**CSci 384: Artificial Intelligence** Date: May 7th, 2020

**Due: by the end of day, May 15th (Fri.)**

**Final Exam:**

Name: \_\_\_\_\_\_\_\_Derek Trom\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **Total: 250 points + 50 (optional)**

**Instruction:**

**Q4 – Q6.** In order to compute the required probability, you have to define the variable and the proper formula of probability with the defined (or given) variables and show the ***essential*** computational steps precisely; for instance,

1. Define what needs to be computed: e.g.) P(C|B)
2. Derive the formulas for computation in the essential steps:

e.g.) P(C|B) = P(B| C)⋅P(C)

1. Assign the values to the derived formula to complete the computation.

e.g.) P(C|B) = P(B|C)⋅P(C ) = 0.2 ⋅ 0.6 = 0.12

**Q7 – Q8.** In order to get the hypothesis or do prediction, you have to define the proper formula and show the ***essential*** computational steps.

**Q9 – Q11**. Please refer to the slides, handout and the lecture video for the topics whose home assignment was not offered.

**Q12.** You have to clearly define the formula of information gain or the entropy and show their ***essential*** computational steps. In the full decision tree, the final classification should be specified in each leaf.

• Any answer without clear computational steps or sufficient description will **NOT** get a full point while an excellent answer would be ***rewarded***.

• If you like to, you can ***write*** or ***reuse a program*** for any computation; but, please show the ***essential steps*** and ***values*** in your answer as well as the final answer: e.g.) an entropy or an information gain of an attribute, Bayesian learning, etc.

• You should work on the exam ***independently.*** ***Any kind of plagiarism will be graded as zero point for the final exam.***

• Any answer which is unreadable due to the unrecognizable symbols will get zero point. – check your file in the Window system and convert the symbols properly (if you’ve created it in Mac.).

• Hours taken to complete the exam: \_\_11\_ Hours \_0\_\_ Minutes.

• Mark the difficulty of the Exam:

Very Easy: \_\_\_\_\_ Easy: \_\_\_\_\_ Moderate: \_\_X\_\_\_ Difficult: \_\_\_\_ Very Difficult: \_\_\_\_\_

**Q1. [10] Propositional Logic.**

Prove the following sentence is ***valid, unsatisfiable*** or ***satisfiable*** by applying a ***sequence of logical inference procedures***, **not** by truth table enumeration.

(I pass CSci 384 and I do not make an A) and (If I pass CSci 384 then I make an A)

First, convert the sentence in the Propositional Logical sentence by defining the propositional symbols and connectives; then, prove it.

P = Pass 384

A = Make an A

(P ∧¬A) ∧(P→A)

(P ∧¬A) ∧(¬P∨A) -implication elim

P ∧(¬P∨A) -simplification

A -disjunctive inference

Unsatisfiable when A is False

**Q2 [25] 1st-Order Logic (FOL).**

Convert the following sentences into 1st-order logic sentences, using the following predicates:

• *P(x)* = *x* is a programmer. • *R(x)* = *x* is red-haired.

• *S(x)* = *x* is smart. • *L(x, y)* = *x* likes *y.*

1. [5] No programmer is smart.
   1. x(P(x)->S(x))
2. [5] Not everyone is a programmer.
   1. x P(x)
3. [5] Everyone is not a programmer.
   1. xP(x)
4. [5] Some programmers are not smart.
   1. x (P(x) ∧ S(x))
5. [5] There is some with red-hair whom everyone likes.
   1. xy (R(y) ∧ L(x,y))

**Q3. [35]** **Inference in the 1st-Order Logic.**

Consider the following statements.

* (A) No software is guaranteed.
* (B) All programs are software.
* **Conclusion:** Thus, no program is guaranteed.

1. [10] Translate the above statements in the 1st –Order Logical sentences using the following predicates. Clearly use the universal and/or existential quantifiers.

