CS 188 — Machine Learning: Midterm

Winter 2017

Name: Sam Rubenacker

UID: 504295581

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	4	
2	Application	6	
3	True/False	12	
4	Multiple choice	7	
5	Maximum likelihood	5	
6	Decision Trees	10	
7	Regression	6	
Total		50	

1. (4 pts) **Machine Learning Basics**

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

   Supervised learning takes instances and labels as inputs, and tries to predict a label for an unseen instance.

   Unsupervised learning takes instances which are not labeled, and tries to come up with the underlying structure of the input data (clustering).

   (b) (1 pts) What is the main difference between classification and regression?

   Classification deals with discrete inputs and outputs, while regression deals with real valued inputs and outputs.

   (c) (1 pts) Learning is about generalizing from training data to (unseen) test data. What does this assume about the training and test set?

   This assumes the training and test data come from the same probability distribution.

2. (6 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?

I would choose logistic regression because it outputs a probability along with a classification.

(b) (2 pts) (True/False) Suppose this dataset had 900 cancer-free images and 100 images from cancer patients. If you train a model which achieves 85% accuracy, it is a good model (Hint: think about a baseline).

No. You could use a majority vote classifier which would achieve 90% accuracy by always predicting cancer-free, (since 90% of data is cancer free).

(c) (2 pts) (True/False) Suppose that you split your dataset into a training set and test set. A model that attains 100% accuracy on the training set and 70% accuracy on the test set is better than a model that attains 70% accuracy on the training set and 75% accuracy on the test set.

False. A model which achieves 100% accuracy on training data could simply memorize it. Therefore the model which better generalizes is preferred, so the model with 75% test accuracy is better.

True/False

3. (2 pts) (True/False) 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, 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.

False. Adding that copy of the instance to the training set may affect the rankings of information gain among the features, leading to different features being chosen to split on, creating a different tree.

4. (2 pts) (True/False) Stochastic Gradient Descent is faster per iteration than Gradient Descent.

False, it still takes the same amount of time to compute the gradient, but you may end up doing it fewer times with stochastic than with batch.

5. (2 pts) (True/False) 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.

False. If the training data is linearly separable, then the perceptron algorithm will eventually converge. You may need more iterations for it to eventually converge, especially if you do not permute the training instances in an acceptable manner.

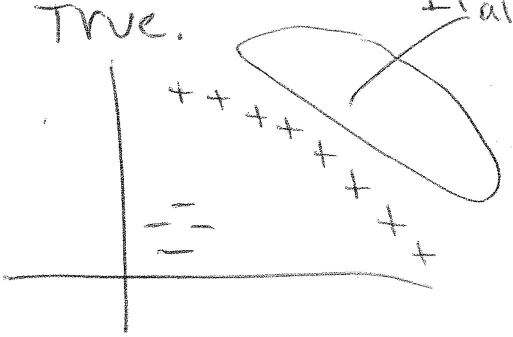
6. (2 pts) (True/False) You learn a decision tree with the *MaxDepth* parameter set to

infinity and then prune the resulting decision tree. Pruning the decision tree tends to reduce overfitting.

True. Decision trees with max depth will memorize the training set and perform poorly on test data, and thus overfit. Pruning nodes will make the decision tree perform worse on training data and may be better on test.

7. (2 pts) (True/False) We want to use 1-NN to classify data into one of two classes. It is possible for 1-NN to always classify all new instances as positive even though there are negative instances in the training data. (If true, show an instance. If false, briefly explain.)

True.



If all new instances appear here, ~~and~~ they will always be classified as positive, even though there are negative instances in the training data.

8. (2 pts) (True/False) 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.

$$f(x) = (2x-3)^2 = (2x-3)(2x-3) = 4x^2 - 12x + 9$$
$$f'(x) = 8x - 12$$
$$f''(x) = 8$$

True, $f(x)$ is convex because its second derivative is > 0, and thus there is only 1 minima, and it is global.

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 $\{x, y\}$ pairs). Further assume that we have an unlimited amount of data. We would like to compare the following two models on our input dataset:

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

For each one, 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 perform *better* than B.
   (b) There are datasets for which B would perform *better* than A.
   $\boxed{C}$ [✓ (c) Both (i) and (ii) are correct.]
   (d) They would perform equally well on all datasets.

There may be datasets where you need to predict a negative y ~~answer with A you can't~~. for a positive x, and with A, you would not be able to since Θ^2 is always positive. Also, there may be times where Θ^2 is a better weight than just Θ, so it depends on the data set.

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! Blank pages have been provided for this purpose, or you may attach extra pages as needed. (If you use additional pages, please indicate clearly the problem being solved and write your name and UID on each page.)

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.*, $P(X_n = 1) = \theta$): $(1, 1, 0, 1)$.

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

$$L(\theta) = \prod_{n=1}^{N} \theta^{x_n}(1-\theta)^{1-x_n}$$

$$l(\theta) = \sum_{n=1}^{N} \log \theta^{x_n} + \log(1-\theta)^{1-x_n}$$

$$l(\theta) = \sum_{n=1}^{N} x_n \log\theta + (1-x_n)\log(1-\theta)$$

$$l(\theta) = 1\log\theta + 1\log\theta + 1\log(1-\theta) + 1\log\theta = 3\log\theta + \log(1-\theta)$$

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

$$l'(\theta) = \frac{3}{\theta} - \frac{1}{1-\theta}$$

(c) What is the maximum likelihood estimate of θ? [1 pts]

$$l'(\theta) = 0 = \frac{3}{\theta} - \frac{1}{1-\theta}$$

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

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

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

$$4\theta = 3$$

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

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 variables.

