

CS M146 Final Exam

Michael Donald Wu

TOTAL POINTS

98.5 / 100

QUESTION 1

1 True or False 14 / 14

- ✓ - 0 pts all correct
- 2 pts a. incorrect
- 2 pts b. incorrect
- 2 pts c. incorrect
- 2 pts d. incorrect
- 2 pts e. incorrect
- 2 pts f. incorrect
- 2 pts g. incorrect

QUESTION 2

Hidden Markov Models 9 pts

2.1 a 3 / 3

- ✓ - 0 pts Correct
- 3 pts incorrect

2.2 b 3 / 3

- ✓ - 0 pts Correct
- 1 pts partially incorrect computation
- 2 pts incorrect computation
- 3 pts incorrect

2.3 c 3 / 3

- ✓ - 0 pts Correct
- 2 pts incorrect justification
- 3 pts incorrect

QUESTION 3

3 Naive Bayes 12 / 12

- ✓ - 0 pts Correct
- 1.5 pts a) incomplete
- 3 pts a) incorrect
- 1 pts b) minor mistake
- 4 pts b) incorrect
- 2 pts c) incorrect
- 1 pts c) minor mistake

- 3 pts d) incorrect

QUESTION 4

Kernels and SVM 25 pts

4.1 a.i 3 / 3

- ✓ - 0 pts Correct
- 1 pts minor error
- 2 pts unclear prove, but partially correct
- 3 pts Incorrect

4.2 a.ii 5 / 5

- ✓ - 0 pts Correct
- 1 pts minor error
- 2 pts Partially correct
- 3 pts You haven't reach the key point yet, but you are on the way
- 4 pts Wrong way!! 1 point for proving $v^T A v \geq 0$ with a special v (or B)
- 5 pts Incorrect

4.3 b.i 4 / 4

- ✓ - 0 pts Correct
- 2 pts first blank incorrect
- 2 pts second blank incorrect

4.4 b.ii 4 / 4

- ✓ - 0 pts Correct
- 2 pts minor incorrect
- 4 pts incorrect. (A typical wrong statement is saying that it turns out to be hard SVM.)

4.5 b.iii 3 / 3

- ✓ - 0 pts False statement, with reasonable explanation
- 2 pts False statement, without/ with wrong explanation
- 3 pts True statement

4.6 b.iv 2 / 3

- 0 pts True statement, mentioned dual form,

support vectors or similar

- **1 pts** True statment, mentioned a perceptron-style update, but fail to discuss the difference with SVM (i.e. stating that alpha is # of mistakes)

✓ - **2 pts** True statement without/ with wrong explanation

- **3 pts** False statement

+ **1 Point** adjustment

☞ I can update my model using multiple training data with non-linear combination

4.7 b.v 2.5 / 3

- **1 pts** W is wrong: either w_1/w_2 does not equal to +1 or w_1, w_2 are negative

- **0.5 pts** b is wrong: either b is positive; or b does not suitable for W

✓ - **1.5 pts** sv's are wrong (0.5 for each)

- **0 pts** Correct

+ **1 Point** adjustment

☞

QUESTION 5

Short Answer Questions 38 pts

5.1 Adaboost 3 / 3

✓ - **0 pts** Correct

- **2 pts** Correct point, Incorrect justification

- **3 pts** Incorrect answer

- **1 pts** Wrong decision stump

- **2 pts** Circled positive points on one side of the decision stump but reasoning is correct

5.2 Clustering 4 / 4

✓ - **0 pts** Correct

- **2 pts** Incorrect explanation

- **4 pts** Incorrect

- **1 pts** Insufficient explanation

5.3 LOOCV 3 / 3

✓ - **0 pts** Correct

- **3 pts** Incorrect

5.4 Probability 4 / 4

✓ - **0 pts** Correct

- **1 pts** Wrong denominator

- **1 pts** Wrong numerator

- **4 pts** Incorrect

5.5 Multiclass 6 / 6

✓ - **0 pts** Correct

- **2 pts** Minor mistake / Didn't sum over all examples / Didn't sum over all classes