• *S(x)* = *x* is a software. • *G(x)* = *x* is guaranteed. • *P(x)* = *x* is a program.

1. x(S(x)->G(x))
2. x(P(x)->S(x))
3. x(P(x)->G(x))
4. [5] Negate the conclusion in (1).
   1.  x (P(x) ∧ G(x))
5. [10] Convert your sentences in (1.A & 1.B) and the negated conclusion in (2) to Conjunctive Normal Form (CNF).
   1. x(S(x)->G(x)) = x(S(x) G(x))
   2. x(P(x)->S(x)) = x (P(x) S(x))
   3.  x (P(x) ∧ G(x)) = P(K) ∧ G(K) where K is skolem
6. [10] Using ***resolution***, prove the conclusion is either true or false. Show you proof clearly with the substitution.
   1. If P(K) ∧ G(K) = true, P(K) must be true and G(K) must be true. If that’s the case, for P(x) S(x) to be true, S(x) = true because P(x) = false . For (S(x) G(x)) to be true, either G(x) has to be false or S(x) has to be false. Because both S(x) and G(x) are implied true, the conclusion must be false as G(x) cannot be both true and false.

**Q4. [20] Uncertainty.**

A screening test of virus is a low-cost way of checking large groups of people for a virus. A more costly

but accurate test shows that 1% of all people have the disease. The screening test indicates the virus

(test positive(+)) in 98% of those who have it, and in 10% of those who do not have the disease (false

positive(+)).

For the following questions, define the variables, their domains and the probability to compute and compute it by showing its essential computational steps.

P(a) is the percentage of people who do not have the disease, or .99

P(b) is the percentage of people who do have the disease, or .01

P(c|b) is the percentage of people who test positive for the disease who have it, or .98

P(c|a) is the percentage of people who give a false positive test for the disease, or .1

P(e) is the percentage of people who test positive for the disease which is P(b)\*P(c)+P(a)\*P(d) or (.01\*.98)+(.99\*.1) = .1088 or 10.88%

P(f) is the percentage of people who test negative for the disease which is ((1-P(c))\*P(b))+((1-P(d))\*P(a)) or (.1\*.01)+(.98\*.99) = .9712

1. [10] What percent of people who test positive don’t have the virus (false +)?
   1. P(a|c) = (P(c|a)\*P(a))/P(e) = (.1\*.99)/.108 = .9166667 or 91.6667%
2. [10] What percent of people who test negative do have the virus (false -)?
   1. P(b|f) = ((1-P(c|b))\*P(b))/P(f) = (.1\*.01)/.9712 = .0010297 or 0.103%

**Q5. [25] Bayesian Network and Inference**

NOTE: All the hypotheses and the numeric values are randomly given, neither based on the medical fact nor on the statistics of each disease.

Suppose you want the diagnostic assistant to be able to reason about the possible causes of a patient’s coughing and the shortness of breath.

The agent can observe coughing, fever, chills and the shortness of breath from patients.

The agent can ask whether the patient smokes.

The agent can use Bronchitis and COVID-19 to predict the outcomes of patients.

The following statistics has been reported and will be used for further inference.

1. The 2% of patients were diagnosed for ***COVID-19***.
2. The 20% of patients were observed for ***smoking*** a cigarette.
3. Whether patients have ***chills*** depends on whether they have ***COVID-19*** or not.
   1. 30% of patients who have COVID-19 have chills while 0.1% of patients who don’t have COVID-19 have chills.
4. Whether patients have ***fever*** depends on whether they have ***COVID-19*** or not.
   1. 90% of patients who have COVID-19 have fever while 5% of patients who don’t have COVID-19 have fever.
5. Whether patients have bronchitis depends on whether they have COVID-19 and whether they smoke;
   1. 95% of patient have bronchitis if (s)he has COVID-19 and smokes.
   2. 85% of patient have bronchitis if (s)he has COVID-19 but doesn’t smoke.
   3. 70% of patient have bronchitis if (s)he doesn’t have COVID-19 but smokes.
   4. 1% of patient have bronchitis if (s)he neither has COVID-19 nor smokes.
6. Whether patients have ***the shortness of breath*** depends on whether they have ***bronchitis*** or not.
   1. 60% of patients who have bronchitis have the shortness of breath while 2% of patients who don’t have bronchitis have the shortness of breath.
7. Whether patients do ***coughing*** depends on whether they have ***bronchitis*** or not.

80% of patients who have bronchitis do coughing while only 7% of patients who don’t have bronchitis do coughing.

1. [10] Using the given Boolean variables below, define the above statistics (A) – (G) in the formula of probability with the given values. Use the following variables:

*CV* –COVID-19, S – Smokes, CH – Chills, F - Fever,

B – Bronchitis, C – Coughing, SB – Shortness of Breath.

e.g.) P(SB | CV ) = 0.2

P(CV) = 0.02

P(S) = 0.2

P(CH|CV) = 0.3

P(CH|¬CV) = 0.001

P(F|CV) = 0.9

P(F|¬CV) = 0.05

P(B|CV,S) = 0.95

P(B|CV¬S) = 0.85

P(B|¬CV,S) = 0.7

P(B|¬CV,¬S) = 0.01

P(SB|B) = 0.6

P(SB|¬B) = 0.02

P(C|B) = 0.8

P(C|¬B) = 0.07

1. [5] Draw a Bayesian Network which represents the above information correctly and give the (conditional) probability table for each node.

|  |  |
| --- | --- |
|  | P(CV) |
| T | .02 |
| F | .98 |

|  |  |
| --- | --- |
|  | P(S) |
| T | .2 |
| F | .8 |

|  |  |
| --- | --- |
| CV | P(CH|CV) |
| T | .3 |
| F | .001 |

|  |  |
| --- | --- |
| B | P(SB|B) |
| T | .6 |
| F | .02 |

|  |  |
| --- | --- |
| B | P(C|B) |
| T | .8 |
| F | .07 |

|  |  |  |
| --- | --- | --- |
| CV | S | P(B|CV,S) |
| T | T | .95 |
| T | F | .85 |
| F | T | .7 |
| F | F | .01 |

|  |  |
| --- | --- |
| CV | P(F|CV) |
| T | .9 |
| F | .05 |

[10] ***Define*** and ***compute*** the ***probability that patients who had the shortness of breath and fever had smoked***. – The tense of verbs can be ignored.

P(S|SB,F)=

Σ P(s) Σb P(sb| b) [ P(b| cv, s) P( f | cv)P(cv) + P(b| ¬cv, s) P( f | ¬cv)P(¬cv)]

P(s) Σb P(sb| b) [ P(b| cv, s)\*.9\*.02 + P(b|¬cv, s)\*.05\*.98]

P(s) { P(sb| b) [ P(b| cv, s)\*.9\*.02 + P(b|¬cv, s)\*.05\*.98 ]

+ P(sb| ¬b) [ P¬b| cv, s)\*.9\*.02 + P(¬b| ¬cv, s)\*.05\*.98 ] }

.2{.6[.95\*.9\*.02 + .7\*.05\*.98]+.02[.05\*.9\*.02+.3\*.05\*.98]}

.2{.6[.0514]+.02[.0156]} = 0.0062304 given S

P(¬s) { P(sb| b) [ P(b| cv, ¬s)\*.9\*.02 + P(b|¬cv, ¬s)\*.05\*.98 ]

+ P(sb| ¬b) [ P¬b| cv, ¬s)\*.9\*.02 + P(¬b| ¬cv, ¬s)\*.05\*.98 ] }

.8{.6[.85\*.9\*.02 + .01\*.05\*.98] + .02[.15\*.9\*.02+.99\*.05\*.98]}

.8{.6[.01579]+.02[.05121]}

.8{.009474+.0010242}

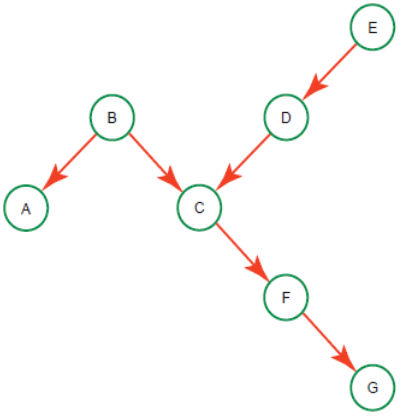
= .00839856 when ¬S

<0.0062304, 0.00839856> normalized ~= <0.0062304/0.01462896, 0.00839856/0.01462896> ~=

<0.4259, 0.5741>

**Q6. [25] Bayesian Network**

In the given Bayesian Network (BN) below,



*P(b) = 0.7 P(e) = 0.91*

*P(a | b) = 0.8 P(a | ¬ b) = 0.3*

*P(c | b, d) = 0.9 P(c | b, ¬ d) = 0.3 P(c | ¬b, d ) = 0.5 P(c | ¬ b, ¬ d) = 0.8*

*P(d | e ) = 0.1 P(d | ¬ e) = 0.8*

*P(f | c ) = 0.4 P(f | ¬ c) = 0.8*

*P(g | f ) = 0.2 P(g | ¬ f) = 0.9*

1. [5] Find the nodes which is ***conditionally independent*** of ’C’ given its parent ’B’ and ’D’.
   1. E and A
2. [5] (a) Find the ***Markov blanket*** of a node ‘B’.
   * 1. Parents = None + Children=A,C + Childrens parents = D = A,C,D

(b) Indicate what nodes are conditionally independent of B given its Markov blanket found in

i. E, F, G

1. [5] Find the irrelevant node(s) to compute P(C | G).
   1. A
2. [5] Give the **formula** of computing the full joint distribution **P(*a, b, c, ¬ d, ¬ e, ¬ f, g*)** where *a, b, … ¬ f, g* are propositions such that A = true, B = true, …. F = false and G=true.
   1. P(A|B)\*P(B)\*P(C|B,¬D)\*P(¬D| ¬E)\*P(¬E)\*P(¬F|C)\*P(G|¬F)
3. [5] Compute P(*a, b, c, ¬ d, ¬ e, ¬ f, g*) in (4).
   1. .8\*.7\*.3\*.2\*.09\*.6\*.9 = 0.00163296

**Q7. [40] Bayesian Learning**

Consider a medical diagnosis problem in which there are two alternative hypotheses:

* *h*1: the patient has a particular form of cancer,
* *h*2 = *¬h*1: the patient does not has a cancer.

The available data is from a particular lab test with two possible outcomes: + (positive) and *−*(negative). We have prior knowledge that over the entire population of people only *.*008 have this disease. Further- more, the lab test is only an imperfect indicator of the disease. The test returns a correct positive result in only 98% of the cases in which the disease is actually present and a correct negative result in only 97% of the cases in which the disease is not present. In other cases, the test returns the opposite result.

1. [10] Compute the following probabilities which summarize the above situation
   1. *P* (*h*1) = *P* (*cancer*) = .008
   2. *P* (*h*2) = *P* (*¬cancer*)= .992
   3. *P* (+*|h*1)= .98
   4. *P* (*−|h*1)= .02
   5. *P* (+*|h*2)=.03
   6. *P* (*−|h*2)=.97
2. [5] Suppose we now observe a new patient for whom the lab test returns a positive result,

< d1= + >. What is the ***maximum a posteriori (MAP)* hypothesis**?

P(h1|d1 = +) = P(d1=+|h1)P(h1)= .98\*.008 = 0.00784

P(h2|d1 = +) = P(d1=+|h2)P(h2) = .03 \*.992 = 0.02976

<0.00784, 0.02976> normalize = <0.209, 0.791>

Thus hmap = h2

1. [5] With the lab test of the positive result in (2), what is the ***maximum likelihood (ML)* hypothesis**?
   1. P(+/h1) = 0.98 > P(+/h2)= 0.03
   2. Thus ML = h1
2. [10] Suppose the doctor decides to order a second laboratory test for the same patient, and suppose the second test returns a positive result as well, < d2= + >. What are the ***posterior probabilities*** of *cancer* (*h*1)and *¬cancer* (*h*2) following these two tests of evidence? Assume that the two tests are independent.

P(h1|d1 = +, d2=+) = P(d1=+|h1)2P(h1)= .982\*.008 = 0.00768

P(h2|d1 = +, d2 = +) = P(d1=+|h2)2P(h2) = .032\*.992 = 0.0008928

<0.00768, 0.0008928> normalize <.8953, .1047>

1. [10] What is your ***prediction*** for the 3rd test result (**d3**), based on the previous two lab test results (*d*1*, d*2)? Compute the predicted probability that the 3rd test result is also ***positive***,

i.e. *P* (d3= +|*d*1 = +*, d*2 = +)

* 1. [5, optional] by Full Bayesian learning
     1. <P(+|h1) P(h1|d1 = +, d2=+)+P(+|h2) P(h2|d1 = +, d2 = +), P(-|h1) P(h1|d1 = +, d2=+)+P(-|h2) P(h2|d1 = +, d2 = +)>
     2. <0.98\*0.8953+0.03\*0.1047,0.02\*0.8953+0.97\*.1047>
     3. <0.8805,0.1195>
     4. Thus it will be predicted positive
  2. [5] by MAP learning
     1. If we use hmap from 4 prediction will be h1 for the 3rd test as well given h1>h2 and it will predict positive.
  3. [5] by ML learning
     1. P(d1,d2|h1) = P(+|h1)2 =.982
     2. P(d1,d2|h2) = P(+|h2)2= .032
     3. .982>.032 so hml= h1
     4. Thus it will also predict positive.

**Q8. [25] Naïve Bayesian Model**

We have 2 classes of movies: *NEW* and *OLD*.

The following training set of 3 Boolean attributes, x, y, z, and a class, C, represent each of three features of movie and the class of movie, respectively, where 1 = true and 0 = false.

Suppose you have to predict Class of a movie using a Naïve Bayes Model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data | x | y | z | Class |
| D1 | 1 | 0 | 0 | OLD |
| D2 | 0 | 1 | 0 | NEW |
| D3 | 1 | 0 | 0 | NEW |
| D4 | 0 | 0 | 1 | NEW |
| D5 | 0 | 1 | 1 | OLD |
| D6 | 1 | 1 | 0 | NEW |
| D7 | 0 | 1 | 0 | OLD |

