Problem 1 – Maximum Entropy Discrimination (MED)

In this problem, you will implement a MED classifier. Notation: to be consistent with the course notes, \( x_i \in \mathbb{R}^n \) will denote example \( i \), and \( x_{i,j} \) will denote the \( j \)th coordinate of example \( i \), and \( a_j \) will be the \( j \)-th coordinate of a vector \( a \in \mathbb{R}^n \). The number of samples is \( N \).

We consider the family of classifiers
\[
\mathcal{F} = \{ f_w(x) = w^T x \mid w_{1:3} \in [-W, W] \} \tag{1}
\]
The data domain is \( \mathbb{R}^3 \), with \( x = [x_1 \ x_2 \ 1] \). In other words, we have 2D data, but we add a constant coordinate to make \( f_w \) implement an affine classifier.

In MED, we search for a distribution \( q \) over \( \mathcal{F} \) that has maximum entropy subject to classifying the data correctly. Since \( \mathcal{F} \) is parametrized by \( w \in [-W, W]^3 \), our \( q \) will be a distribution over \( [-W, W]^3 \).

1. Write the MED primal for the domain we are considering. Denote the dual variables/Lagrange multipliers with \( \lambda_i \), \( i = 1 : N \). Write the expression of \( L(q, \lambda) \). Note that this expression will involve integrals.

2. Compute the partial derivative \( \frac{\partial L}{\partial q(w)} \). and from it show that
\[
q(w) \propto e^{\sum_i \lambda_i y_i w^T x_i} \tag{2}
\]

3. Denote \( A = \sum_i \lambda_i y_i x_i \). Show that the normalization constant for \( q(w) \) in (2) is
\[
Z(\lambda) = \prod_{j=1}^{3} \frac{e^{A^j W} - e^{-A^j W}}{A^j} \tag{3}
\]

4. Calculate the expression of \( \frac{\partial \ln Z(\lambda)}{\partial \lambda_i} \). Show that it is of the form \( y_i x_i^T B(\lambda) \) with \( B \in \mathbb{R}^3 \).

5. Calculate the expression of \( \frac{\partial^2 \ln Z(\lambda)}{\partial \lambda_i \partial \lambda_k} \). Show that it is of the form \( y_i y_k x_i^T C(\lambda)x_k \) with \( C \) a diagonal \( 3 \times 3 \) matrix.

6. Write the dual MED problem as a convex minimization problem.

7. Compute the expected \( \hat{w} = E_q(\lambda)[w] \), as a function of \( \lambda \), (it will be a function of quantities you already have).

1
8. Write the expression of the MED classifier (it will be a function of quantities you already have), and show that it is a linear classifier.

9. Show that the expression of the negative entropy is

\[ -H(q) = A^T \bar{w} - \ln Z(\lambda) \]  

(4)

(There is more than one way to obtain this expression)

10. Now we will start solving the MED dual problem by the barrier method. Denote the dual objective \( g(\lambda) \). Write the expression of the penalized dual \( g_t(\lambda) = g(\lambda) - \phi(\lambda)/t \) and of its gradient and Hessian.

11. Implementation: Implement the barrier method to solve this problem. Parameters: Initial \( \lambda_{1:n} = 1 \) (guaranteed feasible), initial \( t = 0.1, \mu = 10 \), tolerance \( \epsilon = 10^{-4} \). Describe exactly how you do the unconstrained minimization in the inner loop, what criterion you use for stopping the centering step, and what criterion you use to stop the optimization altogether. (Hint: the duality gap can be easily obtained from quantities you calculated in questions 1–9).

Notes on implementation:

1. To make the debugging simpler, you can implement gradient descent with fixed step size and fixed number of iterations in order to see that you are obtaining the right result. After you are sure that the code works, you can change to the Newton step and you can implement the appropriate stopping criteria.

2. The algorithm becomes unstable when a \( \lambda_i \) reaches 0. Sometimes because the steps are discrete \( \lambda_i \) can cross into negative values, which will ruin the following steps. A quick fix is to set \( \lambda = \max(\lambda, 0) \). Another problem is that for \( \lambda_i = 0 \) the gradient of \( g_t \) is unstable (or infinite). A possibility is to set to 0 the coefficient of \( 1/t \) in \( \partial g_t / \partial \lambda_i \) when \( \lambda_i = 0 \).

3. Related to the above: an easy method to debug such cases is to remove the data points for which \( \lambda_i \) becomes 0 from the data set. This way you can debug the rest of the algorithm.

12. Run your MED algorithm on the small.dat data set that consists of 4 points, taking \( W = 2 \). Plot the primal and dual objective on the same graph for all iterations, \( \lambda_{1:3} \) on the same graph for all iterations, \( \bar{w}_{1:3} \) on the same graph for all iterations.

The data file format is \( y_i \) \( x^1_i \) \( x^2_i \) \( 1 \)

Hints: in this problem a number of symmetries will be useful to notice. For instance, you can tell by inspection which will be the support vectors and which will be the \( \lambda_i \)’s to take 0 value. Or, of all the non-zero \( \lambda \)’s are equal, then there are some symmetries in \( A, B \) and so on.
13. Plot the data and decision boundary (it will be a line) on the same plot. Circle (by hand OK) the support vectors.

14. Are $w^j$ mutually independent under $q^j$? Prove or disprove. Plot the distribution $q(w)$ over the 2D domain $w_1 = w_2, w_3$. Mark $\bar{w}$ on this plot.

15. [Optional] What will change in the MED problem (qualitatively OK) if $W = 0.2$ instead of 2, and why? Same questions if $W = 4$ instead of 2.

16. Now run the MED algorithm on the data `linear-med.dat` which has $N = 100$. Plot the same variables as for the small data, questions 12, 13.

**Problem 2 – The Markov chain as maximum entropy model**

*Requires working familiarity with the Markov Chain, Markov Random Fields*

Let $X_1, X_2, \ldots, X_n \in \{0, 1\}$ be a collection of binary random variables. We want to define a joint distribution $P$ over $X_{1:n}$ which has prescribed marginals, and maximum entropy.

2.1 Denote $x = (x_{1:n}) \in \{0, 1\}^n$ a configuration of $X$, by $p_x = P(X_{1:n} = x)$ the value of the joint distribution at $x$, and by $p$ the vector $p = (p_x)_{x \in \{0, 1\}^n}$ The above problem can be formulated as

$$\max_p \quad H(p) \tag{5}$$

$s.t. \quad E[X_i X_{i+1}] = r_i, \quad i = 1 : n - 1 \tag{6}$

$$E[X_1] = r_0 \tag{7}$$

$$E[1] = 1 \tag{8}$$

Write the problem explicitly as a function of $p$ and show that this problem is/can be formulated as a convex program.

2.2 Denote by $\theta_i, \theta_0, \theta'$ the Lagrange multipliers associated respectively with (6), (7), (8). Write the Lagrangean function $L(p, \theta_i, \theta_0, \theta')$ for this optimization problem, then calculate the value $p(\theta)$ that achieves $\inf_p L(p, \theta)$, for $\theta = (\theta_i, \theta_0, \theta')$.

Feel free to leave any normalization constant unevaluated (e.g. replace it with $Z(\theta)$).

2.3 Examine the form of the maximum entropy solution $p(\theta)$ obtained in 2.2. Show that it represents a Markov chain.

[Problem 3 – $\nu$-SVN Not graded]

3.a Find the dual for the $\nu$-SVM Support Vector Machine (see also lecture notes)
defined by

\[
\begin{align*}
\text{minimize}_{w,b,\xi,\rho} & \quad \frac{1}{2} ||w||^2 - \nu \rho + \frac{1}{N} \sum_{i} \xi_i \\
\text{s.t.} & \quad y^i (w^T x^i + b) \geq \rho - \xi_i \\
& \quad \xi_i \geq 0 \\
& \quad \rho \geq 0
\end{align*}
\]

(9)

where \( \nu \in [0, 1] \) is a parameter. 3.b Prove the following statements (also from the lecture notes) If \( \rho > 0 \) then:

- \( \nu \) is an upper bound on \( \# \)margin errors/\( N \) (if \( \sum_i \alpha_i = \nu \))
- \( \nu \) is a lower bound on \( \# \)original support vectors + margin errors)/\( N \)
- \( \nu \)-SVM leads to the same \( w, b \) as C-SVM with \( C = 1/\nu \)