- **4 pts** Only procedure / Attempt to derive (taken log somewhere in the derivation)

- **6 pts** Incorrect

- **3 pts** Mostly correct formulation

- **1 pts** Tiny mistake

5.6 PAC_i 3 / 3

✓ - **0 pts** Correct (200 examples)

- **3 pts** Incorrect

- **2 pts** Correct approach but no answer

- **1 pts** minor mistake

5.7 PAC_ii 3 / 3

✓ - **0 pts** Correct (PAC theorem only shows the upper bound)

- **1 pts** Incorrect explanation

- **3 pts** Incorrect

5.8 Generative vs Discriminative 4 / 4

✓ - **0 pts** Both correct

- **2 pts** One incorrect answer

- **4 pts** Both incorrect

5.9 VC Dimension 8 / 8

✓ - **0 pts** Correct

- **8 pts** Incorrect

- **4 pts** $VC(DT_3)=8$ w/ explanation or examples

- **2 pts** incorrect Prove $VC(DT_3) \geq 8$

- **2 pts** Prove $VC(DT_k) < 9$ ($=2^3 + 1$)

- **2 pts** minor mistake

QUESTION 6

6 Name and Id 2 / 2

✓ - **0 pts** Correct

CM146: Introduction to Machine Learning

Winter 2018

Final Exam

Mar. 22nd, 2018

- This is a closed book exam. Everything you need in order to solve the problems is supplied in the body of this exam.
- This exam booklet contains five problems.
- You have 150 minutes to earn a total of 100 points.
- Besides giving the correct answer, being concise and clear is very important. To get the full credit, you must show your work and explain your answers.

Good Luck!

Name and ID: (2 Point)

Michael Wu 401752542

Name		/2
True/False Questions		/14
Hidden Markov Models		/9
Naive Bayes		/12
Kernels and SVM		/25
Short Answer Questions		/38
Total		/100

1 True or False [14 pts]

Choose either True or False for each of the following statements. For the statement you believe it is False, please give your brief explanation of it. Two points for each question. Note: the credit can only be granted if your explanation for the false statement is correct. Also note, a negated statement is not counted as a correct explanation.

(a) Training a k -class classification model using one-against-all is always faster than using one-vs-one because one-vs-one requires to train more binary classifiers.

False, one-vs-one requires less examples per classifier, so it may be faster if training time is large as in large examples.

(b) We would expect the support vectors to remain the same in general as we move from a linear kernel to higher order polynomial kernels.

True, their generation should be the same but their appearance as a decision boundary should look different.

(c) In a mistake-driven algorithm such as the Perceptron algorithm, if we make a mistake on example x_i with label y_i , we update the weights w so that we can guarantee that we now predict y_i correctly.

False, y_i may still be incorrect, we only update w so that it is correct to predicting y_i correct.

(d) Consider a classification problem with n features. The VC dimension of the corresponding (linear) SVM hypothesis space is larger than that of the corresponding logistic regression hypothesis space.

False, both algorithms separate along a hyperplane, leading to the same type of decision boundary.

(e) A 3-layer neural network with non-linear activation functions can learn non-linear decision boundaries.

True

(f) In AdaBoost, the weight associated with each weak learner can be negative (less than 0).

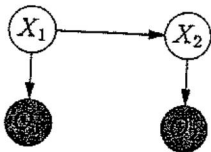
False, our weak learner should get less than 50% weight even, and so our weights will be a

(g) Using MAP to estimate model parameters always give us better performance.

False, MLE may sometimes perform bet

2 Hidden Markov Models [9 pts]

Consider the following Hidden Markov Model.



X_1	$\Pr(X_1)$
0	0.3
1	0.7

X_t	X_{t+1}	$\Pr(X_{t+1} X_t)$
0	0	0.4
0	1	0.6
1	0	0.8
1	1	0.2

X_t	O_t	$\Pr(O_t X_t)$
0	A	0.9
0	B	0.1
1	A	0.5
1	B	0.5

Suppose that $O_1 = A$ and $O_2 = B$ is observed.

