School of Information Science and Technology

ShanghaiTech University

Final exam; Spring semester; 2019-2020 academic year

SI151: Optimization and Machine Learning

Instructor: Lu Sun
10:15-11:55, June 29, 2020

Your name: (last name)____________________________, (first name)____________________

Your email ID: __________________________________@shanghaitech.edu.cn

Your student ID: ____

School: ____

INSTRUCTIONS:

- You have 100 minutes (10:15-11:55) to complete the exam.
- Your exam will not be graded unless you complete the cover sheet, and turn in both this cover sheet and the exam book.
- Mark your answers on the exam itself. We will not grade answers written on scratch paper.

STOP! Do not turn this page until the instructor tells you to do so.

Important: Verify that your exam book has 16 pages.

Do NOT write in this section.

Problem	1	2	3	4	5	6	7	8	Total
Max	12	12	12	12	10	16	13	13	100
Points									

Signature of Examiner:
Date:

Signature of Reviewer:
Date:

1. (12 points) *Regression function and least squares.*
Given the input variables $X \in \mathbb{R}^d$ and response variable $Y \in \mathbb{R}$, the Expected Prediction Error (EPE) is defined by

$$\text{EPE}(f) = \mathbb{E}[L(Y, f(X))], \tag{1}$$

where $\mathbb{E}(\cdot)$ denotes the expectation over the joint distribution $\Pr(X, Y)$, and $L(Y, f(X))$ is the squared error loss function

$$L(Y, f(X)) = (Y - f(X))^2, \tag{2}$$

measuring the difference between the observed Y and estimated function $f(X)$.

(a) [5pts] Based on the assumption of linearity, we can approximate $f(X)$ by a linear model $X^\top \beta$. Please derive the linear estimator β by minimizing $\text{EPE}(f)$ w.r.t. β.

(b) [5pts] Given a dataset $\{(x_i, y_i)\}_{i=1}^n$, where x_i and y_i denote the i-th sample and the i-th label ($\forall i$), respectively, please derive the least squares (LS) estimator by minimizing the residual sum of squares (RSS):

$$\text{RSS}(\beta) = \sum_{i=1}^{n} (y_i - x_i^\top \beta)^2 = \|\mathbf{y} - \mathbf{X}\beta\|_2^2, \tag{3}$$

where $\|\cdot\|_2$ denotes the ℓ_2 norm, $\mathbf{y} = [y_1, y_2, ..., y_n]^\top \in \mathbb{R}^n$, and $\mathbf{X} = [x_1, x_2, ..., x_n]^\top \in \mathbb{R}^{n \times d}$ with full column rank.

(c) [2pts] Explain how the LS estimator in (b) approximates the linear estimator in (a).

Your answer:

2. (12 points) *Linear regression and parameter estimation.*
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.,

$$y = wx + \epsilon, \tag{4}$$

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, and 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 Gaussian distributed variable $X \sim \mathcal{N}(\mu, \sigma^2)$ takes the form

$$f(X = x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \tag{5}$$

where μ and σ^2 denote mean and variance, respectively.

(a) [2pts] Write down the probability distribution of y conditioned on x and w.

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

(c) [6pts] The primary target of this question is for you to derive the expression for obtaining a MAP (maximum a posterior probability) estimate of w from the training data. Suppose a Gaussian prior over w with mean 0 and standard deviation τ (i.e., $w \sim \mathcal{N}(0, \tau^2))$. Please show that finding the MAP estimate w^* is equivalent to solving the following optimization problem

$$w^* = \arg\min_w \frac{1}{2}\sum_{i=1}^{n}(y_i - wx_i)^2 + \frac{\lambda}{2}w^2. \tag{6}$$

Also explicitly express the regularization parameter λ in terms of σ and τ.

Your answer:

3. (12 points) *Linear classification and decision boundary.*
A binary image is a digital image where each pixel has only possible values: zero (white) or one (black). A binary image, which consists of a grid of pixels, can therefore naturally be represented as a vector with entries in $\{0,1\}$.

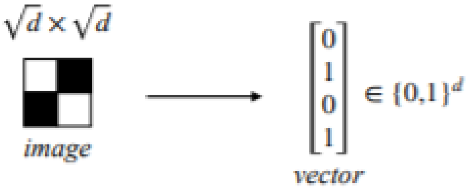

$\sqrt{d} \times \sqrt{d}$
image
$\begin{bmatrix}0\\1\\0\\1\end{bmatrix} \in \{0,1\}^d$
vector

Figure 1: Example of vectorization of a binary image.

