CS182: Introduction to Machine Learning
Final Exam

June 19, 2022

I Regression and Probability Estimation [12 points]

We consider the following linear regression model in which y is the sum of a deterministic linear function of x, plus random noise ϵ, i.e.,

$$y = wx + \epsilon, \tag{1}$$

where x is the real-valued input, y is the real-valued output, and w is a single real-valued parameter to be learned. Here ϵ is a real-valued random variable that represents noise which follows a Gaussian distribution with mean 0 and standard deviation σ, that is, $\epsilon \sim \mathcal{N}(0, \sigma^2)$.
Note: the probability density function $f(X)$ of a Gaussion distributed variable $X \sim \mathcal{N}(\mu, \sigma^2)$ takes the form

$$f(X = x) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(x-\mu)^2}{2\sigma^2}). \tag{2}$$

1. **[4 points]** Write down the probability distribution of y conditioned on x and w., i.e. $\Pr(y \mid w, x)$.

2. **[4 points]** Given n *i.i.d.* training examples $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$. Let $\mathcal{Y} = (y_1, ..., y_n)$ and $\mathcal{X} = (x_1, ..., x_n)$, please write down an expression for the conditional data likelihood: $\Pr(\mathcal{Y} \mid \mathcal{X}, w)$.

3. **[4 points]** Suppose a Laplace prior over w with $\mu = 0$ and b (i.e., $w \sim \mathit{Laplace}(0, b)$). Now you need to use MAP(maximum a posterior probability) to estimate w from the training data. Please show that finding the MAP estimate w^* is equivalent to solving the following optimization problem

$$w^* = \operatorname*{argmin}_w \frac{1}{2} \sum_{i=1}^{n} (y_i - wx_i)^2 + c|w|. \tag{3}$$

   Express the regularization parameter c in terms of σ and b.
   Hint: the probability density function $f(X)$ of a Laplace distributed variable $X \sim \mathit{Laplace}(\mu, b)$ takes the form

$$f(X = x) = \frac{1}{2b} \exp(-\frac{|x-\mu|}{b}). \tag{4}$$

II Linear Classification [12 points]

Let X be a d-dimensional binary vector, drawn from one of two classes: P or Q. Assume each element X_i in X is an independent Bernoulli random variable with parameter p_i when X drawn from class P (similarly, with parameter q_i for class Q). That is

$$\begin{aligned} X_i|(Y = P) &\sim Bernoulli(p_i), \qquad 1 \leq i \leq d, \\ X_i|(Y = Q) &\sim Bernoulli(q_i), \qquad 1 \leq i \leq d. \end{aligned}$$

Note: suppose that the values of p_i and q_i and priors $\Pr(Y = P) = \pi_p$ and $\Pr(Y = P) = \pi_q$ are known.

1. **[3 points]** Given a vector $x \in \{0, 1\}^d$, compute the probabilities $\Pr(X = x|Y = P)$ and $\Pr(X = x|Y = Q)$ in terms of class parameters p_i and q_i.

2. **[4 points]** Please write down the equation which holds if and only if x is at the decision boundary of the Bayes' optimal classifier.

3. **[5 points]** The decision boundary derived above is actually linear in x, which can be expressed as:

   $$\{x \in \{0, 1\}^d | w^T x + b = 0\},$$

   for some vector w and scalar b. Please find expressions for w and b in terms of priors (π_p and π_q) and class parameters (p_i and q_i).

III Graphical Model [12 points]

We have a Bayesian network shown in Fig. 1, in which $X_1, X_2, ..., X_8$ are eight boolean random variables. Please answer the following questions.
Note: correct answers without proof will get 0 point.

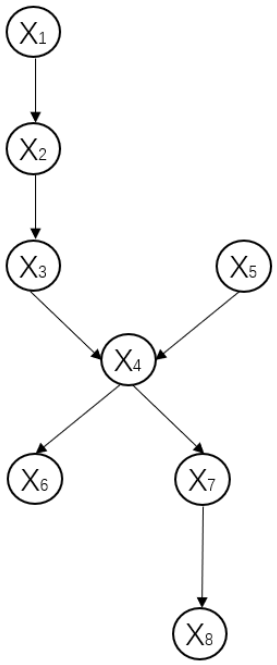

Figure 1: The Bayesian network with eight variables.