- (a) (3 pts) What is the probability of $P(O_1 = A, O_2 = B, X_1 = 0, X_2 = 1)$?

$$P(X_1=0) P(X_2=1 | X_1=0) P(O_1=A | X_1=0) P(O_2=B | X_2=1)$$

$$0.3 \quad 0.6 \quad 0.9 \quad 0.5$$

$$= \boxed{0.081}$$

- (b) (3 pts) What is the most likely assignment for X_1 and X_2 ?

X_1	X_2	O_1	O_2
-------	-------	-------	-------

$$P(0 \ 0 \ A \ B) = 0.3 * 0.4 * 0.9 * 0.1 = 0.0108$$

$$P(0 \ 1 \ A \ B) = 0.3 * 0.6 * 0.9 * 0.5 = 0.081$$

$$P(1 \ 0 \ A \ B) = 0.7 * 0.8 * 0.5 * 0.1 = 0.028$$

$$P(1 \ 1 \ A \ B) = 0.7 * 0.2 * 0.5 * 0.5 = 0.035$$

Most likely is $X_1=0, X_2=1$

- (c) (3 pts) [True/False] Based on the independent assumptions in HMM, the random variable O_1 is independent of the random variable X_2 . Justify your answer.

$$P(X_2=1) = 0.3 * 0.6 + 0.7 * 0.2 = 0.32$$

$$P(O_1=A) = 0.3 * 0.9 + 0.7 * 0.5 = 0.62$$

$$P(X_2=1, O_1=A) = 0.3(0.6 * 0.9) + 0.7(0.2 * 0.5) = 0.162 + 0.073 = 0.235$$

$$P(X_2=1 | O_1=A) = \frac{0.235}{0.62} = \frac{58}{155} \neq P(X_2=1)$$

False

3 Naive Bayes [12 pts]

Data the android is about to play in a concert on the Enterprise and he wants to use a Naive Bayes classifier to predict whether he will impress Captain Picard. He believes that the outcome depends on whether Picard has been reading Shakespeare or not for the three days before the concert. For the previous five concerts, Data has observed Picard and noted on which days he read Shakespeare. His observations look like this:

D1 (Day 1)	D2 (Day 2)	D3 (Day 3)	LC (Liked Concert)
1	1	0	yes
0	0	1	no
1	1	1	yes
1	0	1	no
0	0	0	no

(a) (3 pts) What does the modeling assumption make in the Naive Bayes model?

Reading Shakespeare on a given day is mutually independent. $P(D_1, D_2, D_3 | LC) = \prod_{i=1}^3 P(D_i | LC)$

(b) (4 pts) Show the Naive Bayes model that Data obtains using maximum likelihood from these instances. (Write down the numerical values of the model parameters.)

$$\begin{aligned}
 P(D_1=1 | LC=yes) &= \frac{1}{3} \\
 P(D_1=0 | LC=yes) &= \frac{2}{3} \\
 P(D_1=1 | LC=no) &= \frac{2}{3} \\
 P(D_1=0 | LC=no) &= \frac{1}{3}
 \end{aligned}$$

$$\begin{aligned}
 P(D_2=1 | LC=yes) &= \frac{1}{2} \\
 P(D_2=0 | LC=yes) &= \frac{1}{2} \\
 P(D_2=1 | LC=no) &= \frac{2}{3} \\
 P(D_2=0 | LC=no) &= \frac{1}{3}
 \end{aligned}$$

$$P(LC=yes) = \frac{2}{5}$$

(c) (2 pts) If Picard reads Shakespeare only on day 1 and day 2, how likely is he to enjoy Data's concert?

$$P(D_1=1 | LC=yes) = \frac{1}{3}$$

$$P(D_2=1 | LC=yes) = \frac{1}{2}$$

$$\frac{1}{3} + 0 = P(D_1=1, D_2=1)$$

(d) (3 pts) Estimate $P(LC = yes | D_2 = 1)$.