In this problem, we consider a classification scheme based on a simple generative model. Let X be a random binary image, represented as a d-dimensional binary vector, drawn from one of two classes: P or Q. Assume every pixel X_i is an independent Bernoulli random variable with parameter p_i and q_i when drawn from classes P and Q respectively.

$$X_i|Y = P \sim \text{Bernoulli}(p_i), \qquad \text{independently for all } 1 \leq i \leq d, \tag{7}$$

$$X_i|Y = Q \sim \text{Bernoulli}(q_i), \qquad \text{independently for all } 1 \leq i \leq d. \tag{8}$$

Note: for this problem, we focus on the ideal case, where the true values of p_i and q_i , along with priors π_p and π_q, are known.

(a) [2pts] Given an image $x \in \{0,1\}^d$, compute the probabilities $\Pr(X = x|Y = P)$ and $\Pr(X = x|Y = Q)$ in terms of the priors, image pixels and/or class parameters. Your answer must be a single expression for each probability.

(b) [4pts] In terms of the probabilities above, write an equation which holds if and only if x is at the decision boundary of the Bayes' optimal classifier. No simplification is necessary for full credit.

(c) [6pts] It turns out that the decision boundary derived above is actually linear in the features of x, so for some vectors w and scalar b, it can be succinctly expressed as:

$$\{x \in \{0,1\}^d | w^T x + b = 0\}. \tag{9}$$

Find the entries of the vector w and value of b in terms of class priors and parameters.

Your answer:

4. (12 points) *Representation by Bayesian network.*
The Bayesian network shown in Fig. 2 represents the joint probability distribution of eight boolean random variables, $X_1, X_2, ..., X_8$. Please answer the following questions.
Note: correct answers without proof will get 0 point.

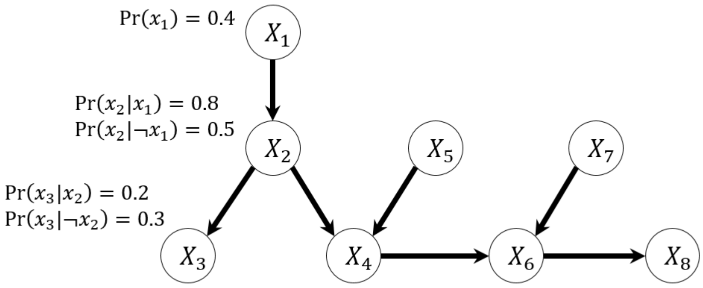


Figure 2: Bayesian network with eight boolean random variables.

(a) [4pts] Apply the method of inference to calculate marginal probability $\Pr(\neg x_3)$.

(b) [4pts] Apply the method of inference to calculate conditional probability $\Pr(x_2 \mid \neg x_3)$.

(c) [2pts] Validate the statement $X_1, X_3 \perp\!\!\!\perp X_7 \mid X_8$.

(d) [2pts] Validate the statement $X_5 \perp\!\!\!\perp X_7 \mid \varnothing$.

Your answer:

5. (10 points) *VC dimension and sample complexity.*
The VC dimension, $\mathrm{VC}(H)$, of hypothesis space H defined over instance space $\mathcal{X}$, is the size of the largest number of points (in some configuration) that can be shattered by H. Suppose with probability $(1-\delta)$, a PAC learner outputs a hypothesis within error ϵ of the best possible hypothesis in H. It can be shown that the lower bound on the number of training examples m sufficient for successful learning, stated in terms of $\mathrm{VC}(H)$ is

$$m \geq \frac{1}{\epsilon}(4\log_2 \frac{2}{\delta} + 8 \times \mathrm{VC}(H) \log_2 \frac{13}{\epsilon}). \tag{10}$$

