CIS 4190/5190: Applied Machine Learning	Spring 2023

Homework 3

Handed Out: February 8 *Due: February 22, 8:00 p.m.*

- You are encouraged to format your solutions using LaTeX. You'll find some pointers to resources for learning LaTeX among the Canvas primers. Handwritten solutions are permitted, but remember that you bear the risk that we may not be able to read your work and grade it properly — do not count on providing post hoc explanations for illegible work. You will submit your solution manuscript for written HW3 as a single PDF file.

- The homework is **due at 8:00 PM** on the due date. We will be using Gradescope for collecting the homework assignments. Please submit your solution manuscript as a PDF file via Gradescope. Post on Ed Discussion and contact the TAs if you are having technical difficulties in submitting the assignment.

1 Multiple Choice & Written Questions

Note: You do not need to show work for multiple choice questions. If formatting your answer in LaTeX, use our LaTeX template `hw3_template.tex` (This is a read-only link. You'll need to make a copy before you can edit. Make sure you make only private copies.).

1. [Logistic Regression/Regularization] (10 pts) Alice is given a task to classify the data-points given in Figure 1.

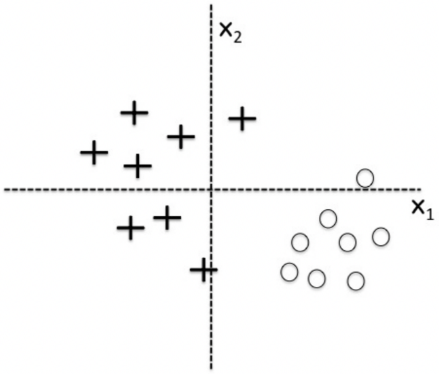

Figure 1: Training Data

Here, the plus signs correspond to class $y = 1$ and the circles correspond to class $y = 0$. She tries to approach the problem with a logistic regression model:

$$P(y = 1|\mathbf{x}, \boldsymbol{\theta}) = h(\theta_0 + \theta_1 x_1 + \theta_2 x_2) = \frac{1}{1 + exp(-\theta_0 - \theta_1 x_1 - \theta_2 x_2)} \tag{1}$$

Alice observes that the datapoints can be perfectly divided with a linear boundary (training error is zero). She plans to train a logistic regression model with the following form of regularization, aiming to minimize:

$$-\sum_{i=1}^{N}[y_i \log h_\theta(x_i) + (1 - y_i)\log(1 - h_\theta(x_i))] + \lambda_0\theta_0^2 + \lambda_1\theta_1^2 + \lambda_2\theta_2^2 \tag{2}$$

for large λ_d, where $d \in \{0, 1, 2\}$. Note that when all λ_d's are equal, this is equivalent to ℓ_2 regularization, discussed in class.

(a) State how the training error changes (increases, decreases, or remains the same with respect to the original training error) for each of the following experiments Alice performs. Briefly justify with reasoning.

i. (2pts) Only regularizes θ_0 (i.e., $\lambda_1 = 0, \lambda_2 = 0$ and λ_0 is large, tending to ∞)

ii. (2pts) Only regularizes θ_1 (i.e., $\lambda_0 = 0, \lambda_2 = 0$ and λ_1 is large, tending to ∞)

iii. (2pts) Only regularizes θ_2 (i.e., $\lambda_0 = 0, \lambda_1 = 0$ and λ_2 is large, tending to ∞)

(b) For the following scenarios, estimate what value(s) Alice can expect θ_0 to take.

i. (2pts) Assume we have equal number of points from each class. We regularize θ_1 and θ_2 (i.e., $\lambda_0 = 0, \lambda_1$ and λ_2 are large, tending to ∞).

ii. (2pts) Assume we have more points for class 1 (plus sign). We regularize θ_1 and θ_2 (i.e., $\lambda_0 = 0, \lambda_1$ and λ_2 are large, tending to ∞).

2. [k Nearest Neighbors] (10 pts) Consider properties of k-NN models:

a. (2 pts) Suppose that we are using k-NN with just two training points, which have different (binary) labels. Assuming we are using $k = 1$ and Euclidean distance, what is the decision boundary? Include a drawing with a brief explanation.

b. (2 pts) For binary classification, given infinite data points, can k-NN with $k = 1$ express any decision boundary? If yes, describe the (infinite) dataset you would use to realize a given classification decision boundary. If no, give an example of a decision boundary that cannot be achieved.

c. (2 pts) Suppose we take $k \to \infty$; what is the resulting model family?

d. (2 pts) What effect does increasing the number of nearest neighbors k have on the bias-variance tradeoff? Explain your answer. [Hint: Use parts (b) and (c) in your explanation.]

e. (2 pts) In logistic regression, we learned that we can tune the threshold of the linear classifier to trade off the true negative rate and the true positive rate. Explain how we can do so for k-NNs for binary classification. [Hint: By default, k-NN uses majority vote to aggregate labels of the k nearest neighbors; consider another option.]

