CM146 Midterm

Richard Sun

TOTAL POINTS

55 / 60

QUESTION 1

1 Machine Learning Basics **6 / 6**

✓ - 0 pts Correct

QUESTION 2

2 Logistic Regression **4 / 4**

✓ - 0 pts Correct

QUESTION 3

3 True/False **12 / 12**

✓ - 0 pts Correct

QUESTION 4

Multiple Choice 7 pts

4.1 **1 / 2**

✓ - 1 pts circle incorrect answer

4.2 **2 / 2**

✓ - 0 pts Correct

4.3 **2 / 3**

✓ - 1 pts (a) wrong

QUESTION 5

5 Maximum Likelihood **5 / 5**

✓ - 0 pts Correct

QUESTION 6

Decision Trees 10 pts

6.1 Entropy Y **1 / 1**

✓ - 0 pts Correct

6.2 Information Gain **4 / 4**

✓ - 0 pts Correct

6.3 Root split **1 / 1**

✓ - 0 pts Correct

6.4 Zero training error tree **2 / 2**

✓ - 0 pts Correct

6.5 Change instance **2 / 2**

✓ - 0 pts Correct

QUESTION 7

Weighted Linear Regression 16 pts

7.1 objective function **3 / 3**

✓ - 0 pts Correct

7.2 optimal value **2 / 5**

✓ - 3 pts wrong intermediate steps

7.3 OLS log likelihood **3 / 3**

✓ - 0 pts Correct

7.4 MLE; variance and weight relation **5 / 5**

✓ - 0 pts Correct

CM 146 — Machine Learning: Midterm

Fall 2017

Name: Richard Sun

UID: 904444918

Instructions:

1. This exam is CLOSED BOOK and CLOSED NOTES.
2. You may use scratch paper if needed.
3. The time limit for the exam is 1hour, 45 minutes.
4. Mark your answers ON THE EXAM ITSELF. If you make a mess, clearly indicate your final answer (box it).
5. For true/false questions, CIRCLE True OR False and provide a brief justification for full credit.
6. Unless otherwise instructed, for multiple-choice questions, CIRCLE ALL CORRECT CHOICES (in some cases, there may be more than one) and provide a brief justification if the question asks for one.
7. If you think something about a question is open to interpretation, feel free to ask the instructor or make a note on the exam.

Q	Problem	Points	Score
1	ML basics	6	
2	Application	4	
3	True/False	12	
4	Multiple choice	7	
5	Maximum likelihood	5	
6	Decision Trees	10	
7	Regression	16	
Total		60	

1. (6 pts) **Machine Learning Basics**

   (a) (2 pts) Consider supervised and unsupervised learning. What is the main difference in the inputs and the goals?

   In supervised learning, the input consists of examples and labels. The goal is to learn the correct label for new examples.

   In unsupervised learning, the input does not have labels. The goal is to learn the structure of the data, such as what groups there are in the data.

   (b) (2 pts) What is the main difference between classification and regression?

   In classification, the goal has discrete values, such as yes or no.

   In regression, the goal is continuous, such as a temperature.

   (c) (2 pts) What is the motivation to separate the available data into training and test data?

   Test data is needed to verify the effectiveness of the model.

   If there was only training data and the model had very low training error, it could still perform poorly in general due to overfitting.

2. (4 pts) **Application** Suppose you are given a dataset of cellular images from patients with and without cancer.

   (a) (2 pts) Consider the models that we have discussed in lecture: decision trees, k-NN, logistic regression, perceptrons. If you are required to train a model that predicts the probability that the patient has cancer, which of these would you prefer, and why?

   Logistic regression. The other models use classification, which is not useful for getting a probability. Logistic regression uses the sigmoid function in its cost function. The sigmoid function is bounded between 0 and 1, so its value can be interpreted as a probability.

   (b) (2 pts) A model that attains 100% accuracy on the training set and 70% accuracy on the test set is better than a model that attains 80% accuracy on the training set and 75% accuracy on the test set.