Instance number	a_1	a_2	Target
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):

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

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

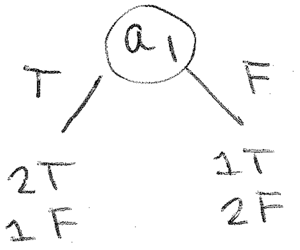
$$H[\text{target}] = -\sum_{n=1}^{N} P(X=a_n) \log(P(X=a_n))$$

$$= -\left(\frac{1}{2} \log\left(\frac{1}{2}\right) + \frac{1}{2} \log\left(\frac{1}{2}\right)\right)$$

$$= -\left(\log_2 2^{-1} \right)$$

$$= \underline{1}$$

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

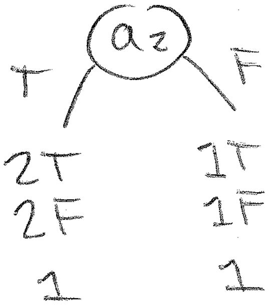


0.92 0.92

$$\frac{3}{6} \cdot 0.92 + \frac{3}{6} \cdot 0.92 = 0.92$$

information gain $= 1 - 0.92$

$= 0.08$



$$\frac{4}{6} \cdot 1 + \frac{2}{6} \cdot 1 = 1$$

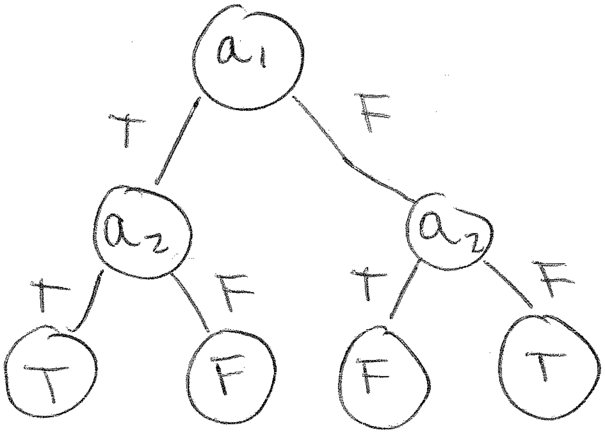
information gain $= 1 - 1$

$= 0$

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

It will pick a_1, since it has a 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 (either the attributes or labels) 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 so that $a_1 = T$.

Linear Regression

14. We are given a set of N two-dimensional inputs and their corresponding output: $\{\boldsymbol{x}_n, y_n\} \mid \boldsymbol{x}_n = \begin{pmatrix} x_{n,1} \\ x_{n,2} \end{pmatrix} \in \mathbb{R}^2 \mid y_n \in \mathbb{R}, n \in \{1, \ldots, N\}$. Given $\boldsymbol{x}_n$, we would like to use the following regression model to predict y_n:

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}_n) = \theta_1^2\, x_{n,1} + \theta_2^2\, x_{n,2}.$$

We learn this model by finding values of the parameters (θ_1 and θ_2) that minimize the cost function defined as the sum of squared errors between the predicted and true labels (also called the residual sum of squares).

(a) Write out the cost function that is minimized (your answer should be expressed in terms of y_n, $x_{n,1}$, $x_{n,2}$, θ_1 and θ_2). [1 pts]

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

$$J(\theta) = \sum_{n=1}^{N} (y_n - (\theta_1^2 X_{n,1} + \theta_2^2 X_{n,2}))^2$$

(b) Derive the optimal value(s) for θ_1. (You should find a closed-form solution. Note that θ_2 may appear in your resulting equation and that there may be more than one possible value for θ_1.) [5 pts]

$$\frac{\partial J(\theta)}{\partial \theta_1} = \sum_{n=1}^{N} 4X_{n,1}\theta_1 (y_n - (\theta_1^2 X_{n,1} + \theta_2^2 X_{n,2}))$$

$$0 = 4 \sum X_{n,1}\theta_1 y_n - X_{n,1}\theta_1(\theta_1^2 X_{n,1} + \theta_2^2 X_{n,2})$$

$$0 = \sum X_{n,1}\theta_1 y_n - \sum(\theta_1^3 X_{n,1}^2 + X_{n,1}\theta_1\theta_2^2 X_{n,2})$$

next page

$$\sum \Theta_1^3 x_{n,1}^2 + \sum \Theta_1 x_{n,1} \Theta_2^2 x_{n,2} = \sum x_{n,1} \Theta_1 y_n$$

$$\Theta_1^3 \sum x_{n,1}^2 + \Theta_1 \Theta_2^2 \sum x_{n,1} x_{n,2} = \Theta_1 \sum x_{n,1} y_n$$

$$\Theta_1^3 \overline{x_1^2} + \Theta_1 \Theta_2^2 \overline{x_{n,1} x_{n,2}} = \Theta_1 \overline{x_1 y}$$

~~$\Theta_1^3 \overline{x_1^2} + \Theta_1 \Theta_2^2 \overline{x_1 x_2} - \Theta_1 \overline{x_1 y} = 0$~~

~~$\Theta_1 (\Theta_1^2 \overline{x_1^2} + \Theta_2^2 \overline{x_1 x_2} - x_1$~~

$$\Theta_1^3 \overline{x_1^2} = \Theta_1 \overline{x_1 y} - \Theta_1 \Theta_2^2 \overline{x_1 x_2}$$

$$\Theta_1^3 \overline{x_1^2} = \Theta_1 (\overline{x_1 y} - \Theta_2^2 \overline{x_1 x_2})$$

$$\Theta_1^2 \overline{x_1^2} = \overline{x_1 y} - \Theta_2^2 \overline{x_1 x_2}$$

$$\Theta_1 = \sqrt{\frac{\overline{x_1 y} - \Theta_2^2 \overline{x_1 x_2}}{\overline{x_1^2}}}$$

(Blank page provided for your work)