3. [Decision Trees] (15) You are a surfing enthusiast visiting Hawaii for, of course, surfing, but you don't know how to tell if it is a good day for surfing here. So you step into a surf shop, and the shop owner, who happens to love machine learning, shows the following table to you.

Date	Weather	Water Temperature	Wave Height	Good Day to Surf?
1/8	Sunny	Low	$\geq 8ft$	Yes
3/15	Rain	Medium	$< 8ft$	No
4/27	Sunny	High	$\geq 8ft$	Yes
6/1	Cloudy	Low	$< 8ft$	No
7/21	Sunny	Medium	$\geq 8ft$	Yes
8/23	Cloudy	High	$< 8ft$	No
10/6	Rain	Medium	$\geq 8ft$	No
11/9	Cloudy	High	$\geq 8ft$	Yes

The shop owner asks you to construct a decision tree to predict whether it is a good day to surf based on the {`Weather, Water Temperature, Wave Height`} features. You will be using the ID3 algorithm introduced in class. Recall that ID3 will split on the feature that has the largest information gain (or equivalently smallest conditional entropy). The shop owner then hands you a note with the equations below for entropy and information gain, which you may find helpful.

$$\begin{aligned} Entropy(\mathcal{D}) &= -(p_+ log_2 p_+ + p_- log_2 p_-) \\ Gain(\mathcal{D}, X_i) &= Entropy(\mathcal{D}) - \sum_{v \in values(X_i)} \frac{|\mathcal{D}_v|}{|\mathcal{D}|} \cdot Entropy(\mathcal{D}_v) \end{aligned}$$

where $\mathcal{D}_v \subseteq \mathcal{D}$ such that feature X_i has value v.

a. (8 pts) Based on the principle of information gain, decide which attribute is to be used for the first split? Be sure to show your computations.

b. (4 pts) Draw the complete (unpruned) decision tree derived from the ID3 algorithm. Your decision tree should be showing the class predictions at the leaves. You may (1) very neatly hand draw the tree, or (2) draw it using a graphics program, or (3) express the tree in a series of if statements, preferably using LaTeX's `verbatim` environment.

c. (1 pts) Using the Decision Tree constructed in the previous question, predict whether it is a good day to surf when the weather is cloudy, the water temperature is medium, and the wave height is at 11 ft.

d. (2 pt) In general, does the ID-3 algorithm always guarantee a globally optimal tree? By globally optimal, we mean a decision tree that perfectly fits the training data, while having the minimal depth among all trees of such. Briefly justify your answer (you don't need to give a formal proof for this part).

4. [Decision Trees] (5 pts) Describe clearly how to modify a classic decision tree algorithm (ID3 / C4.5) to obtain oblique splits (i.e, splits that are *not* necessarily parallel to an axis). Consider real-valued input features, and describe how you might modify the process of computing gains and selecting good splits. In specific, explain the criterion by which the nodes are split and the loss/optimizer you'll use to select good splits.

5. [Nearest Neighbours; Mandatory for CIS 5190, Optional for CIS 4190] (5 pts) Recall that we can use k-NN for regression by taking the average of the k nearest neighbors. More generally, given a new input x rather than simply choosing the k nearest neighbors of x in the training inputs X, we can think of k-NN as assigning a weight $k_i(x; X)$ to *each* training input x_i based on how "similar" x is to x_i; note that k_i depends on both i and X; e.g., for k-NN, the latter is needed to determine whether x_i is a k nearest neighbor of x. Mathematically, we can express the k-NN prediction for x as

$$f_{\text{KNN}}(x; Z) = \sum_{i=1}^{n} k_i(x; X) y_i \qquad \text{where} \qquad k_i(x; X) = \begin{cases} \frac{1}{k} & \text{if } x \text{ is } k\text{-NN of } x_i \\ 0 & \text{otherwise.} \end{cases}$$

In general, we can use other weighting functions; one option is to use exponential weighting in terms of the Euclidean distance: $k_i(x; X) = e^{-\|x - x_i\|_2^2}/N$, where $N = \sum_{i=1}^{n} e^{-\|x - x_i\|_2^2}$ is a normalizing constant. Show an alternative choice of $k_i(x; X)$ such that the resulting predictions equal the linear regression predictions $f_{\hat{\beta}(Z)}(x) = \hat{\beta}(Z)^\top x$, where $\hat{\beta}(Z) = (X^\top X)^{-1} X^\top Y$ are the linear regression parameters. [Hint: You have previously worked out this formula!]