COMP 4745: Machine Learning

Instructor: Deepak Venugopal

Fall 2019: Homework 2

Due Date: October 11, 2019 (ecourseware or you can push the hard copy under my door)

1. Naive Bayes (8 points). Given the below dataset,

W	X	Y	Class
T	T	T	T
T	F	T	F
T	F	F	F
F	T	T	F
F	F	F	T

   - Use Naive Bayes to classify the test example, (W=F,X= F, Y=T) using 1-Laplace correction.
   - Suppose I add 100 more instances that I know have Class=T, (I do not know the feature values for these instances), would your classification using Naive Bayes change?

2. Does the Naive Bayes algorithm accurately compute $P(y|X)$, i.e., the conditional probability of class given features? Briefly explain your answer. (8 points)

3. Your boss gives you a dataset. You analyze the data and conclude that the classes in the data are not linearly-separable, and that each of the features have continuous values. Which of the following algorithms is more ideally suited for your dataset a) perceptron b) decision-trees c) neural-networks d) Naive Bayes. Why? (5 points)

4. What is the leave-one-out cross validation error for the following dataset using i) 1 nearest neighbor algorithm (you can break ties randomly) and ii) 3 nearest neighbor algorithm (9 points)

Feature	Class
1	1
2	1
3	1
4	0
5	0
6	0
7	0
8	1
9	1
10	1

5. Suppose I have a dataset D and I run 1-NN on this dataset treating D as the training data and also predicting the accuracy of 1-NN on D. Can you guarantee that the accuracy of prediction will be 100%? How about if I use 3-NN? Can we guarantee that the accuracy will be 100%? (5 points)

6. In each of the following models, explain if prediction is computationally easier as compared to training the model. i) K-NNs ii) Naive Bayes iii) Neural networks. (6 points)

7. Can we can represent all functions that a decision tree using a Neural Network? If so, can we conclude that Neural network is always a better algorithm for any dataset? (9 points)