1. **[3 points]** Now we have known probabilities for some random variables. For X_1, we have $\Pr(x_1) = 0.7$. For X_2, we have $\Pr(x_2|x_1) = 0.6$ and $\Pr(x_2|\neg x_1) = 0.3$. For X_3, we have $\Pr(x_3|x_2) = 0.4$ and $\Pr(x_3|\neg x_2) = 0.8$. Apply the method of inference to calculate marginal probability $\Pr(\neg x_3)$.
   Note: please round your results to 3 decimal places.

2. **[3 points]** Using the same probabilities for X_1, X_2 and X_3 as above, and apply the method of inference to calculate conditional probability $\Pr(\neg x_2|\neg x_3)$.
   Note: please round your results to 3 decimal places.

3. **[3 points]** Prove that $X_1 \perp\!\!\!\perp X_3|X_2$ without using D-separation.

4. **[3 points]** Discuss whether the statement, $X_1 \perp\!\!\!\perp X_5|X_6$, is true or not, and explain the reason based on D-separation.

IV Expectation-Maximization [10 points]

Given a Bayesian network with four discrete variables $\{A, B, C, D\}$, where $\{A, C, D\}$ are boolean variables and $B \in \{0, 1, 2\}$. Suppose that $\{A, C, D\}$ are observed variables and $\{B\}$ is a latent variable. Now we implement EM algorithm for this model. Suppose there are K observations in total. $(\{a_k, c_k, d_k\}_{k=1}^{K})$.

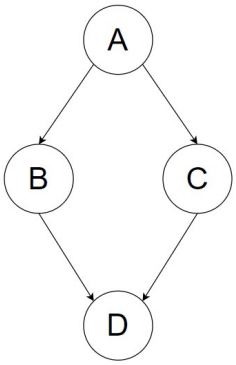

Figure 2: The Bayesian network with four discrete variables $\{A, B, C, D\}$.

1. **[4 points]** Derive the E-step.

2. **[6 points]** Derive the M-step, and update parameters for the Bayesian network

V Support Vector Machines [12 points]

Support vector machines (SVM) are supervised learning models, that directly optimize for the maximum margin separator. Fig. 3 shows an example of maximum margin separator over a dataset $S = \{(x_i, y_i)\}_{i=1}^{n}$, in which $x_i \in \mathbb{R}^2$ and $y_i \in \{-1, 1\}$ denote the i-th sample and the i-th label ($\forall i$), respectively. For simplicity, here we assume that the dataset S has been standardized, and thus the bias can be omitted in the linear model. In Fig. 3, "+" and "-" denote the samples with labels "1" and "-1", respectively, and $\mathbf{w}$ is the normal vector of the maximum margin separator $\mathbf{w}^\top x = 0$. You need to derive the optimization problem of SVM in the linearly separable case.

Note: correctly giving the results without detailed derivation will get 0 point.

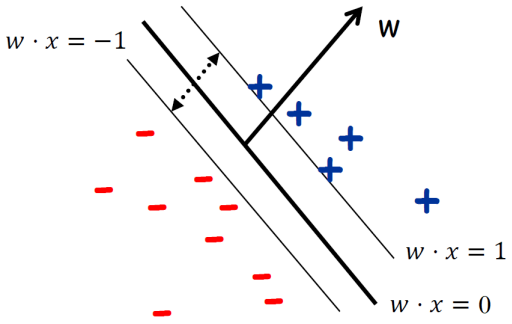

Figure 3: Maximum margin separator in the linearly separable case.

1. **[5 points]** Derive the constraint optimization problem of SVM in the separable case shown in Fig. 3.

2. **[5 points]** Derive the dual problem of the above primal problem based on K.K.T. conditions.

