University of Illinois Fall 2023

ECE 544: Pattern Recognition
Problem Set 2

Due: Friday, September 29, 2023, 11:59 pm

1. **[Max-Margin Support Vector Machine]**

   We are given a dataset $\mathcal{D} = \left\{ \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, 1 \right), \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, 1 \right), \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, -1 \right), \left(\begin{bmatrix} -1 \\ -1 \end{bmatrix}, -1 \right) \right\}$ containing four pairs (x, y), where each $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$ denotes a 2-dimensional point and $y \in \{-1, +1\}$.

   We want to train the parameters w and the bias b of a max-margin support vector machine (SVM) using (with hyperparameter C)

   $$\min_{w,b} \frac{C}{2} \|w\|_2^2 \quad \text{s.t.} \quad \forall (x^{(i)}, y^{(i)}) \in \mathcal{D} \quad y^{(i)}(w^\top x^{(i)} + b) \geq 1. \tag{1}$$

   (a) For the given data $\mathcal{D}$, how many constraints are part of the program in Eq. (1)? Specify all of them **explicitly**.

   (b) Highlight the feasible set in w_1-w_2-space for $b = 0$, $b = -1$ and $b = -2$. For each of the three choices for b also highlight the optimal w. Given only the three options $b \in \{0, -1, -2\}$ what is the optimal solution? Does a better solution exist (reason)?

   (c) Draw the dataset in x_1-x_2-space using crosses for the points belonging to class 1 and circles for the points belonging to class -1. Find by inspection and highlight the support vectors, *i.e.*, those points for which the constraints hold with equality at the optimal solution. Solve the resulting linear system w.r.t. w and b and draw the solution into x_1-x_2-space.

   (d) What conditions do the datapoints have to fulfill such that the program in Eq. (1) has a feasible solution?

   (e) In practice, for large datasets, it is hard to find the support vectors by inspection. A gradient based method is applicable. Use **general** notation, introduce slack variables into the program given in Eq. (1) and state the corresponding program (including all constraints). Subsequently, reformulate this program into an unconstrained program. Finally compute the gradient of this unconstrained program w.r.t. w (use $\frac{\partial}{\partial x} \max\{0, x\} = 1$ for $x > 0$, 0 otherwise). Evaluate the gradient at $w_1 = 2$, $w_2 = 2$ and $b = -1$. What can we conclude?

   (f) Complete `A3_SVM.py` and verify your reply for the previous answer. What is the optimal solution (w, b) that your program found and what is the corresponding loss? Explain the solution and what you observe when running the program, as well as how to fix this issue.

2. **[L2 SVM]**

   We are given a dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{|\mathcal{D}|}$ with feature vectors $\mathbf{x}^{(i)} \in \mathbb{R}^d$ and their corresponding labels $y^{(i)} \in \{-1, 1\}$. The following is the primal formulation of L2 SVM, a variant of the standard SVM obtained by **squaring** the hinge loss,

   $$\begin{aligned} \min_{\mathbf{w}, b, \boldsymbol{\xi}} \quad & \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + \tfrac{C}{2} \textstyle\sum_{i=1}^{|\mathcal{D}|} (\xi^{(i)})^2 \\ s.t. \quad & y^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \xi^{(i)}, \ i \in \{1, \cdots, |\mathcal{D}|\} \\ & \xi^{(i)} \geq 0, i \in \{1, \cdots, |\mathcal{D}|\} \end{aligned} \tag{2}$$

Here, $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ are learnable weights and $\boldsymbol{\xi} = (\xi^{(1)}, \ldots, \xi^{(|\mathcal{D}|)}) \in \mathbb{R}^{|\mathcal{D}|}$ are slack variables. We will first show that removing the last set of constraints $\boldsymbol{\xi} \geq 0$ does not change the optimal solution of the problem, *i.e.*, we will show that for the optimal solution $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*)$, the inequality $\boldsymbol{\xi}^* \geq 0$ always holds.

(a) Assume that the dataset consists of 3 samples, that $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*)$ is the optimal solution to the problem without the last set of constraints, *i.e.*, $\boldsymbol{\xi} \geq 0$, and that $(\xi^{(3)})^* = 0$. Write down the resulting expression of the loss as a function of $\mathbf{w}^*$, $(\xi^{(1)})^*$ and $(\xi^{(2)})^*$.

(b) Assume that the dataset consists of 3 samples, that $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*)$ is the optimal solution to the problem without the last set of constraints, *i.e.*, $\boldsymbol{\xi} \geq 0$, and that $(\xi^{(3)})^* < 0$. Write-down the expression of the resulting loss and compare it with the one from **part a**. Which of the losses has a larger value?

(c) Consider the general case where $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*)$ is the optimal solution to the problem without the constraints $\boldsymbol{\xi} \geq 0$. Suppose, there exists some $(\xi^{(j)})^* < 0$. Show that $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}})$ with $\hat{\mathbf{w}} = \mathbf{w}^*$, $\hat{b} = b^*$, $\hat{\xi}^{(i)} = (\xi^{(i)})^*$ $(\forall i \neq j)$ and $\hat{\xi}^{(j)} = 0$, is a feasible solution.

(d) Compare the losses obtained for $(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*)$ and $(\hat{\mathbf{w}}, \hat{b}, \hat{\boldsymbol{\xi}})$. Conclude that the optimal solution does not change when removing the constraints $\boldsymbol{\xi} \geq 0$.

(e) After removing the constraints $\boldsymbol{\xi} \geq 0$, we get a simpler program

$$\begin{aligned} \min_{\mathbf{w}, b, \boldsymbol{\xi}} \quad & \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + \tfrac{C}{2}\textstyle\sum_{i=1}^{|\mathcal{D}|}(\xi^{(i)})^2 \\ s.t. \quad & y^{(i)}(\mathbf{w}^T\boldsymbol{x}^{(i)} + b) \geq 1 - \xi^{(i)},\ i \in \{1, \cdots |\mathcal{D}|\}. \end{aligned} \tag{3}$$

Give the Lagrangian of the program above as a function of $\mathbf{w}$, b, $\xi^{(i)}$, $y^{(i)}$, $\boldsymbol{x}^{(i)}$, C, $\mathcal{D}$ and Lagrange multipliers $\alpha^{(i)}$. What's the range of the Lagrange multipliers $\alpha^{(i)}$?

(f) Show that the dual of the program in Eq. (3) is

$$\begin{aligned} \max_{\boldsymbol{\alpha} \geq 0} \quad & -\tfrac{1}{2}\boldsymbol{\alpha}^T(\mathbf{Q} + \mathbf{I}\tfrac{1}{C})\boldsymbol{\alpha} + \mathbf{1}^T\boldsymbol{\alpha} \\ s.t. \quad & \mathbf{y}^T\boldsymbol{\alpha} = 0 \end{aligned}$$

with $Q_{ij} = y_i y_j \mathbf{x}_i^T \mathbf{x}_j$, $\boldsymbol{\alpha} \in \mathbb{R}^{|\mathcal{D}|}$ is a vector of Lagrange multipliers and $\mathbf{y} \in \mathbb{R}^{|\mathcal{D}|}$ a vector of labels.

3. **[Support Vector Machine]**

(a) Recall, a hard-margin support vector machine in the primal form optimizes the following program

$$\min_{\mathbf{w}, b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{s.t. } y^{(i)}(\mathbf{w}^\intercal \mathbf{x}^{(i)} + b) \geq 1 \ \ , \forall (\mathbf{x}^{(i)}, y^{(i)}) \in \mathcal{D} \tag{4}$$

What is the Lagrangian, $L(\mathbf{w}, b, \alpha)$, of the constrained optimization problem in Eq. (4)?

(b) Consider the Lagrangian

$$L(\mathbf{w}, \alpha) := \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \alpha^{(i)}(1 - y^{(i)}\mathbf{w}^\intercal \mathbf{x}^{(i)}) \tag{5}$$

where $\alpha^{(i)}$ are elements of α. *Note: This Lagrangian is not the same as the solution in the previous part.*

Derive the dual program for the Lagrangian given in Eq. (5). Provide all its constraints if any.

(c) Recall that a kernel SVM optimizes the following program

$$\max_{\alpha} \sum_{i=1}^{|\mathcal{D}|} \alpha^{(i)} - \frac{1}{2} \sum_{i,j=1}^{|\mathcal{D}|} \alpha^{(i)}\alpha^{(j)} y^{(i)} y^{(j)} \kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \tag{6}$$

s.t. $\alpha^{(i)} \geq 0$ and $\sum_{i}^{|\mathcal{D}|} \alpha^{(i)} y^{(i)} = 0$

We have chosen the kernel to be

$$\kappa(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\intercal \mathbf{z})^2 + 1$$

Consider the following dataset $\mathcal{D}_2$ in the one-dimensional space; $x^{(i)}, y^{(i)} \in \mathbb{R}$.

i	$x^{(i)}$	$y^{(i)}$
1	$\frac{1}{2}$	+1
2	-1	+1
3	$\sqrt{3}$	-1
4	4	-1

What are the optimal primal parameters, $\mathbf{w}^*$, b^* when optimizing the program in Eq. (6) on the dataset $\mathcal{D}_2$. Note: b is NOT included in the margin or the features (treat it explicitly).
Hint: First, construct a feature vector $\phi(\mathbf{x}) \in \mathbb{R}^2$ such that $\kappa(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^\intercal \phi(\mathbf{z})$ for the given one dimensional dataset. Then use this feature vector to transform the data $\mathcal{D}_2$ into feature space and plot the result. Read of the bias term b and the optimal weight vector $\mathbf{w}^*$.

(d) (Continuing from previous part) Which of the points in $\mathcal{D}_2$ are support vectors? What are $\alpha^{(1)}$ and $\alpha^{(2)}$?
Hint: To find $\alpha^{(2)}$ make use of the relationship between the primal solution and the dual variables, *i.e.*, $\mathbf{w}^* = \sum_{i=1}^{N} \alpha^{(i)} y^{(i)} \phi(\mathbf{x}^{(i)})$. Assume $\mathbf{w}^* = [-4 \quad 0]^T$ if you couldn't solve part (c).

4. **[Multiclass Logistic Regression]**

We are given a dataset $\mathcal{D} = \left\{ \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, 0 \right), \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, 1 \right), \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, 2 \right) \right\}$ containing three pairs (x, y), where each $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$ denotes a 2-dimensional point and $y \in \{0, 1, 2\}$.

We want to train by minimizing the negative log-likelihood the parameters w (includes bias) of a multi-class logistic regression classifier using

$$\min_{w} - \sum_{(x,y)\in\mathcal{D}} \log p(y|x) \qquad \text{where} \qquad p(y|x) = \frac{\exp w_y^\top \begin{bmatrix} x \\ 1 \end{bmatrix}}{\sum_{\hat{y}\in\{0,1,2\}} \exp w_{\hat{y}}^\top \begin{bmatrix} x \\ 1 \end{bmatrix}}. \tag{7}$$

(a) How many parameters do we train, *i.e.*, what's the domain of w? Explain what w_y means and how it relates to w?

(b) Alternatively, we can use the equivalent probability model

$$p(y|x) = \frac{\exp w^\top \psi(x,y)}{\sum_{\hat{y}\in\{0,1,2\}} \exp w^\top \psi(x,\hat{y})}.$$

Explain how we need to construct $\psi(x,y)$ such that $w^\top \psi(x,y) = w_y^\top \begin{bmatrix} x \\ 1 \end{bmatrix}$ $\forall y \in \{0,1,2\}$.

(c) Alternatively, we can use the equivalent probability model

$$p(y|x) = \frac{\exp F(y,w,x)}{\sum_{\hat{y}\in\{0,1,2\}} \exp F(\hat{y},w,x)} \qquad \text{with} \qquad F(y,w,x) = [\mathbf{W}x + b]_y\,,$$

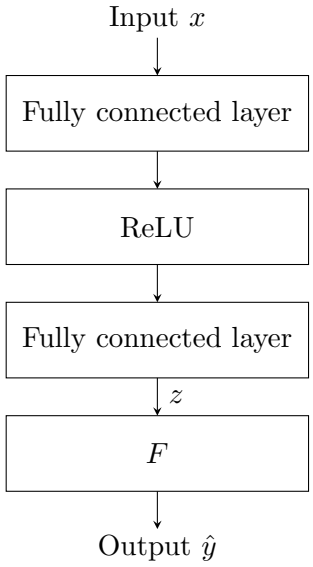
where $\mathbf{W}$ is a matrix of weights and b is a vector of biases. The notation $[a]_y$ extracts the y-th entry from vector a. What are the dimensions of $\mathbf{W}$ and b and how are $\mathbf{W}$ and b related to the originally introduced w?

(d) Assume we are given $\mathbf{W} = \begin{bmatrix} 3 & 0.5 \\ 0 & 1 \\ -1.5 & -1.5 \end{bmatrix}$ and $b = \begin{bmatrix} 0 \\ 0 \\ 1.5 \end{bmatrix}$. Draw the data points, and the lines $[\mathbf{W}x + b]_y = 0$ $\forall y \in \{0,1,2\}$ in x_1-x_2-space and explain whether these weights result in correct prediction for all datapoints in $\mathcal{D}$?

(e) Complete `A4_Multiclass.py`. After optimizing, what values do you obtain for $\mathbf{W}$, b and what probability estimates $p(\hat{y}|x)$ do you obtain for all points $x \in \mathcal{D}$ in the dataset and for all classes $\hat{y} \in \{0,1,2\}$. (**Hint:** a total of nine probability estimates are required.)

5. **[Multiclass Classification via Neural Networks]**

Suppose we use a multi-layer neural network to classify any input image into one of the following three classes: *apple*, *pear* & *orange*. The neural network architecture is summarized in the following figure:



The output $\hat{y} = F(z)$ is a three-dimensional vector $(\Pr(apple|x), \Pr(pear|x), \Pr(orange|x))$, where $\Pr(c|x)$ denotes the probability of x being in class c.

(a) Which function should be used as activation function F, for multiclass classification? (a) logistic (b) softmax (c) ReLU (d) sigmoid. Suppose the input to F is $z = (z_1, z_2, z_3)$, write down the expression of $F(z)$. (Hint: $F(z)$ should sum to 1.)

(b) Suppose we have an alternative activation function G:

$$G(z) = \left(\frac{z_1}{z_1 + z_2 + z_3}, \frac{z_2}{z_1 + z_2 + z_3}, \frac{z_3}{z_1 + z_2 + z_3} \right)$$

which normalizes vector z to sum to 1. Consider the following two inputs $z^{(1)} = (0.01, 0.01, 0.02)$ and $z^{(2)} = (1.01, 1.01, 1.02)$. Use the given inputs to answer whether F and G are *translation invariant*, i.e. the value of the function does **not** change when we add a constant to all its inputs z_i. Use this fact to give an advantage of using F over G. (Hint: you do not need to exactly evaluate the expressions.)

(c) Suppose for an input image, the second fully-connected layer outputs $z = (1, 10^{-5}, 10^{-5})$, which means it is very confident that the image is *apple*, while the true label $y =$ *orange*. Considering this input, give another advantage of using F over G, by evaluating (1) the cross entropy between the true label and classifier prediction $\text{CE}(y, F(z))$, $\text{CE}(y, G(z))$ and (2) their derivatives w.r.t. z_3, where $z = (z_1, z_2, z_3)$.

(d) As an alternative to a multiclass neural network, we can use one-vs-rest multiclass classification, which fits a single-output neural network for each class. Suppose we use the same number of hidden units in the two approaches. Which approach will have faster prediction (output $\hat{y}$ given x) speed? Explain your reason.