   True [✓ False]

   The test accuracy is better than the training accuracy for evaluating a model's performance. 100% training accuracy may indicate overfitting.

True/False

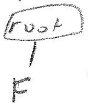
3. (2 pts) You are given a training dataset with attributes $A_1, \ldots, A_m$ and instances $x^{(1)}, \ldots, x^{(n)}$ and you use the ID3 algorithm to build a decision tree D_1. You then take one of the instances, add a copy of it to the training set (so your new training set will have $n + 1$ instances), and rerun the decision tree learning algorithm (with the same random seed) to create D_2. D_1 and D_2 are necessarily identical decision trees.

True [✓ False]

Counterexample! Suppose maxdepth=0. Given a random seed, a tree for

x	y
T	T
F	F

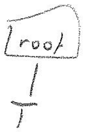
is



. But the tree for

x	y
T	T
T	T
F	F

is



.

4. (2 pts) Stochastic Gradient Descent is faster per iteration than Batch Gradient Descent.

[✓ True] False

One iteration of Stochastic Gradient Descent runs in $O(D)$ time.
One iteration of Batch Gradient Descent runs in $O(ND)$ time.

D = # features
N = # examples

5. (2 pts) You run the PerceptronTrain algorithm with $maxIter = 100$. The algorithm terminates at the end of 100 iterations with a classifier that attains a training error of 1%. This means that the training data is not linearly separable.

True [✓ False]

The algorithm terminated due to reaching max iterations. It is possible that the classifier can have 0% training error with a higher max Iter. If so, the data is linearly separable.

6. (2 pts) We want to learn a non-linear regression function to predict y from $\boldsymbol{x}$ where $y \in \mathbb{R}, \boldsymbol{x} \in \mathbb{R}^D$ given training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$. To do so, we transform $\boldsymbol{x}$ by a function $\boldsymbol{\phi}(\boldsymbol{x})$ and minimize the residual sum of squares objective function on the transformed features: $\sum_{i=1}^n \left(y_i - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\phi}(\boldsymbol{x}_i)\right)^2$. This optimization problem is convex.

[✓ True] False

$(x^2-2x)^2$

$x^4-2x^3+x^2$

$4x^3-6x^2+2x$

$12x^2-12x+2$

Transforming the data makes $\theta^T \phi(x)$ linear, so it is convex.

7. (2 pts) We want to use 1-Nearest Neighbors (1-NN) to classify houses into one of two classes (cheap vs expensive) given a single feature that measures the area of the house. The predictions made by the 1-NN classifier data <u>can change</u> if the area of the house is measured in square metres instead of square feet. (You can neglect the effect of ties *i.e.*, two training instances that are both nearest neighbors to a test instance.)

True [✓ False]

The units of the features are important for determining relative importance to other features. Since there is only one feature, the units do not affect the classification.

8. (2 pts) You run gradient descent to minimize the function $f(x) = (2x-3)^2$. Assume the step size has been chosen appropriately and you run gradient descent till convergence. Then gradient descent will return the global minimum of f.

[✓ True] False

$f'(x) = 2(2x-3)(2) = 8x-6$

$f''(x) = 8 > 0$, so f is convex and has no suboptimal minimums.

Multiple choice

9. (2 pts) In k-nearest neighbor classification, which of the following statements are true? (circle all that are correct)

   [✓ (a)] The decision boundary is smoother with smaller values of k.
   [✓ (b)] k-NN does not require any parameters to be learned in the training step (for a fixed value of k and a fixed distance function).
   [✓ (c)] If we set k equal to the number of instances in the training data, k-NN will predict the same class for any input. ↑ underfit
   (d) For larger values of k, it is more likely that the classifier will overfit than underfit.

10. (2 pts) Assume we are given a set of one-dimensional inputs and their corresponding output (that is, a set of $\{(x_i, y_i)\}, x_i \in \mathbb{R}, y_i \in \mathbb{R}$). We would like to compare the following two models on our input dataset where $\theta \in \mathbb{R}$:

$$\mathrm{A} : y = \theta^2 x$$
$$\mathrm{B} : y = \theta x$$

For each model, we split into training and testing set to evaluate the learned model. Which of the following is correct? Choose the answer that best describes the outcome, and provide justification.

