CalPolyPomona

CS 4210 – Assignment #2

Maximum Points: 100 pts.

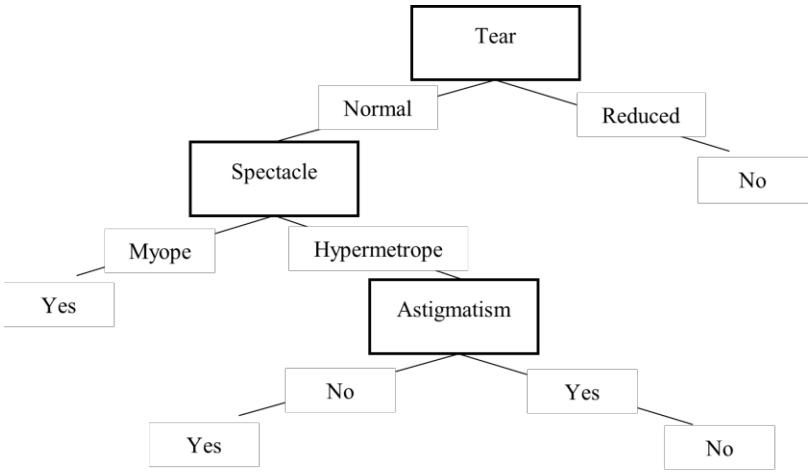
Bronco ID: 014785429

Last Name: James

First Name: Tommy

Note 1: Your submission header must have the format as shown in the above-enclosed rounded rectangle.
Note 2: Homework is to be done individually. You may discuss the homework problems with your fellow students, but you are NOT allowed to copy – either in part or in whole – anyone else's answers.
Note 3: Your deliverable should be a .pdf file submitted through Gradescope until the deadline. Do not forget to assign a page to each of your answers when making a submission. In addition, source code (.py files) should be added to an online repository (e.g., github) to be downloaded and executed later.
Note 4: All submitted materials must be legible. Figures/diagrams must have good quality.
Note 5: Please use and check the Canvas discussion for further instructions, questions, answers, and hints. The bold words/sentences provide information for a complete or accurate answer.

1. [16 points] Considering that ID3 built the decision tree below after analyzing a given training set, answer the following questions:



a) [12 points] What is the accuracy of this model if applied to the test set below? You must **identify each** True Positive, True Negative, False Positive, and False Negative for full credit. For instance: TP = 1,5 | TN = 2,3 …

#	Age	Spectacle	Astigmatism	Tear	Lenses (ground truth)
1	Young	Hypermetrope	Yes	Normal	Yes ✗ FN b
2	Young	Hypermetrope	No	Normal	Yes ✓ TP a
3	Young	Myope	No	Reduced	No ✓ TN d
4	Presbyopic	Hypermetrope	No	Reduced	No ✓ TN d
5	Presbyopic	Myope	No	Normal	No ✗ FP c
6	Presbyopic	Myope	Yes	Reduced	No ✓ TN d
7	Prepresbyopic	Myope	Yes	Normal	Yes ✓ TP a
8	Prepresbyopic	Myope	No	Reduced	No ✓ TN d

b) [4 points] What is the precision, recall, and F1-measure of this model when applied to the same test set?

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

TP = 2 (2, 7)
FP = 1 (5)
TN = 4 (3, 4, 6, 8)
FN = 1 (1)

a) $$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$= \frac{2+4}{8} = \frac{6}{8} = 0.75$$

b)

Precision $$p = \frac{TP}{TP+FP}$$

$$= \frac{2}{2+1} = \frac{2}{3} = 0.67$$

Recall $$r = \frac{TP}{TP+FN}$$

$$= \frac{2}{2+1} = \frac{2}{3} = 0.67$$

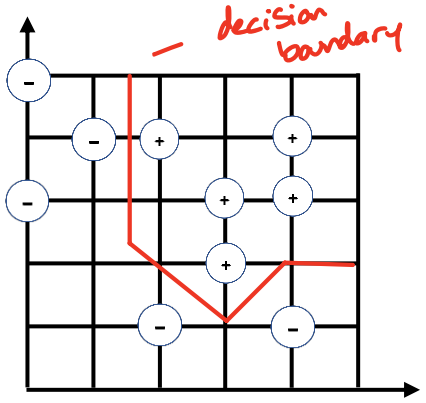
$$F_1 = \frac{2}{\frac{1}{r}+\frac{1}{p}} = \frac{2rp}{r+p}$$