$$\begin{aligned}
 P(LC=yes | D_2=1) &= \frac{P(LC=yes) P(D_2=1 | LC=yes)}{P(D_2=1)} \\
 &= \frac{\frac{2}{5} \cdot \frac{1}{2}}{\frac{2}{3}} = \frac{2}{5} = 40\%
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{P(LC=yes) P(D_2=1 | LC=yes)}{P(D_2=1)} = \frac{\frac{2}{5} \cdot \frac{1}{2}}{\frac{2}{3}} = \frac{2}{5} = 40\%
 \end{aligned}$$

$$\begin{aligned}
 P(D_2=1) &= \frac{2}{5} + 0 \\
 &= \frac{2}{5} + 0
 \end{aligned}$$

4 Kernels and SVM [25 pts]

(a) (8 pts) Properties of Kernels

i. (3 pts) Given n training examples $\{x_i\}_{i=1}^n$, the kernel matrix A is an $n \times n$ square matrix, where $A(i, j) = K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j)$. Prove that the kernel matrix is symmetric (i.e., $A_{i,j} = A_{j,i}$).

hints: Your proof will not be longer than 2 or 3 lines.

$$A(i, j) = \Phi(x_i)^T \Phi(x_j) = \Phi(x_j)^T \Phi(x_i) = A(j, i)$$

↓
since this is a dot product and commutes ↓

ii. (5 pts) Prove that the kernel matrix A is positive semi-definite.

hints: (1) Remember that an $n \times n$ matrix A is positive semi-definite if and only if for any n dimensional vector $v \neq 0$, we have $v^T A v \geq 0$. (2) Consider a matrix $B = [\Phi(x_1), \dots, \Phi(x_n)]$ and use it to prove A is positive semi-definite.

$$A = B B^T$$

$$v^T A v = v^T B B^T v = (B^T v)^T B^T v = x^T x$$

$$\text{let } x = B^T v$$

$x^T x$ is the dot product of some vector with itself, which must always be ≥ 0 , thus $v^T A v \geq 0 \forall v$.
 A is semi-definite

(b) (17 pts) Given a dataset $D = \{x_i, y_i\}, x_i \in \mathbb{R}^k, y_i \in \{-1, +1\}, 1 \leq i \leq N$.

A hard SVM solves the following formulation

$$\min_{w, b} \frac{1}{2} w^T w \quad \text{s.t.} \quad \forall i, y_i (w^T x_i + b) \geq 1, \quad (1)$$

and soft SVM solves

$$\min_{w, \xi_i, b} \frac{1}{2} w^T w + C \sum_i \xi_i \quad \text{s.t.} \quad \forall i, y_i (w^T x_i + b) \geq 1 - \xi_i, \quad \forall i, \xi_i \geq 0 \quad (2)$$

i. (4 pts) Complete:

If $C =$ very large, soft SVM will behave exactly as hard SVM.

In order to reduce over-fitting, one should decrease (decrease or increase) the value of C .

ii. (4 pts) Show that when $C = 0$, the soft SVM returns a trivial solution and cannot be a good classification model.

if $C=0$, let $w=0, b=0, \forall i: \xi_i = \gamma$
 then $\frac{1}{2}w^T w + C \sum \xi_i = 0$ and is minimum
 and $\forall i: y_i(w^T x_i + b) \geq 1 - \xi_i$ because $1 - \xi_i = 0$
 $y_i(w^T x_i + b) = 0$

But everything is zero, tells us nothing

iii. (3 pts) [True/False] The slack variable ξ_i in soft SVM for a data point x_i always takes the value 0 if the data point is correctly classified by the hyper-plane. Explain your answer.

False. $y_i(w^T x_i + b) = 0.5$ is on the right side of the hyperplane, but ξ_i can be 0.5

iv. (3 pts) [True/False] The optimal weight vector w can be calculated as a linear combination of the training data points. Explain your answer. [You do not to prove this.]

