CM 146 — Machine Learning: Midterm

Fall 2017

Name: Zhouyang Xue

UID: 104629708

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?

① Supervised learning takes labels as input. Its goal is to label an object into certain categories based on its knowledge

② Unsupervised learning doesn't have labels. its goal is to form clusters of the given data.

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

① The result/data of classification is discrete

② while that of regression is real numbers (continuous)

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

Training data and test data don't intersect. The seperation of training and test data allows a model to learn on a part of the data base, and test on unfamiliar, unvisited data. This makes the model much more reliable in real world, because normally the data to be predicted is totally new towards the model.

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.
   Because only this model has probability interpretation.
   The range of the result is from 0 to 1.

   (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

   Memorization allows a model to attain 100% accuracy on training set, yet it doesn't guarantee accuracy on testing set (there might be problems such as overfitting, etc.). Not useful either.
   Therefore, accuracy on test set indicates a model's performance much better than training set.
   The latter has 75% accuracy on training, while the former one only has 70%.
   The latter model is better.

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

   Adding a copy to the training set might affect the entropy of some/multiple expansions, thus influencing the information gain. This could make D_2 different from D_1.

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

   True False

   Time complexity Stochastic Gradient Descent : $O(D)$
   Batch Gradient Descent : $O(N \cdot D)$

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

   It might take more than 100 iterations for the algorithm to converge. It is possible that after some more iterations, training error lowers down to 0%.

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: $\sum_{i=1}^n \left(y_i - \boldsymbol{\theta}^T \boldsymbol{\phi}(\boldsymbol{x}_i)\right)^2$. This optimization problem is convex.

True   False

$(y_i - \theta^T \phi(x_i))^2$

the outer layer of the composed function is $(y_i - \theta^T f_i)^2$, where $f_i = \phi(x_i)$

which is convex, the whole function 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 can change 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

If all houses are measured in m^2,

the ratio remains the same.

The nearest neighbors to the new house remains the same.

Thus predictions remain the SAME.

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

$f(x) = 4x^2 - 12x + 9$

$f'(x) = 8x - 12$

$f''(x) = 8 > 0$

$f(x)$ is convex, so the local minima is also a global minima

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.
   (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 $\theta \in \mathbb{R}$:

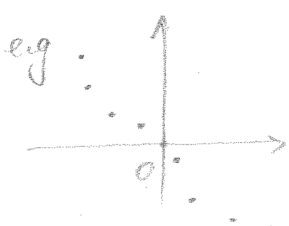
$$\text{A} : y = \theta^2 x$$
$$\text{B} : y = \theta x$$

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.

for A, θ^2 is always non-negative ($\theta^2 \geq 0$)
for B, θ could be any real number
This makes A unable to perfectly fit some data set

e.g.:



$\theta = -1$ for B fits well,
yet no possible value of θ allows A to fit well

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]

$$L(\theta) = P(X_1, X_2, X_3, X_4; \theta)$$
$$= \prod_{i=1}^{4} P(X_i; \theta)$$
$$\ell(\theta) = \log(L(\theta)) = \log\left(\prod_{i=1}^{4} P(X_i; \theta)\right)$$
$$= \sum_{i=1}^{4} \log(P(X_i; \theta))$$
$$\cancel{= \log(1 \cdot \theta) + \log(1 \cdot \theta) + \log(0 \cdot \theta) + \log(1 \cdot \theta)}$$
$$= 3\log\theta + \log(1-\theta)$$

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

$$\cancel{\ell'(\theta) = \frac{3}{\theta}}$$
$$\ell'(\theta) = \frac{d}{d\theta}\left(3\log\theta + \log(1-\theta)\right)$$
$$= \frac{3}{\theta} - \frac{1}{1-\theta}$$
$$= \frac{3}{\theta} + \frac{1}{\theta-1}$$

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

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

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

$$3\hat{\theta} - 3 = -\hat{\theta}$$

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

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

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]

