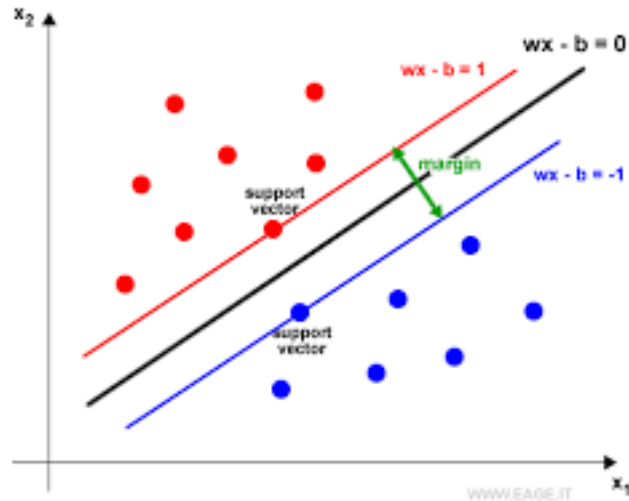


SVM_from_scratch



In this project, we have implemented a support vector classifier to solve a binary class classification

task. The task is distinguishing between images of handwritten digits from the MNIST database.

Each sample is a 28x28 gray scale image, associated with a label. The project contains 3 different implementations grouped as Q1, Q2 and Q2 as show in the repository structure

Repository structure

```

SVM_from_scratch
|  README.md
|  file001.txt
|  data.py
|  train-images-idx3-ubyte
|  train-labels-idx1-ubyte
├── Q1
|   |  SVM.py
|   |  run_1.py
|
├── Q2
|   |  SVM_decompose.py
|   |  run_2.py
|
```

```

└─Q3
  |   MVP.py
  |   run_3.py
  ...

```

Q1

In this implementation , we will only consider numbers the images and labels of numbers 3 and 8. To solve this task, we used "SLSQP" constraint optimizer from scipy library which solves the dual soft margin svm quadratic optimization problem with a linear/rbf/polynomial kernel. We have splitted data into train/test with fixed seed and 80%/20% proportions.

Q2

Here we have implemented the SVM light decomposition method. SVM light is an iterative method where in each step a sub set of data points are selected (which is called the working set) and SLSQP algorithm solves the quadratic optimization problem for the subset. Here the crucial part is selection of the subset at each iteration and the stopping criteria of the decomposition algorithm. The SVM light algorithm suggests choosing $q/2$ indices from $R(\alpha^k)$ and $q/2$ indices from $S(\alpha^k)$ where q cardinality of the working set and $R(\alpha^k)$ and $S(\alpha^k)$ are shown in formulas bellow.

$$S(\alpha^k) = \{i : (\alpha_i \leq C \ \& \ y_i = -1), (\alpha_i \geq 0 \ \& \ y_i = 1)\}$$

$$R(\alpha^k) = \{i : (\alpha_i \leq C \ \& \ y_i = 1), (\alpha_i \geq 0 \ \& \ y_i = -1)\}$$

. Based on that, the selection rule is defined as,

- select first $q_1 = q/2$ indices from $R(\alpha^k)$ such that $R(\alpha^k)$ is sorted as bellow;

$$\frac{-\nabla f(\alpha^k)_{i^1}(k)}{y_{i^1}(k)} \geq \frac{-\nabla f(\alpha^k)_{i^2}(k)}{y_{i^2}(k)} \geq \dots \geq \frac{-\nabla f(\alpha^k)_{i^{q_1}}(k)}{y_{i^{q_1}}(k)}$$
- select first $q_2 = q/2$ indices from $S(\alpha^k)$ such that $S(\alpha^k)$ is sorted as bellow;

$$\frac{-\nabla f(\alpha^k)_{j^1}(k)}{y_{j^1}(k)} \leq \frac{-\nabla f(\alpha^k)_{j^2}(k)}{y_{j^2}(k)} \leq \dots \leq \frac{-\nabla f(\alpha^k)_{j^{q_2}}(k)}{y_{j^{q_2}}(k)}$$

The ideal stopping criteria is satisfaction of the KKT conditions shown below.

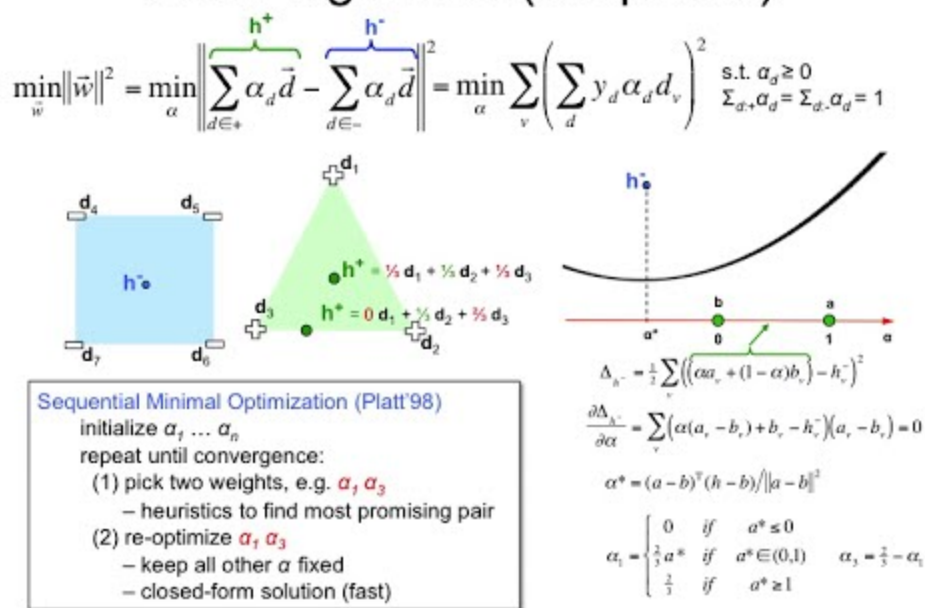
$$m(\alpha^k) = \max_{i \in R(\alpha^k)} \{-\nabla f(\alpha^k)_i y_i\} \leq \min_{j \in S(\alpha^k)} \{-\nabla f(\alpha^k)_j y_j\} = M(\alpha^k)$$

But generally it takes a lot of iterations to satisfy this inequality exactly and this extremely increases the computation time. To avoid this, we let the violation of the KKT condition up to some small threshold which is our tolerance hyper-parameter equal to

0.0001

Q3

SMO algorithm (simplified)



Here we have implemented the Sequential Minimal Optimization (SMO) algorithm to solve the same soft margin svm quadratic problem with an iterative method. SMO is a decomposition method. The value of q is fixed to 2, all the time. Fixing $q = 2$ makes the sub problem a convex quadratic problem of two variables which can be solved analytically. In the selection of these 2 indices, we use the Most Violating (MVP) pairs. Two indices $i \in I(\alpha)$ and $j \in J(\alpha)$ are MVP pairs, if and only if equations below are satisfied and KKT condition not satisfied.

$$I(\alpha) = \{i : i \in \operatorname{argmax}_{i \in R(\alpha)} \{-\nabla f(\alpha)_i y_i\}\}$$

$$J(\alpha) = \{j : j \in \operatorname{argmin}_{j \in S(\alpha)} \{-\nabla f(\alpha)_j y_j\}\}$$