   (a) There are datasets for which A would be more *accurate* than B.
   [✓ (b)] There are datasets for which B would be more *accurate* than A.
   (c) Both (a) and (b) are correct.
   (d) They would perform equally well on all datasets.

Suppose the dataset is $\{(1, -1)\}$. Then B will be accurate with $\theta = -1$ $(y = -x)$. However, no value of θ will make A accurate.

$A: -1 = \theta^2 (1)$

$\theta^2 = \sqrt{-1}$

So, (b) is correct.

Let $y = c^2 x$, $c \in \mathbb{R}$ be the best model for A. Then $y = \theta x$, where $\theta = c^2$, is a model for B that performs equally well. Thus, (a) is incorrect.

11. (3 pts) If your model is overfitting, increasing the training set size (by drawing more instances from the underlying distribution) will tend to result in which of the following? (circle the best answer for each)

   (a) training error will ... increase / decrease / [✓ unknown]
   (b) test error will ... increase / [✓ decrease] / unknown
   (c) overfitting will ... increase / [✓ decrease] / unknown

For these problems, you must show your work to receive credit!

Maximum likelihood

12. We observe the following data consisting of four independent random variables $X_n, n \in \{1, \ldots, 4\}$ drawn from the same Bernoulli distribution with parameter θ (*i.e.*, $P(X_n = 1) = \theta$): $(X_1, X_2, X_3, X_4) = (1, 1, 0, 1)$.

    (a) Give an expression for the log likelihood $l(\theta)$ as a function of θ given this specific dataset. [2 pts]

$$
\begin{aligned}
L(\theta) &= P(X_1 = 1) P(X_2 = 1) P(X_3 = 0) P(X_4 = 1) \\
&= \theta\, \theta (1-\theta) \theta \\
&= \theta^3 (1-\theta) \\
l(\theta) = \log L(\theta) &= \log(\theta^3(1-\theta)) \\
&= \log \theta^3 + \log(1-\theta) \\
&= 3 \log \theta + \log(1-\theta)
\end{aligned}
$$

    (b) Give an expression for the derivative of the log likelihood for this specific dataset. [2 pts]

$$\frac{1}{\theta^3(1-\theta)} (3\theta^2 - 4\theta^3)$$

$$
\begin{aligned}
\frac{\partial l(\theta)}{\partial \theta} &= 3 \frac{1}{\theta} + \frac{1}{1-\theta}(-1) \\
&= \frac{3}{\theta} - \frac{1}{1-\theta}
\end{aligned}
$$

(c) What is the maximum likelihood estimate $\hat{\theta}$ of θ? [1 pts]

$$\frac{\partial \ell(\theta)}{\partial \theta} = 0$$

$$\frac{3}{\theta} - \frac{1}{1-\theta} = 0$$

$$\frac{3}{\theta} = \frac{1}{1-\theta}$$

$$3(1-\theta) = \theta$$

$$3 - 3\theta = \theta$$

$$3 = 4\theta$$

$$\hat{\theta} = \frac{3}{4}$$

Decision Trees

13. We would like to learn a decision tree given the following pairs of training instances with attributes (a_1, a_2) and target variable Y.

Instance number	a_1	a_2	Y
1	T	T	T
2	T	T	T
3	T	F	F
4	F	F	T
5	F	T	F
6	F	T	F

For reference, for a random variable X that takes on two values with probability p and $1 - p$, here are some values of the entropy function (we use **log to the base 2** in this question):

$$p = \frac{1}{2} : H(X) = 1 \qquad p \in \{\frac{1}{3}, \frac{2}{3}\} : H(X) \approx .92$$

(a) What is the entropy of Y? [1 pts]

$p = \frac{1}{2}$, $\boxed{H(X) = 1}$

3 T

3 F

(b) What is the information gain of each of the attributes a_1 and a_2 relative to Y? [4 pts]