$$= \frac{2(0.67)(0.67)}{0.67+0.67}$$

$$= 0.67$$

2. [15 points] Complete the Python program (decision_tree_2.py) that will read the files contact_lens_training_1.csv, contact_lens_training_2.csv, and contact_lens_training_3.csv. Each of those training sets has a different number of instances. You will observe that now the trees are being created setting the parameter *max_depth = 3*, which it is used to define the maximum depth of the tree (pre-pruning strategy) in *sklearn*. Your goal is to train, test, and output the performance of the **3 models created by using each training set** on the test set provided (contact_lens_test.csv). **You must repeat this process 10 times** (train and test by using a different training set), choosing the average accuracy as the **final classification performance of each model**.

3. [32 points] Consider the dataset below to answer the following questions:



d)

Hide each and find the nearest

a. [4 points] What is the leave-one-out cross-validation error rate (LOO-CV) for **1NN**? Use Euclidean distance as your distance measure and the error rate calculated as:

$$error\ rate = \frac{number\ of\ wrong\ predictions}{total\ number\ of\ predictions}$$

draw lines

b. [4 points] What is the leave-one-out cross-validation error rate (LOO-CV) for **3NN**?

c. [4 points] What is the leave-one-out cross-validation error rate (LOO-CV) for **9NN**?

d. [5 points] Draw de **decision boundary** learned by the 1NN algorithm.

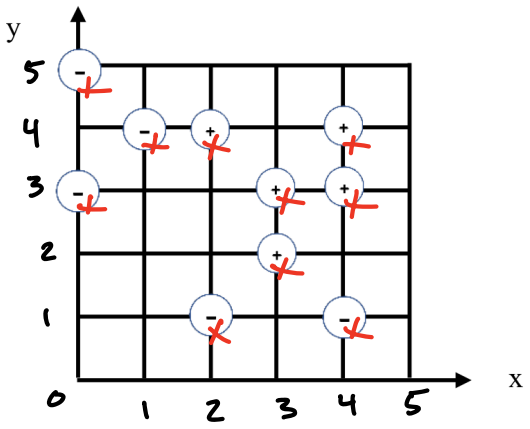
don't use a)

e. [15 points] Complete the Python program (knn.py) that will read the file binary_points.csv and output the LOO-CV error rate for 1NN (**same answer of part a**).

4. [12 points] Find the class of instance #10 below following the 3NN strategy. Use Euclidean distance as your distance measure. You must **show all your calculations** for full credit.

ID	Red	Green	Blue	Class
#1	220	20	60	1
#2	255	99	21	1
#3	250	128	14	1
#4	144	238	144	2
#5	107	142	35	2
#6	46	139	87	2
#7	64	224	208	3
#8	176	224	23	3
#9	100	149	237	3
#10	154	205	50	?

find 3 instances that are nearest



	x	y	class
1	2	1	1
2	4	1	1
3	3	2	2
4	3	3	2
5	4	3	2
6	4	4	2
7	2	4	2
8	1	4	1
9	0	3	1
10	0	5	1

+ = 2

- = 1

$$\text{error rate} = \frac{\#\text{ of wrong predictions}}{\text{total }\#\text{ of predictions}}$$

a) 1NN : error rate $= \frac{4}{10} = 0.40$

b) 3NN : error rate $= \frac{2}{10} = 0.20$

c) 9NN : error rate $= \frac{10}{10} = 1$

4. [12 points] Find the class of instance #10 below following the 3NN strategy. Use Euclidean distance as your distance measure. You must **show all your calculations** for full credit.