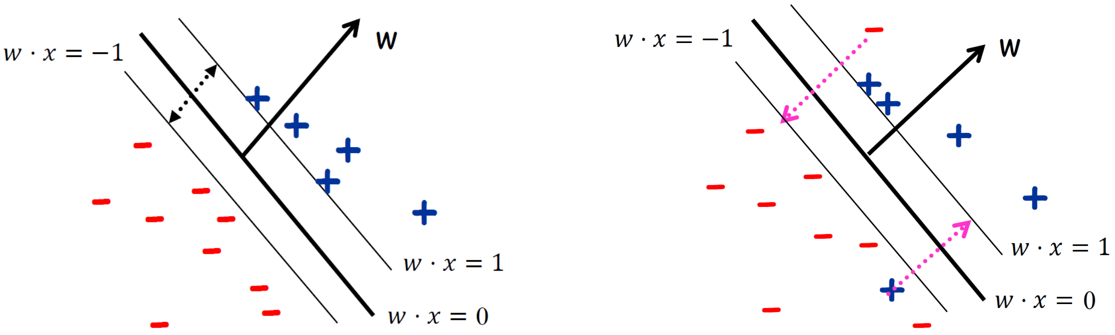
Consider a learning problem in which $\mathcal{X} = \mathbb{R}$ is the set of real numbers, and the hypothesis space is the set of intervals $H = \{(a < x < b) \mid a, b \in \mathbb{R}\}$, which labels points inside the interval as positive, and negative otherwise.
Note: correct answers without proof will get 0 point.

(a) [5pts] What is the VC dimension of H?

(b) [5pts] What is the probability that a hypothesis consistent with m examples will have error at least ϵ?

Your answer:

6. (16 points) *SVM, duality and kernel methods.*
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, in both separable case and non-separable case. 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 $\mathbf{w}^\top x = 0$. In this problem, you need to derive the linear optimization problem of SVM in both primal and dual forms, and finally extend it for non-linear classification.
Note: correctly giving the results without detailed derivation will get 0 point.



(a) Separable case

(b) Non-separable case

Figure 3: Maximum margin separator.

(a) [4pts] Derive the constraint optimization problem of SVM in the separable case shown in Fig. 3(a).

(b) [2pts] Extend the results in (a) to handle the non-separable case shown in Fig. 3(b).

(c) [3pts] Determine the convexity of the problem in (b), and explain whether strong duality holds.

(d) [5pts] Derive the dual problem of the original problem in (b) based on K.K.T. conditions.

(e) [2pts] Extend the linear model in (d) for non-linear classification by the kernel trick.

Your answer:

7. (13 points) *Neural network and backpropagation.*
Figure 4 shows a 2-layer, feed-forward neural network with two hidden-layer nodes and one output node. x_1 and x_2 are the two inputs. For the following questions, assume the learning rate η in gradient descent is fixed by $\eta = 0.1$. Each node also has a bias input value of +1. Assume there is a sigmoid activation function at the hidden layer nodes and at the output layer node. A sigmoid activation function takes the form: $g(z) = \frac{1}{1+e^{-z}}$, where $z = \sum_{j=1}^{d} w_j x_j$ and w_j is the jth incoming weight to a node, x_j is the jth incoming input value, and d is the number of incoming edges to the node.
Note: please round your results to 3 decimal places.

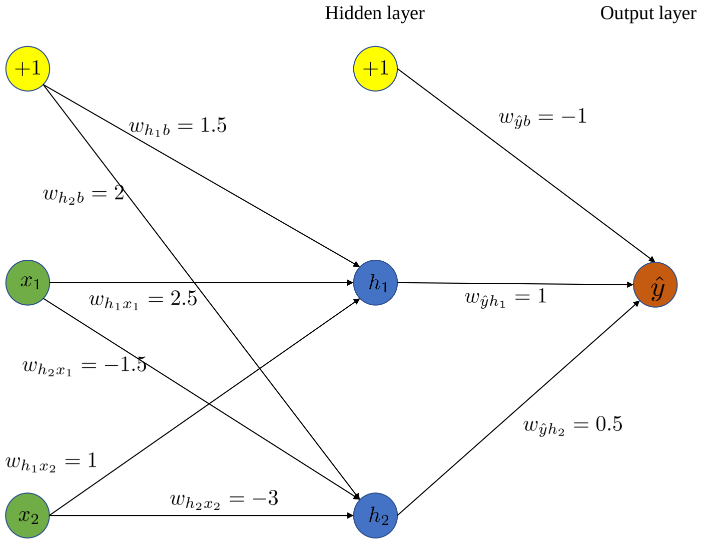


Figure 4: The neural network architecture.

(a) [5pts] Calculate the output values at nodes h_1, h_2 and $\hat{y}$ of this network for input $\{x_1 = 0, x_2 = 1\}$. Show all steps in your calculation. ($e \approx 2.7813$, $e^{-0.0586} \approx 0.9431$, $e^{-2.5} \approx 0.0821$.)