True, our algorithm works by using multiples of the training data, so it's a linear combination

v. (3 pts) We are given the dataset in Figure 1 below, where the positive examples are represented as black circles and negative points as white squares. (The same data is also provided in Table 1 for your convenience). Recall that the equation of the separating hyperplane is $\hat{y} = w^T x + b$.

i. Write down the parameters for the learned linear decision function.

$W = (w_1, w_2) = (1, 1), b = -1$

ii. Circle all support vectors in Figure 1.

index	x_1	x_2	y
1	0	0	-
2	0	-4	-
3	-1	-1	-
4	-2	-2	-
5	3	0	+
6	0	3	+
7	1	1	+
8	3	-1	+

Table 1: The dataset S

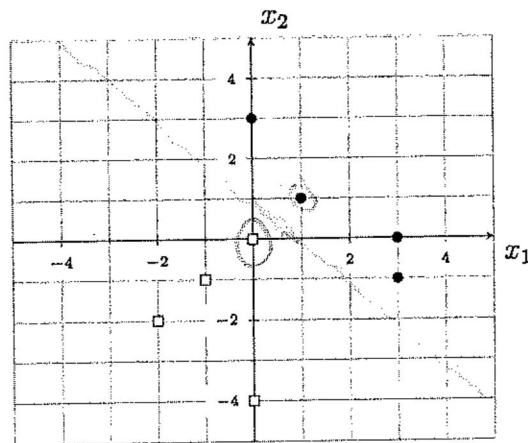


Figure 1: Linear SVM

Decision boundary

5 Short Answer Questions [38 pts]

Most of the following questions can be answered in one or two sentences. Please make your answer concise and to the point.

- (a) (3 pts) Consider training a classifier using AdaBoost with decision stumps (pick a horizontal or a vertical line, and one side of the half-space is positive and the other one is negative) on the following dataset:

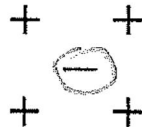


Figure 2: Example 2D dataset for Boosting

Which example(s) will have their weights increased at the end of the first iteration? Circle them and justify.

A classifier will classify all as positive, as this results in least error

- (b) (4 pts) Suppose we clustered a set of N data points using two different clustering algorithms: k-means and Gaussian mixtures. In both cases we obtained 2 clusters and both algorithms return the same set of cluster centers. Can 2 points that are assigned to different clusters in the kmeans solution be assigned to the same cluster in the Gaussian mixture solution? If no, explain. If so, sketch an example and explain in 1-2 sentences.

yes

if the two Gaussian distributions overlap some points may be interspersed. This does not happen in k-means as it uses closest distance.

- (c) (3 pts) Suppose you are running a learning experiment on a new algorithm for binary classification. You have a data set consisting of 100 positive and 100 negative examples. You plan to use leave-one-out cross-validation (i.e., 200-fold cross-validation) to evaluate a baseline method: a simple majority function (i.e., returns the most frequent label on the training set as the prediction). What is the average cross-validation accuracy of the baseline? (Only need to write down the number).

train
200 +
100 -

test	train	test
-	99+	pred
+	100-	+

0% accuracy

- (d) (4 pts) $P(\text{Good Movie} \mid \text{Includes Tom Cruise}) = 0.01$
 $P(\text{Good Movie} \mid \text{Tom Cruise absent}) = 0.1$
 $P(\text{Tom Cruise in a randomly chosen movie}) = 0.01$

What is $P(\text{Tom Cruise is in the movie} \mid \text{Not a Good Movie})$?

$$P(!\text{Good}) = P(!\text{good} \mid \text{TC}) P(\text{TC}) + P(!\text{good} \mid !\text{TC}) P(!\text{TC})$$

$$= 0.99 * 0.01 + 0.9 * 0.99$$

$$P(\text{TC}, !\text{good}) = P(!\text{good} \mid \text{TC}) P(\text{TC}) = 0.99 * 0.01$$

$$P(\text{Tom Cruise in movie} \mid \text{not good movie}) = \frac{0.01}{0.91}$$

- (e) (6 pts) We can easily extend the binary Logistic Regression model to handle multi-class classification. Lets assume we have K different classes, and posterior probability for class k is given as

$$P(y = k \mid X = x) = \frac{\exp(w_k^T x)}{\sum_{k'=1}^{K-1} \exp(w_{k'}^T x)} \quad (3)$$

where x is a d dimensional vector and w_k is the weight matrix for the k^{th} class.

Assuming dataset D consists of n examples, derive the log likelihood condition for this classifier.

hints: Let I_{ik} be an indicator function, where $i = 1, \dots, n$ and $I_{ik} = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{if } y_i \neq k \end{cases}$

(Full points if the derivation is mathematically correct. 2 points if you can describe the procedure for deriving.)

$$\begin{aligned} \log \text{Likelihood} &= \log \prod_{i=1}^n P(y=y_i \mid x=x_i) \\ &= \sum_{i=1}^n \log P(y=y_i \mid x=x_i) \\ &= \sum_{i=1}^n \sum_{j=1}^K I_{ij} \log P(y=j \mid x=x_i) \\ &= \sum_{i=1}^n \sum_{j=1}^K I_{ij} \left(w_j^T x_i - \log \left(\sum_{k=1}^K e^{w_k^T x_i} \right) \right) \end{aligned}$$

- (f) (6 pts) In class we learned the following PAC learning bound for consistent learners:
Theorem 1. Let H be a finite concept class. Let D be an arbitrary, fixed unknown distribution over X . For any $\epsilon, \delta > 0$, if we draw a sample S from D of size

$$m > \frac{1}{\epsilon} \left(\ln(|H|) + \ln \frac{1}{\delta} \right) \quad (4)$$

then with probability at least $1 - \delta$, all hypothesis $h \in H$ have $err_D(h) \leq \epsilon$. Our friend Kai is trying to solve a learning problem that fits in the assumptions above.

- i. Kai tried a training set of 100 examples and observed some test error, so he wanted to reduce the test error to half. How many examples should Kai use, according to the above PAC bound?

$$\epsilon = \frac{\epsilon}{2}$$

then he should use at least 200

examples

- ii. Kai took your suggestion and ran his algorithm again, however the error on the test set did not halve. Do you think it is possible? explain briefly.

it is possible. This occurs because $err_{D(h)}$ may have been less than ϵ in the first training, so even if the bound ϵ decreased, $err_D(h)$ may not have decreased.

- (g) (4 pts) List two differences between generative and discriminative learning models.

Generative
finds $P(x, y)$

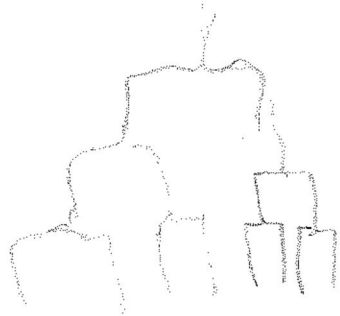
discriminative
finds $P(y|x)$

The generative model cares about how often a data x occurs, whereas a discriminative model only cares about how likely a data x has label y , i.e. $P(y|x)$ is discriminative while $P(x, y)$ is generative

- (h) (8 pts) We define a set of functions $T = f(x) = I[x > a] : a \in \mathbb{R}^1$, where $I[x > a]$ is the indicator function returning 1 if $x > a$ and returning 0 otherwise. For input domain $X = \mathbb{R}^1$, and a fixed positive number k , consider a concept class DT_k consisting of all decision trees of depth at most k where the function at each non-leaf node is an element of T . Note that if the tree has only one decision node (the root) and two leaves, then $k = 1$.

Determine the VC dimension of DT_3 , and prove that your answer is correct.

VC dimension of DT_3 is 8



Consider 8 points $0, 1, 2, 3, 4, 5, 6, 7$

A tree in DT_3 as shown can shatter these points, as there are 8 leaf nodes that correspond to each of the points. So any labeling can be completely identified.

However, for any 9 points DT_3 cannot shatter them, because by the pigeonhole principle will cause two points to be in the same leaf node. If these two points are not labelled the same, they will not be classified correctly, thus VC dimension is 8.