ID	Red	Green	Blue	Class
#1	220	20	60	1
#2	255	99	21	1
#3	250	128	14	1
#4	144	238	144	2
#5	107	142	35	2
#6	46	139	87	2
#7	64	224	208	3
#8	176	224	23	3
#9	100	149	237	3
#10	154	205	50	? 2

$d(10,1) = \sqrt{(154-220)^2 + (205-20)^2 + (50-60)^2} = 196.67$

$d(10,2) = \sqrt{(154-255)^2 + (205-99)^2 + (50-21)^2} = 149.26$

$d(10,3) = \sqrt{(154-250)^2 + (205-158)^2 + (50-147)^2} = 128.22$

$d(10,4) = \sqrt{(154-144)^2 + (205-238)^2 + (50-144)^2} = 100.12$

$d(10,5) = \sqrt{(154-107)^2 + (205-142)^2 + (50-35)^2} = 61.67$

$d(10,6) = \sqrt{(154-46)^2 + (205-139)^2 + (50-87)^2} = 131.87$

$d(10,7) = \sqrt{(154-64)^2 + (205-224)^2 + (50-208)^2} = 182.83$

$d(10,8) = \sqrt{(154-176)^2 + (205-224)^2 + (50-23)^2} = 39.67$

$d(10,9) = \sqrt{(154-100)^2 + (205-149)^2 + (50-237)^2} = 202.54$

The class of instance # 10 is 2

5. [25 points] Use the dataset below to answer the next questions:

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

9 ⊕ $\frac{9}{14}$

5 ⊖ $\frac{5}{14}$

a) [10 points] Classify the instance ‹D15, Sunny, Mild, Normal, Weak› following the Naïve Bayes strategy. **Show all your calculations** until the final normalized probability values. write prediction

b) [15 points] Complete the Python program (naïve_bayes.py) that will read the file weather_training.csv (training set) and output the classification of each test instance from the file weather_test (test set) **if the classification confidence is >= 0.75**. Sample of output:

Day	Outlook	Temperature	Humidity	Wind	PlayTennis	Confidence
D15	Sunny	Hot	High	Weak	No	0.86
D16	Sunny	Mild	High	Weak	Yes	0.78

Important Note: Answers to all questions should be written clearly, concisely, and unmistakably delineated. You may resubmit multiple times until the deadline (the last submission will be considered).

NO LATE ASSIGNMENTS WILL BE ACCEPTED. ALWAYS SUBMIT WHATEVER YOU HAVE COMPLETED FOR PARTIAL CREDIT BEFORE THE DEADLINE!

5a)

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Classify the instance ‹D15, Sunny, Mild, Normal, Weak›

P(PlayTennis = yes) = 9/14 = 0.64

P(PlayTennis = no) = 5/14 = 0.36

outlook	Yes	No
Sunny	2/9	3/5
overcast	4/9	0/5
Rain	3/9	2/5

Humidity	Yes	No
high	3/9	4/5
normal	6/9	1/5

Temperature	Yes	No
Hot	2/9	2/5
mild	4/9	2/5
cool	3/9	1/5

windy	Yes	No
weak	6/9	2/5
Strong	3/9	3/5

P(Yes | Sunny, Mild, Normal, weak)

= P(Sunny | Yes) P(Mild | Yes) P(Normal | Yes) P(weak | Yes) P[Yes)

= (2/9 · 4/9 · 6/9 · 6/9) · (9/14) = 0.0282

$P(\text{No} \mid \text{Sunny, Mild, Normal, weak})$

$= P(\text{Sunny} \mid \text{No})\ P(\text{Mild} \mid \text{No})\ P(\text{Normal} \mid \text{No})\ P(\text{weak} \mid \text{No})\ P(\text{No})$

$= (3/5 \cdot 2/5 \cdot 1/5 \cdot 2/5) \cdot (5/14) = 0.0069$

Normalize

$P(\text{Yes} \mid \text{Sunny, Mild, Normal, weak})$

$= \frac{0.0282}{0.0282 + 0.0069} = 0.803$

$P(\text{No} \mid \text{Sunny, Mild, Normal, weak})$

$= \frac{0.0069}{0.0282 + 0.0069} = 0.197$

The most prdoable classification is Yes

2. GitHub.com/ttjamescpp/cs4210-Assignment2

3e. GitHub.com/ttjamescpp/cs4210-Assignment2

5b. GitHub.com/ttjamescpp/cs4210-Assignment2