(b) [8pts] Compute one step of the backpropagation algorithm for a given example with input $\{x_1 = 0, x_2 = 1\}$ and target output $y = 1$. The network output is the real-valued output of the sigmoid function, so the error on the given example is defined as $E = \frac{1}{2}(y - \hat{y})^2$ where $\hat{y}$ is the real-valued network output of that example at the output node, and y is the integer-valued target output for that example. You are asked to compute the updated weights for the hidden layer h_1 and the output layer (note that there are six updated weights in total (i.e., the three incoming weights to node h_1 and the three incoming weights to node $\hat{y}$)) by performing ONE step of gradient descent. Show all steps in your calculation.

Your answer:

8. (13 points) *Convex sets and convex functions.*
In this problem, you should first write down whether the set or the function is convex, concave or neither, then you should either prove the set or the function is convex or provide an example to show that it's not convex. Correctly guessing whether the set is convex, concave or neither without proof will get 0 point.
Note: here we use $\mathbb{S}^n$ to denote the set of symmetric matrices in $\mathbb{R}^{n\times n}$, and $\mathbb{S}^n_+$ to denote the set of positive semi-definite symmetric matrices in $\mathbb{R}^{n\times n}$.

(a) [3pts] Determine the convexity of set $\mathcal{C}$:

$$\mathcal{C} = \{\mathbf{A} \in \mathbb{S}^n | \lambda_{\min}(\mathbf{A}) \geq 2\}, \tag{11}$$

where $\lambda_{\min}(\mathbf{A})$ refers to the minimum eigenvalue of $\mathbf{A}$, i.e.,

$$\lambda_{\min}(\mathbf{A}) = \min_{\|u\|_2=1} u^\top \mathbf{A} u. \tag{12}$$

(b) [4pts] Let $\mathcal{H}(w)$ denote the hyperplane with normal direction $w \in \mathbb{R}^d$, that is

$$\mathcal{H}(w) = \{x \in \mathbb{R}^d | x^\top w = 0\}. \tag{13}$$

Let $P : \mathbb{R}^d \to \mathbb{R}^d$ be given by

$$P(x) = \arg\min_{y\in\mathcal{H}(w)} \|y - x\|_2. \tag{14}$$

Determine the convexity of set $\mathcal{C}$:

$$\mathcal{C} = \{P(x) | x \in \mathcal{B}\}, \tag{15}$$

where $\mathcal{B} = \{x \in \mathbb{R}^d | \ \|x\|_2 \leq 1\}$.

(c) [6pts] Given a set of labeled data $S_\ell = \{(x_i, y_i)\}_{i=1}^n$, the ℓ_1-regularized support vector machines (SVM) considers the following unconstraint optimization problem:

$$\min_{\mathbf{w}, w_0} \sum_{x_i \in S_\ell} \left(1 - y_i(\mathbf{w}^\top x_i + w_0)\right)_+ + \gamma \|\mathbf{w}\|_1, \tag{16}$$

where $\mathbf{w} \in \mathbb{R}^d$ is the vector of model parameters, $w_0 \in \mathbb{R}$ is the bias, and $(\cdot)_+ = \max\{0, \cdot\}$. Suppose that additional unlabeled data $S_u = \{x_i\}_{i=1}^m$ is presented, and now we enable to extend (16) for handling semi-supervised learning by

$$\min_{\mathbf{w}, w_0} \sum_{x_i \in S_\ell} \left(1 - y_i(\mathbf{w}^\top x_i + w_0)\right)_+ + \gamma \|\mathbf{w}\|_1 + \lambda \mathbf{w}^\top \mathbf{X}^\top \mathbf{L} \mathbf{X} \mathbf{w}, \tag{17}$$

where $\gamma, \lambda > 0$ are regularization parameters. In (17), the term $\mathbf{w}^\top \mathbf{X}^\top \mathbf{L} \mathbf{X} \mathbf{w}$ is *manifold regularization*, in which $\mathbf{X} \in \mathbb{R}^{(n+m)\times d}$ is the data matrix composed of both labeled and unlabeled data, and $\mathbf{L} \in \mathbb{S}^{(n+m)}$ denotes the normalized Laplacian matrix. Based on the fact that $\mathbf{X}^\top \mathbf{L} \mathbf{X} \in \mathbb{S}^d_+$, please determine the convexity of the objective function of (17).

Your answer: