix} = \begin{bmatrix} -99 \\ 100 \end{bmatrix}$$

C: Finding optimal point with Newton method

1.

$$Hessian\ Matrix: \nabla^2 f(x_1, x_2) = \begin{bmatrix} 202 & -200 \\ -200 & 200 \end{bmatrix}$$

2.

Algorithm 3: Newton's Method

input : $f : \mathbb{R}^n \to \mathbb{R}$ a twice-differentiable function
$\mathbf{x}^{(0)}$ an initial solution

output: $\mathbf{x}^\star$, a local minimum of the cost function f.

begin
  $k \leftarrow 0$;
  while STOP-CRIT **and** $(k < k_{max})$ **do**
    $\mathbf{x}^{(k+1)} \leftarrow \mathbf{x}^{(k)} + \boldsymbol{\delta}^{(k)}$;
    with $\boldsymbol{\delta}^{(k)} = -\left(\mathbf{H}_f\left(\mathbf{x}^{(k)}\right)\right)^{-1} \boldsymbol{\nabla} f\left(\mathbf{x}^{(k)}\right)$;
    $k \leftarrow k + 1$;
  return $\mathbf{x}^{(k)}$
end

$$X_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$\boldsymbol{k = 0}$:

$$\nabla f(X_0) = \begin{bmatrix} -2 \\ 0 \end{bmatrix}, \nabla^2 f(X_0) = \begin{bmatrix} 202 & -200 \\ -200 & 200 \end{bmatrix} \to \nabla^2 f(X_0)^{-1} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 101/200 \end{bmatrix}$$

$$X_1 = X_0 - \nabla^2 f(X_0)^{-1} \nabla f(X_0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Since f is a convex function, it converges to optimal point with 1 iteration.

Q2: Non-convex optimization

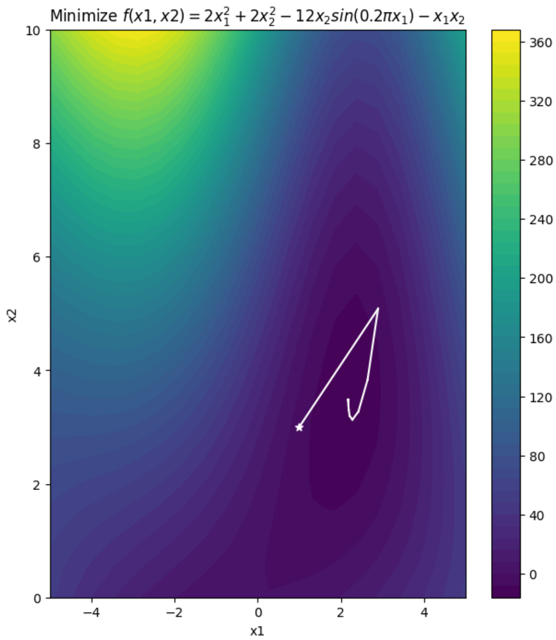
A: Newton implementation

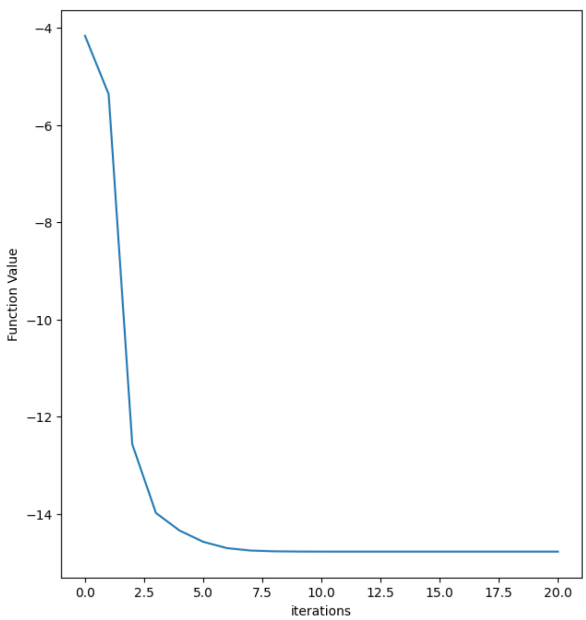
$$f(X) = 2x_1^2 + 2x_2^2 - 12x_2\sin(0.2\pi x_1) - x_1x_2$$

$$\nabla f(X) = \begin{bmatrix} 4x_1 - 2.4\pi x_2\cos(0.2\pi x_1) - x_2 \\ 4x_2 - 12\sin(0.2\pi x_1) - x_1 \end{bmatrix}$$

$$\nabla^2 f(X) = \begin{bmatrix} 4 + 0.48\pi^2 x_2 \sin(0.2\pi x_1) & -2.4\pi\cos(0.2\pi x_1) - 1 \\ -2.4\pi\sin(0.2\pi x_1) - 1 & 4 \end{bmatrix}$$

When starting from the initial point $X_0 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$, the Newton method converges to the optimal point $X_* = \begin{bmatrix} 2.18 \\ 3.48 \end{bmatrix}$ with a corresponding value of $f(X_*) = -14.78$. Below, you can observe the convergence plot and the cost function plot over the course of iterations.





B: Meta-heuristic (Simulated annealing)

Pseudocode of simulated annealing is written as follows.

Simulated annealing algorithm

```
Select the best solution vector x_0 to be optimized
Initialize the parameters: temperature T, Boltzmann's constant k, reduction factor c
while termination criterion is not satisfied do
    for number of new solution
        Select a new solution: x_0+Δx
        if f(x_0+Δx) > f(x_0) then
            f_new = f(x_0 + Δx); x_0 = x_0 + Δx
        else
            Δf = f(x_0 + Δx) − f(x_0)
            random r(0, 1)
            if r > exp(−Δf/kT) then
                f_new = f(x_0 + Δx), x_0 = x_0 + Δx
            else
                f_new = f(x_0),
            end if
        end if
        f = f_new
        Decrease the temperature periodically: T = c x T
    end for
end while
```

In this problem neighbors are defined as below:

$$x_{neighbor} = x_{current} + N(0,0.5)$$

With choosing the following initial parameters, the best solution is $X_* = \begin{bmatrix} -2.21 \\ -3.56 \end{bmatrix}$ with $f(X_*) = -14.76$.

$$Initial\ solution: X_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$Initial\ temprature: T_0 = 100$$

$$Cooling\ rate: T_{k+1} = \frac{T_k}{1+k}$$

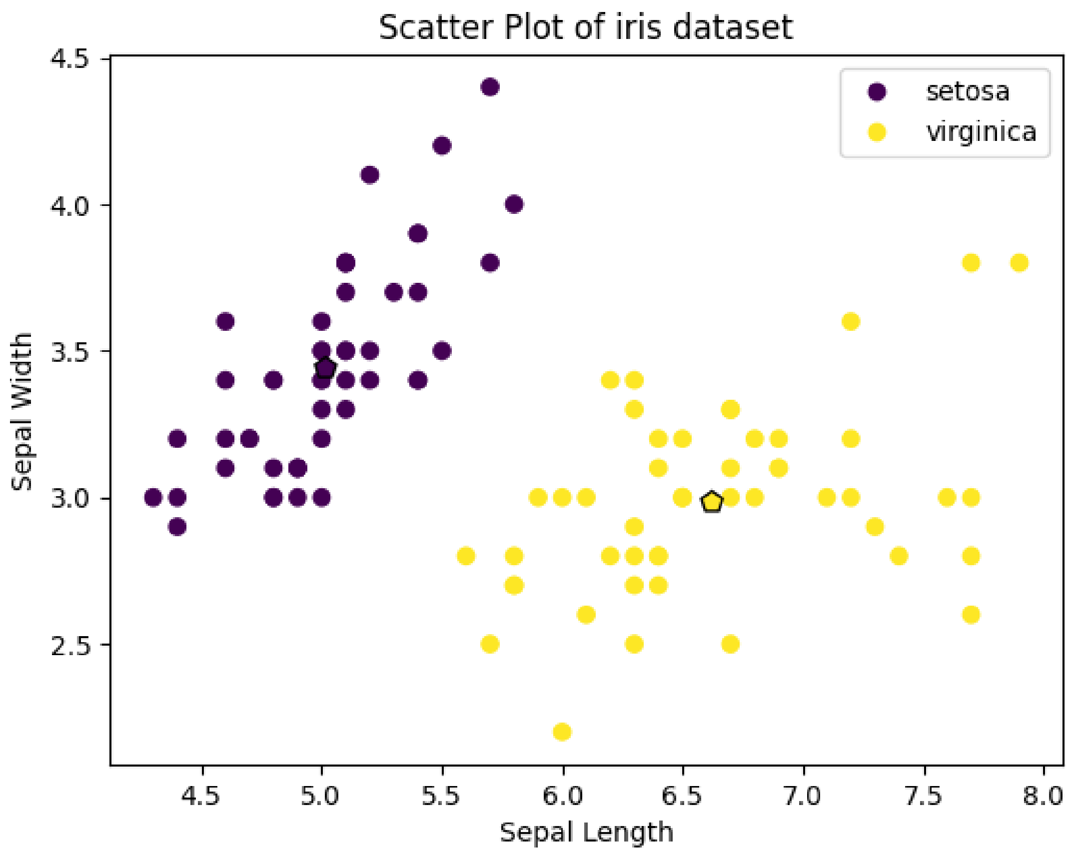
$$Search\ space: \begin{bmatrix} -15 < x_1 < 15 \\ -15 < x_2 < 15 \end{bmatrix}$$

$$Num\ iterations = 100$$

Q3: SVM

A: Data preparation

The scatter matrix of the Iris dataset is depicted below, with each class's mean represented using a '⌂' sign.



B: Classifier implementation

In soft SVM, the cost function is as follows. The weights (w) and bias (b) are updated using mini-batch gradient descent, following these update rules:

$$Cost\ function: L(w,b,c) = 0.5||w||^2 + C\sum Max(0,1 - y_i(w^T x_i + b))$$

$$w \leftarrow w - \eta\nabla L_w = w - \eta w + \eta C y_i x_i$$

$$b \leftarrow b - \eta\nabla L_b = b + \eta C y_i$$

C: Model training

The final weights are recorded as follows:

$$w = \begin{bmatrix} -1.539 \\ 2.48 \end{bmatrix}, b = 0.83$$

D: Performance measurement

The loss and precision for the test set are recorded as follows:

$$loss = 4.34$$

```
              precision    recall  f1-score   support

          -1       1.00      1.00      1.00         8
           1       1.00      1.00      1.00        11

    accuracy                           1.00        19
   macro avg       1.00      1.00      1.00        19
weighted avg       1.00      1.00      1.00        19
```

Also the classification result is depicted below: