CIS 4190/5190: Applied Machine Learning — Fall 2022

Homework 3

Handed Out: October 3 — *Due: October 19, 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. [k Nearest Neighbors] 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.]

2. [Decision Trees] In class, we discussed early stopping of generating splits and post-pruning. Here, we consider how they interact.

a. (4 pts) Naïvely, one might expect that if we are using post-pruning on a validation set, then there is never any benefit to using early stopping conditions (e.g., maximum depth to split). Explain why this is not the case.

b. (4 pts) Suppose we are training a decision tree with both early-stopping conditions and post-pruning. For each of the following, indicate whether it increases bias or variance: (i) increase the maximum depth of the decision tree, (ii) increase the minimum number of samples needed to split, (iii) disable post-pruning, and (iv) assuming we are using a feature map, add more features to the feature map.

3. [Decision Trees] You own a movie theater and are trying to understand your market: what types of people frequently go to the movies? You start with the following dataset with data about 6 people with different age groups, income levels, and professions, and whether or not they frequently go to movie theaters. In particular, you are going to build a decision tree to predict whether or not someone is a frequent movie-goer.

No	Age	Income	Profession	Movie-Goer?
1	$A < 30$	Low	Business	Yes
2	$A \geq 30$	High	Engineering	No
3	$A < 30$	Med	Engineering	Yes
4	$A < 30$	Low	Agriculture	No
5	$A \geq 30$	High	Business	Yes
6	$A \geq 30$	Med	Agriculture	No

Recall the following definitions of entropy and information gain, respectively, which are useful for this problem:

$$H(\mathcal{D}) = -\sum_{c} P(Y = c) \log_2 P(Y = c)$$

$$\mathrm{IG}(\mathcal{D}, X_j) = H(\mathcal{D}) - \sum_{v} H(\mathcal{D}[X_j = v])P(X_j = v).$$

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, showing the class predictions at the leaves. Assuming you are using LaTeX, you may (i) very neatly hand draw the tree, photograph it, and include it as a figure, (ii) draw it using a graphics program or PowerPoint, or (iii) express the tree in a series of if statements, preferably using LaTeX's verbatim environment.

c. (2 pts) From the Decision Tree constructed in the previous question, predict whether an engineer who is 23 years old with low income is a movie goer.

4. [Decision Trees] We consider how to extend the decision tree learning algorithm:

a. (4 pts) In class, we focused on categorical features. Typically, for a real-valued feature, the learning algorithm considers splits of the form $x_j \geq t$ vs. $x_j < t$, where $j \in \{1, ..., d\}$

is a component of input x and $t \in \mathbb{R}$ is a real-valued threshold. Describe clearly how you would modify the decision tree learning algorithm to search over potential splits of this form. In particular, describe how you might modify the process of (i) computing gains, and (ii) selecting good splits.

b. (4 pts) [Mandatory for CIS 5190, optional for CIS 4190] Note that using the strategy outlined in (a), the resulting decision tree will have axis-aligned splits. Describe clearly how to modify the decision tree learning algorithm to obtain oblique splits (i.e, splits that are not necessarily parallel to an axis). In particular, describe how you might modify the process of (i) computing gains, and (ii) selecting good splits. What is the effect of using this strategy instead of the previous one on the bias-variance tradeoff?

5. [Nearest Neighbours; Mandatory for CIS 5190, Optional for CIS 4190] (4 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!]