University of Illinois Fall 2023

ECE 544: Pattern Recognition
Problem Set 1

Due: Thursday, September 14, 2023, 11:59 pm

1. **[Linear Regression]**

   We are given a dataset $\mathcal{D} = \{(1,1),(2,1)\}$ containing two pairs (x,y), where each $x \in \mathbb{R}, y \in \mathbb{R}$ denotes a real-valued number.

   We want to find the parameters $w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \in \mathbb{R}^2$ of a linear regression model $\hat{y} = w_1 x + w_2$ using

$$\min_w \frac{1}{2} \sum_{(x,y)\in\mathcal{D}} \left(y - w^\top \begin{bmatrix} x \\ 1 \end{bmatrix} \right)^2. \tag{1}$$

   (a) Plot the given dataset and find the optimal w^* by inspection.

   (b) Using general matrix vector notation, the program in Eq. (1) is equivalent to

$$\min_w \frac{1}{2} \|\mathbf{y} - \mathbf{X}w\|_2^2. \tag{2}$$

   Specify the dimensions of the introduced matrix $\mathbf{X}$ and the introduced vector $\mathbf{y}$. Also write down explicitly the matrices and vectors using the values in the given dataset $\mathcal{D}$.

   (c) **Derive** the general analytical solution for the program given in Eq. (2). Also plug in the values for the given dataset $\mathcal{D}$ and compute the solution numerically.

   (d) Numerous ways exist to compute this solution via PyTorch. Read the docs for the functions 'torch.linalg.lstsq', 'torch.linalg.solve', and 'torch.linalg.inv'. Use all three approaches when completing the file `A1_LinearRegression.py` and verify your answer.

   (e) We are now given a dataset $\mathcal{D} = \{(0,0),(1,1),(2,1)\}$ of pairs (x,y) with $x, y \in \mathbb{R}$ for which we want to fit a quadratic model $\hat{y} = w_1 x^2 + w_2 x + w_3$ using the program given in Eq. (2). Specify the dimensions of the matrix $\mathbf{X}$ and the vector $\mathbf{y}$. Also write down explicitly the matrix and vector using the values in the given dataset. Find the optimal solution w^* and draw it together with the dataset into a plot.

   (f) Complete `A1_LinearRegression2.py` and verify your reply for the previous answer. How did you specify the matrix $\mathbf{X}$?

2. **[Regression]**

   Suppose we are given a set of observations $\{(x^{(i)}, y^{(i)})\}$ where $x, y \in \mathbb{R}$ and $i \in \{1, 2, \ldots, N\}$. Consider the following program:

$$\operatorname*{argmin}_{w_1,w_2} \sum_{i=1}^{N} \left(y^{(i)} - w_1 \cdot x^{(i)} - w_2 \right)^2. \tag{3}$$

   (a) What is the minimum number of observations required for a unique solution?

   (b) Suppose we now want to fit a quadratic model to the observed data. Modify the program given in Eq. 3 accordingly. Derive the closed form solution for this case. You may assume that you have a sufficient number of data samples. Use matrix vector notation, i.e., $\mathbf{w}$, $\mathbf{X}$, and $\mathbf{Y}$ and define them carefully.

(c) Briefly describe the problem(s) that we encounter if we were to fit a high degree polynomial to a data that is known to be linear.

(d) The program above (Eq. 3) assumes $x \in \mathbb{R}$. State the program for $\mathbf{x} \in \mathbb{R}^D$ and specify the dimensions of $\mathbf{w}$.

(e) If $\dim(\mathbf{x}) = D > N$, how could the program be modified such that a unique solution can be obtained?

(f) Is there a closed form solution to the new program? If so derive it.

3. **[Softmax Regression]**

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 $y^{(i)} \in \{1, \cdots, K\}$. Here, K denotes the number of classes. The distribution over $y^{(i)}$ is given via

$$p(y^{(i)} = k|\mathbf{x}^{(i)}) = \mu_k(\mathbf{x}^{(i)}) = \frac{e^{\mathbf{w}_k^T \mathbf{x}^{(i)}}}{\sum_{j=1}^K e^{\mathbf{w}_j^T \mathbf{x}^{(i)}}}. \tag{4}$$

(a) Show that the negative conditional log-likelihood $\ell(\mathbf{w}_1, \cdots, \mathbf{w}_K)$ is given by the expression,

$$-\log p(\mathcal{D}) = -\log p(\{y^{(i)}\}_{i=1}^{|\mathcal{D}|} | \{\mathbf{x}^{(i)}\}_{i=1}^{|\mathcal{D}|}) = -\sum_{i=1}^{|\mathcal{D}|} \sum_{k=1}^{K} \mathbb{1}_{\{y^{(i)}=k\}} \mathbf{w}_k^T \mathbf{x}^{(i)} + \sum_{i=1}^{|\mathcal{D}|} \log\left(\sum_{j=1}^{K} e^{\mathbf{w}_j^T \mathbf{x}^{(i)}}\right).$$

Here, $\mathbb{1}_{\{y^{(i)}=k\}}$ is an indicator variable, *i.e.*, $\mathbb{1}_{\{y^{(i)}=k\}}$ is equal to 1 if $(y^{(i)} = k)$ and equal to 0 otherwise. *Show intermediate steps and state any used assumptions.*

(b) We want to minimize the negative log-likelihood. To combat overfitting, we add a regularizer to the objective function. The regularized objective is $\ell_r(\mathbf{w}_1, \cdots, \mathbf{w}_K) = \frac{1}{|\mathcal{D}|}\ell(\mathbf{w}_1, \cdots, \mathbf{w}_K) + \lambda \sum_{k=1}^K \|\mathbf{w}_k\|^2$. Justify that λ should be a strictly positive scalar, *i.e.*, $\lambda > 0$.

(c) Show that the gradient of the regularized loss ℓ_r is

$$\nabla_{\mathbf{w}_k} \ell_r(\mathbf{w}_1, \cdots, \mathbf{w}_K) = 2\lambda \mathbf{w}_k + \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} (\mu_k(\mathbf{x}^{(i)}) - \mathbb{1}_{\{y^{(i)}=k\}}) \mathbf{x}^{(i)}.$$

(d) State the gradient update (formula) for **gradient descent** for the regularized loss ℓ_r. Use a learning rate of α.

4. **[Binary Logistic Regression]**

We are given a dataset $\mathcal{D} = \{(-1, -1), (1, 1), (2, 1)\}$ containing three pairs (x, y), where each $x \in \mathbb{R}$ denotes a real-valued point and $y \in \{-1, +1\}$ is the point's class label.

We want to train the parameters $w \in \mathbb{R}^2$ (*i.e.*, weight w_1 and bias w_2) of a logistic regression model

$$p(y|x) = \frac{1}{1 + \exp\left(-y w^\top \begin{bmatrix} x \\ 1 \end{bmatrix}\right)} \tag{5}$$

using maximum likelihood while assuming the samples in the dataset $\mathcal{D}$ to be i.i.d.

(a) Instead of maximizing the likelihood we commonly minimize the negative log-likelihood. Specify the objective for the model given in Eq. (5). Don't use any regularizer or weight-decay.

(b) Compute the derivative of the negative log-likelihood objective in general (the one specified in the previous question, *i.e.*, no regularizer or weight-decay). Sketch a simple gradient-descent algorithm using pseudo-code (use f for the function value, $g = \nabla_w f$ for the gradient, w for the parameters, and show the update rule).

(c) Implement the algorithm by completing `A2_LogisticRegression.py`. State the code that you implemented. What is the optimal solution w^* that your program found?

(d) If the third datapoint $(2, 1)$ was instead $(10, 1)$, would this influence the bias w_2 much? How about if we had used linear regression to fit $\mathcal{D}$ as opposed to logistic regression? Provide a reason for your answer.

(e) Instead of manually deriving and implementing the gradient we now want to take advantage of PyTorch auto-differentiation. Investigate `A2_LogisticRegression2.py` and complete the update step using the 'optimizer' instance. What code did you add? If you compare the result of `A2_LogisticRegression.py` with that of `A2_LogisticRegression2.py` after an equal number of iterations, what do you realize?

5. **[Binary Classifiers]**

Based on a data set, $D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ..., (\mathbf{x_N}, y_N)\}$, where $\mathbf{x_i} \in \mathbb{R}^d$, $y_i \in \{0, 1\}$, and samples are i.i.d., we want to train a logistic regression model. We define our probabilistic model to have the form:

$$\hat{y}_i = \mathrm{g}(\mathbf{w}^T\mathbf{x_i}) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x_i}}},$$

where $\hat{y}_i$ is the probability given data $\mathbf{x_i}$ and model parameters $\mathbf{w} \in \mathbb{R}^d$. Given this notation we define the probability of predicting y_i via

$$P[Y = y_i | X = \mathbf{x_i}] = (\hat{y}_i)^{y_i} \cdot (1 - \hat{y}_i)^{(1-y_i)}.$$

We want to find the model parameters $\mathbf{w}$, such that the likelihood of the data set D is maximized, which is formulated as

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \left(-\sum_{i=1}^{N} (y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)) \right).$$

(a) Let the program above be referred to as:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} L(y, \mathbf{x}, \mathbf{w})$$

Is $L(y, \mathbf{x}, \mathbf{w})$ convex with respect to $\mathbf{w}$? Prove it is convex or non-convex without using knowledge of convexity for any function. (Hint: use the Hessian.)

(b) Can we find a closed form analytic solution for $\mathbf{w}$? How to train the model $\mathbf{w}$ based on the data set D? State your approach and write down the equation.