3T
3F

$a_1 = T$: 2 T, 1 F

$a_1 = F$: 1 T, 2 F

$$H(X) = \frac{3}{6} H(p \in \{\tfrac{1}{3}, \tfrac{2}{3}\}) + \frac{3}{6} H(p \in \{\tfrac{1}{3}, \tfrac{2}{3}\})$$

$$= \frac{1}{2}(.92) + \frac{1}{2}(.92)$$

$$= .92$$

Gain $= 1 - .92 = 0.08$ for a_1

4T
2F

$a_2 = T$: 2T, 2F

$a_2 = F$: 1T, 1F

$$H(X) = \frac{4}{6} H(p = \tfrac{1}{2}) + \frac{2}{6} H(p = \tfrac{1}{2})$$

$$= \frac{4}{6}(1) + \frac{2}{6}(1)$$

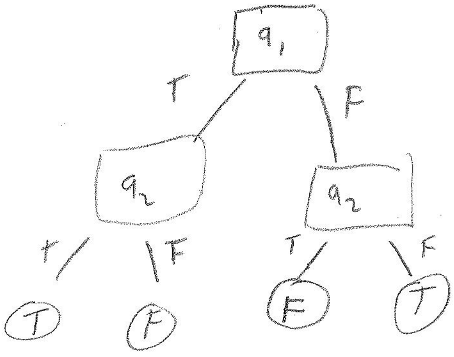
$$= 1$$

Gain $= 1 - 1 = 0$ for a_2

(c) Using information gain, which attribute will the ID3 decision tree learning algorithm choose as the root? [1 pts]

a_1, since it has the higher information gain.

(d) Construct a decision tree with zero training error on this training data. [2 pts]



(e) Change exactly one of the instances (by changing either the attributes or labels but not both) so that **no decision tree can attain zero training error** on this dataset (indicate the instance number and the change). [2 pts]

Change instance 6 to $a_1 = F$, $a_2 = T$, $y = T$. Then instances 5 and 6 have the same attributes but different labels.

But since they have the same attributes, a decision tree cannot distinguish them and will classify one of them incorrectly.

Weighted linear regression

14. In the problem set, we considered weighted linear regression where the input features are 1-dimensional. We now extend this to D-dimensional features. Thus, we want to find $\boldsymbol{\theta}$ that minimizes the cost function

$$J(\boldsymbol{\theta}) = \sum_{n=1}^{N} w_n (y_n - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_n)^2$$

Here $w_n > 0$, $\boldsymbol{x}_n \in \mathbb{R}^{D+1}, \boldsymbol{\theta} \in \mathbb{R}^{D+1}$. $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{x}_N^{\mathrm{T}} \end{pmatrix} \in \mathbb{R}^{N \times (D+1)}$, $\boldsymbol{y} \in \mathbb{R}^N$. For this problem, assume that the intercept term is included in the $\boldsymbol{\theta}$ and that the linear regression solution exists in this setting.

Questions:

(a) Show that $J(\boldsymbol{\theta})$ can also be written as:

$$J(\boldsymbol{\theta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) \qquad W = \begin{pmatrix} w_1 & & 0 \\ & w_2 & \\ 0 & & \ddots \; w_n \end{pmatrix}$$

Here $\boldsymbol{W}$ is a diagonal matrix where the entry on the diagonal on row n, column n is w_n. [3 pts]

$$X = \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_N^T \end{pmatrix} \qquad \vec{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \qquad \vec{\theta} = \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_0 \end{pmatrix}$$

$$\vec{y} - X\vec{\theta} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} - \begin{pmatrix} x_{1,0} & \cdots & x_{1,D} \\ \vdots & & \\ x_{N,D} & \cdots & x_{N,D} \end{pmatrix} \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_D \end{pmatrix}$$

$$= \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} - \begin{pmatrix} \theta^T x_1 \\ \vdots \\ \theta^T x_N \end{pmatrix}$$

$$= \begin{pmatrix} y_1 - \theta^T x_1 \\ \vdots \\ y_N - \theta^T x_N \end{pmatrix}$$

$$J(\theta) = \begin{pmatrix} y_1 - \theta^T x_1 & \cdots & y_N - \theta^T x_n \end{pmatrix} \begin{pmatrix} w_1 & & 0 \\ & \ddots & \\ 0 & & w_N \end{pmatrix} \begin{pmatrix} y_1 - \theta^T x_1 \\ \vdots \\ y_N - \theta^T x_n \end{pmatrix}$$

$$= \begin{pmatrix} w_1(y_1 - \theta^T x_1) & \cdots & w_N(y_N - \theta^T x_N) \end{pmatrix} \begin{pmatrix} y_1 - \theta^T x_1 \\ \vdots \\ y_N - \theta^T x_N \end{pmatrix}$$

$$= w_1(y_1 - \theta^T x_1)^2 + \cdots + w_N(y_N - \theta^T x_N)^2$$

$$= \sum_{n=1}^{N} w_n (y_n - \theta^T x_n)^2$$

(b) Show that the optimal value for $\widehat{\boldsymbol{\theta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}$. For reference, here are some useful gradient identities (where $\boldsymbol{x}, \boldsymbol{b}$ are vectors and $\boldsymbol{A}$ is a symmetric matrix).

$$f(\boldsymbol{x}) = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x} \qquad \nabla f(\boldsymbol{x}) = \boldsymbol{b}$$
$$f(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} \qquad \nabla f(\boldsymbol{x}) = 2\boldsymbol{A}\boldsymbol{x}$$

[5 pts]

$$J(\theta) = (y - X\theta)^T W (y - X\theta)$$

$$\nabla J(\theta) = 0$$

$$\nabla J(\theta) = 2W(y - X\theta) = 0$$

$$Wy - WX\theta = 0$$

$$WX\theta = Wy$$

$$X^T WX\theta = X^T Wy$$

$$\hat{\theta} = (X^T WX)^{-1} X^T Wy$$

(c) In class, we provided a probabilistic interpretation of ordinary least squares (OLS). We now try to provide a probabilistic interpretation of weighted linear regression. Consider a model where each of the N samples is independently drawn according to a normal distribution

$$P(y_n|\boldsymbol{x}_n, \boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{\left(y_n - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x}_n\right)^2}{2{\sigma_n}^2}\right)$$

In this model, each y_n is drawn from a normal distribution with mean $\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x}_n$ and variance ${\sigma_n}^2$. The ${\sigma_n}^2$ are **known**. Write the log likelihood of this model as a function of $\boldsymbol{\theta}$. [3 points]

$$\ell(\theta) = \sum_{n=1}^{N} \log\left(\frac{1}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{(y_n-\theta^T x_n)^2}{2\sigma_n^2}\right)\right)$$

$$= \sum_{n=1}^{N} \log\left(\frac{1}{\sqrt{2\pi\sigma_n^2}}\right) + \log\left(\exp\left(-\frac{(y_n-\theta^T x_n)^2}{2\sigma_n^2}\right)\right)$$

$$= \sum_{n=1}^{N} \log\left(\frac{1}{\sqrt{2\pi\sigma_2^2}}\right) - \frac{(y_n-\theta^T x_n)^2}{2\sigma_n^2}$$

(d) Show that finding the maximum likelihood estimate of $\boldsymbol{\theta}$ leads to the same answer as solving a weighted linear regression. How do ${\sigma_n}^2$ relate to w_n? [5 points]

$$\frac{\partial \ell(\theta)}{\partial \theta} = \sum_{n=1}^{N} -\frac{1}{2\sigma_n^2} \cdot 2(y_n - \theta^T x_n)^2 = 0$$

$$\sum_{n=1}^{N} \frac{1}{\sigma_n^2}(y_n - \theta^T x_n)^2 = 0$$

$$\sigma_n^2 = \frac{1}{w_n}$$

(Blank page provided for your work)