3. **[2 points]** Kernel functions implicitly define some mapping function $\phi(\cdot)$ that transforms an input instance $x \in \mathbb{R}^d$ to a high or even infinite dimensional feature space Q, by giving the form of dot product in $Q : k(x_i, x_j) = \phi(x_i)\dot{\phi}(x_j)$. Please kernelize the dual problem, in order to learn a non-linear classifier.

VI Clustering [10 points]

Given six data points in 2D space (shown in Table 1) and two initial cluster centers $c_1 = (0, 1), c_2 = (0, -1)$, please answer the following questions.
Note: correct answers without detailed derivation will get 0 point.

i	x	y
1	-2	1
2	0	2
3	2	1
4	2	-1
5	0	-2
6	-2	-1

Table 1: Six input data points

1. **[5 points]** Please use k-means algorithm to cluster the given points into two groups.
2. **[5 points]** Please give the center points for the two groups after the algorithm converges.

VII Dimensionality Reduction [**12 points**]

Given three data points in 2D space: $x_1 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, x_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, x_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, please answer the following questions:

Note: correct answers without detailed derivation will get 0 point.

1. [**4 points**] What are the first and second principal components?
2. [**4 points**] If we project the original data points on the new coordinate system represented by the principal components, what are their coordinates?
3. [**4 points**] What is the variance of the data in each direction? Verify that it is equal to the total variance of the origin data.

VIII Neural Networks [12 points]

As shown in Fig.4, we have a feed-forward neural network with two hidden-layer nodes and one output node, and x_1 and x_2 are two inputs. For simplicity, the bias b is omitted here. For the following questions, assume the learning rate η in gradient descent is fixed by $\eta = 0.1$. Both hidden and output units use the same activation function $g(\cdot)$.

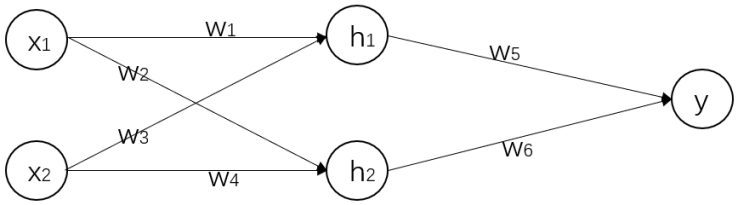

Figure 4: The Neural network with one hidden layer.

1. **[4 points]** Express the output y_{output} in terms of inputs x_1, x_2, weights $w_1, w_2, w_3, w_4, w_5, w_6$ and the activation function g.

2. **[8 points]** Assume that we have one input $\{x_1 = 1, x_2 = 1\}$ and the real target of it is $y_{\text{target}} = 1$. The initial values of $w_1^{(0)}, w_2^{(0)}, w_3^{(0)}, w_4^{(0)}, w_5^{(0)}, w_6^{(0)}$ are 1,2,-1,$\frac{1}{2}$,-2 and 1, respectively. And the loss on the given example is defined as $L = \frac{1}{2}(y_{\text{target}} - y_{\text{output}})^2$. Suppose that the sigmoid activation function $g(z) = \frac{1}{1+e^{-z}}$ is used.
   Note: please round your results to 3 decimal places.

   (1) **[3 points]** Without any optimization, calculate the output h_1, h_2 and y_{output} on the given example.

   (2) **[5 points]** Compute the updated weights $w_1^{(1)}, w_2^{(1)}, w_3^{(1)}, w_4^{(1)}, w_5^{(1)}, w_6^{(1)}$ by performing ONE step of gradient descent. Show all steps in your calculation.

IX Convolutional Neural Networks [8 points]

Convolutional neural networks are designed to process 2D features instead of the 1D ones in multi-layer perceptron (MLP).

1. **[4 points]** Please calculate the feature map based on 2D convolution, if you are given the following 5×5 image matrix in Table 2 and 2×2 kernel matrix in Table 3. (stride = 1, no padding)

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Table 2: 5×5 image matrix.

1	0
0	1

Table 3: 2×2 kernel matrix.

2. **[4 points]** Based on the above result, calculate the feature maps after max-pooling and average-pooling, respectively. (both pooling with 2×2 filters and stride = 2)