Y: TTTFFF

$H[Y] = \sum_{i=1}^{2} -p(Y=T)\log_2 p(Y=T))$

$= -\frac{1}{2}\cdot(-1) - \frac{1}{2}\cdot(-1)$

$= 1$

$p = \frac{1}{2} \cdot H[Y] = 1$

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

expand on a_1 : TTF $F: \frac{1}{3}$ pt $\{\frac{1}{3}, \frac{2}{3}\}$: $H[a_1] = 0.92$
FTT $F: \frac{1}{3}$

information gain $G = 1 - 0.92 = 0.08$

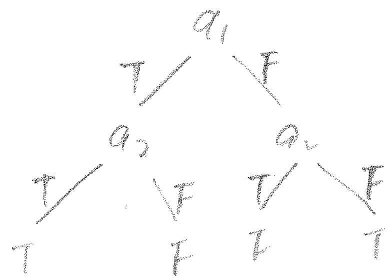
expand on a_2 : TTFF $F: \frac{1}{2}$ pt $\{\frac{1}{2}\}$: $H[a_2] = 1$
TF $F: \frac{1}{2}$

information gain $G = 1 - 1 = 0$

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

it'll choose a_1

(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 the result (T) of the first instance (1) to F

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:

$$J(\boldsymbol{\theta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})$$

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

$$X\theta = \begin{pmatrix} X_1^T \\ X_2^T \\ \vdots \\ X_N^T \end{pmatrix} \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_D \end{pmatrix} = \begin{bmatrix} X_1^T\theta \\ X_2^T\theta \\ \vdots \\ X_N^T\theta \end{bmatrix}$$

$$\therefore\ y - X\theta = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} - \begin{pmatrix} X_1^T\theta \\ X_2^T\theta \\ \vdots \\ X_N^T\theta \end{pmatrix} = \begin{pmatrix} y_1 - X_1^T\theta \\ y_2 - X_2^T\theta \\ \vdots \\ y_N - X_N^T\theta \end{pmatrix}$$

$$W(y - X\theta) = \begin{bmatrix} W_1 & 0 & 0 & \cdots \\ 0 & W_2 & & \\ 0 & 0 & W_3 & \\ 0 & 0 & & \\ \vdots & \vdots & \vdots & \end{bmatrix} \cdot \begin{bmatrix} y_1 - X_1^T\theta \\ y_2 - X_2^T\theta \\ \vdots \\ y_N - X_N^T\theta \end{bmatrix} = \begin{bmatrix} W_1(y_1 - X_1^T\theta) \\ W_2(y_2 - X_2^T\theta) \\ \vdots \\ W_N(y_N - X_N^T\theta) \end{bmatrix}$$

$$(y - X\theta)^T W (y - X\theta) = [y_1 - X_1^T\theta,\ y_2 - X_2^T\theta,\ \dots,\ y_N - X_N^T\theta] \cdot \begin{bmatrix} W_1(y_1 - X_1^T\theta) \\ W_2(y_2 - X_2^T\theta) \\ \vdots \\ W_N(y_N - X_N^T\theta) \end{bmatrix}$$

$$= w_1(y_1 - X_1^T\theta)^2 + w_2(y_2 - X_2^T\theta)^2 + \dots + W_N(y_N - X_N^T\theta)^2$$

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

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

same as $J(\theta)$

(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]

$$\nabla J(\theta) = \frac{d\,(y - X\theta)^T w\,(y - X\theta)}{d\theta}$$

$X^T \cdot 2w(y - X\theta)$
$2X^T w(y - X\theta)$
$2X^T wy - 2X^T wX\theta$

let $f(\theta) = y - X\theta \quad f'(\theta) = -X^T$

$$\nabla J(\theta) = \frac{d\, f(\theta)^T w f(\theta)}{d\theta}$$

$$= f'(\theta) \cdot 2 \cdot w f(\theta)$$

$$= -X^T \cdot 2 \cdot w \cdot (y - X\theta)$$

$$= -2X^T wy + 2X^T wX\theta$$

$$-2X^T wy + 2X^T wX\hat{\theta} = 0$$

$$2X^T wX\hat{\theta} = 2X^T wy$$

$$\hat{\theta} = \frac{X^T wy}{X^T wX}$$

$$= (X^T wX)^{-1} \cdot X^T wy$$

(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

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

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]

$$\ell(\theta) = \sum_{n=1}^{N} \log(P(y_n|x_n,\theta))$$

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

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

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

(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]

$$\ell'(\theta) = -\frac{d}{d\theta}\sum_{n=1}^{N} \frac{(y_n - x_n^T\theta)^2}{2\sigma_n^2}$$

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

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

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

$$\sum_{n=1}^{N} x_n^T \cdot \frac{1}{\sigma_n^2} \cdot x_n \hat{\theta} = \sum_{n=1}^{N} x_n^T \cdot \frac{1}{\sigma_n^2} y_n$$

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

$$= (X^T \cdot W X)^{-1} \cdot X^T W y$$

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

(Blank page provided for your work)