1. [5] What is P(*NEW* | *x* = 0) learned for the above training data?
   1. P(*NEW* | *x* = 0) = 2/3
2. [5] What is P(*NEW* | *x* = 0; *y* = 1; *z* = 0) learned for the above training data?
   1. P(*NEW* | *x* = 0; *y* = 1; *z* = 0) = 1/2
3. [5] After learning is complete, what would be the ***predicted probability***
   1. P(*NEW* | *x* = 0, *y* = 1, *z* = 0)=
   2. P(NEW)\* P(x=0|NEW)\*P(y=1|NEW)\*P(z=0|NEW)/
   3. P(NEW) \*P(*x* = 0, *y* = 1, *z* = 0|NEW)+P(OLD)\*P(*x* = 0, *y* = 1, *z* = 0|OLD)
   4. (4/7\*1/2\*1/2\*3/4) /(4/7\*1/4)+(3/7\*1/3)= .1071/.1429+.1429 = .3747
4. [5] How would a Naïve Bayesian Model ***predict*** ***Class*** given the input < *x*=0, *y*=1, *z*=0 >? Assume that in case of a time the classifier always prefers to predict *NEW* for Class.
   1. P(OLD) = 3/7 P(NEW) = 4/7
   2. P(OLD| *x*=0, *y*=1, *z*=0) =
   3. P(OLD)\*P(x=0|OLD)\*P(y=1|OLD)\*P(z=0|OLD)
   4. =3/7\*2/3\*2/3\*2/3 = .1269/.234 = 0.542 normalized
   5. P(NEW | *x*=0, *y*=1, *z*=0) =
   6. P(NEW)\* P(x=0|NEW)\*P(y=1|NEW)\*P(z=0|NEW)
   7. = 4/7\*1/2\*1/2\*3/4 = .1071/.234 = 0.458 normalized
   8. Since OLD > NEW in this case it will predict OLD
5. [5] Using the probabilities obtained during the Bayes classifier training, compute the ***predicted probability*** P(*OLD* | x = 0).
   1. P(OLD| x=0) =
   2. P(OLD)\*P(x=0|OLD)/P(OLD)\* P(x=0|OLD)+ P(NEW)\* P(x=0|NEW)
   3. (3/7\*2/3)/(2/3\*3/7)+ (4/7\*1/2) = .2857/.5714 = 0.5

**Q9. [15] Decision Making**

Consider a student who has the choice to buy or not buy a textbook for a course.

We'll model this as a decision problem with one Boolean decision node, B, indicating whether the agent chooses to buy the book, and two Boolean chance nodes, M, indicating whether the student has mastered the material in the book, and P, indicating whether the student passes the course. Of course, there is also a utility node, U. A certain student, Sam, has an additive utility function: 0 for not buying the book and -$100 for buying it; and $2000 for passing the course and 0 for not passing.

Sam's conditional probability estimates are as follows:

*P(p|b, m) = 0.8 P(p|¬b, ¬m) = 0.2*

*P(p|b, ¬m) = 0.4 P(m|¬b) = 0.7*

*P(p|¬b, m) = 0.7*

*P(m|b) = 0.8*

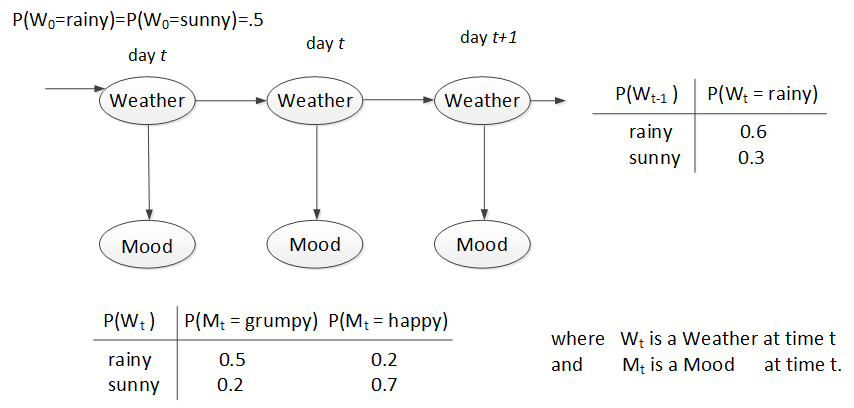
You might think that P would be independent of B given M, But this course has an openbook final-so having the book helps.

1. [10] Compute the **expected utility** of ***buying the book*** and that of ***not buying*** it, respectively.
   1. P(p|b) = P(p|b,m)P(m|b) + P(p|b, ¬m)P(¬m|b)
   2. = 0.8\*0.8 + 0.4\*0.2 = 0.72
   3. P(p|¬b) = P(p|¬b,m)P(m|¬b)+ P(p|¬b, ¬m)P(¬m|¬b)
   4. = 0.7\*0.7 + 0.2 \*0.3 = 0.55
   5. EU(b) = .72(2000-100)+.28(-100)= 1340
   6. EU(¬b) = .55\*2000 + .28\*0 = 1100
2. [5] What should Sam do?
   1. Because the EU of buying a book is greater than that of not buying a book Sam should buy the book.

**Q10. [25] HMM(Optional)**

Andrew is in a self-quarantine in the room with no window, so his mood is affected by the weather state. Some days, it’s rainy and some days it’s sunny. But he is unable to check the weather, and so all we can observe is his mood: grumpy, happy or sad.

We start on day 0 in the Rainy state, and there’s one transition per day.



Wt = Weather State on day t ∈ {rainy, sunny}

Mt = Mood on day t ∈ {grumpy, happy, sad}

P(W0=Rainy) = 1

1. [5] Compute P(*W2* = *sunny*).
   1. P(*W2* = *sunny*) =
   2. P(W2=sunny|W1=rainy) P(W1=rainy|W0=rainy)P(W0=rainy)+ P(W2=sunny|W1=sunny) P(W1=sunny|W0=rainy)P(W0=rainy)
   3. .4\*.6\*1+.7\*.4\*1 = 0.52
2. [5] Compute P(*M2* = *grumpy*).
   1. P(*W2* = *sunny*) P(M2=grumpy| *W2* = *sunny)+P(W2* = *rainy)P(*M2=grumpy| *W2* = *rainy)*
   2. =.52\*.2+.48\*.5 = 0.344
3. [5] Compute P(*W2* = *sunny* | *M2* = *grumpy*).
   1. P(*W2* = *sunny*) P(M2=grumpy| *W2* = *sunny)/* P(*M2* = *grumpy*)
   2. .2\*.52/.344=0.302
4. [5] Compute P(*M200* = *happy*)*.*
   1. P(Mt=happy)
   2. P(Mt=happy,Wt=sunny)+ P(Mt=happy,Wt=rainy)
   3. P(Mt=happy|Wt=sunny)P(Wt=sunny)+ P(Mt=happy|Wt=rainy) P(Wt=rainy)
   4. 0.7\* P(Wt=sunny)+0.2 \*P(Wt=rainy)
   5. 0.7\* P(Wt=sunny)+0.2(1- P(Wt=sunny))
   6. 0.5\* P(Wt=sunny) +0.2
   7. I stopped here because I did not want to write a program for the M200
5. [5] Assume that *M1* = *sad, M2* = *sad, M3* = *grumpy, M4* = *grumpy, M5* = *happy* are observed for five days. What is the most likely *weather state* on day 5?

**Q11. [20, optional] Nearest Neighbor and kNN classification**

In the table of the training data, classifiy a test data. Use the Manhattan distance d(a, b), defined as

***d***(***a, b***) = |*a1* – *b1* | + | *a2* – *b2*|.

|  |  |  |  |
| --- | --- | --- | --- |
| Data | *x1* | *x2* | **Class** |
| v1 | 6 | 1 | 0 |
| v2 | 7 | 3 | 0 |
| v3 | 8 | 2 | 0 |
| v4 | 9 | 0 | 0 |
| v5 | 8 | 4 | 1 |
| v6 | 8 | 6 | 1 |
| v7 | 9 | 2 | 1 |
| v8 | 9 | 5 | 1 |

1. [10] Classify the data v9 = (9, 3.5) with the Nearest Neighbor method.
   1. D(v1, v9) = |6-9 |+ |1-3.5| = 3+2.5 = 5.5
   2. D(v2,v9) = |7-9| + |3-3.5| = 2+.5 = 2.5
   3. D(v3,v9) = |8-9|+|2-3.5| = 1+1.5 = 2.5
   4. D(v4,v9) = |9-9|+ |0-3.5| = 0+3.5 = 3.5
   5. D(v5,v9) = |8-9|+|4-3.5| = 1+.5 = 1.5
   6. D(v6,v9) = |8-9|+|6-3.5| =1+2.5 = 3.5
   7. D(v7,v9) = |9-9|+|2-3.5| = 0+1.5 = 1.5
   8. D(v8,v9)= |9-9|+|5-3.5| = 0 + 1.5 = 1.5
   9. V9 will be class 1 as the lowest of the distances are 1.5 and they are all class 1.
2. [10] Classify the data v9 in (1) with k-Nearest Neighbor method for *k*=5.
   1. 3/5 nearest neighbors are class 1 thus it would classify as class 1.

**Q12. [30] Decision Tree Learning**

The following dataset will be used to learn a decision tree for predicting whether a strawberry is edible or not based on its shape, color and odor.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Example | Shape | Color | Odor | **Edible** |
| e1 | A | R | 1 | Yes |
| e2 | A | R | 2 | Yes |
| e3 | A | V | 2 | No |
| e4 | A | R | 3 | No |
| e5 | A | P | 3 | No |
| e6 | B | R | 1 | Yes |
| e7 | B | P | 1 | Yes |
| e8 | B | P | 2 | Yes |
| e9 | B | R | 2 | No |
| e10 | B | G | 2 | No |
| e11 | B | P | 3 | No |

1. [6] What is the entropy H(Edible | Order = 1 or Odor = 3)?
   1. -(1/2log(1/2)+1/2log(1/2)) = 1
2. [6] Which ***attribute*** would the decision tree algorithm choose to use the ***root*** of the tree? Show the computation of information gain for each attribute for the decision of root.
   1. Shape
      1. E(shape = A) = -(2/5log2/5) – 3/5log3/5 = .971
      2. E(shape = B)= -(3/6log3/6) – 3/6log3/6 = 1
      3. I(shape) = 5/11\*.971 + 1\*6/11 = .9868
      4. Gain(shape) = .994 - .9868 = 0.0072
   2. Color
      1. E(color = r) = -(3/5log3/5) – 2/5log2/5 = .971
      2. E(color = v) = -(0/1log0/1) -1/1log1/1 = 0
      3. E(color = P) = -(2/4log2/4)-2/4log2/4 = 1
      4. E(color = G) = 0
      5. I(color) = .971 \* + 5/11 + 1\*4/11 = .441 + .3636 = .8046
      6. Gain(color) = .994 -.8046 = .1893
   3. Odor
      1. E(odor = 1) = 0
      2. E(odor = 2) = -(2/5log2/5) – (3/5log3/5) = .971
      3. E(odor=3) = 0
      4. I(odor) = .971 \*5/11 = .4414
      5. Gain(odor) = .5526
   4. Because odor information gain is the greatest it will be chosen as the root
3. [6] Draw the ***full decision tree*** that would be learned for this data.
   1. Calculation shown below. Copied and pasted from pdf in file attached named Output12.6.pdf
4. [6] Express the ***hypothesis*** of (3) in the ***logical sentence***.
   1. ∀xEdible(x) = yes↔(Odor(x) = 1) ∨(Odor(x) =2 ∧Color(x)=P) ∨( Odor(x) =2∧Color(x)=R ∧Shape(x) = A) where x is datum
5. [6] Suppose we have a test data set as follows. What will be the ***error rate of the test set*** of the decision tree? i.e. the percentage of examples that would be misclassified by the tree.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Example | Shape | Color | Odor | **Edible** |
| e12 | A | R | 2 | Yes |
| e13 | B | R | 2 | No |
| e14 | A | P | 2 | Yes |
| e15 | B | G | 2 | Yes |

12 = correct

13 = correct

14 = correct

15 = incorrect

This means accuracy = ¾ = 75%

File - HW7-TromDQ1

Page 1 of 2

/usr/local/bin/python3.7 /Users/derektrom/Desktop/

CSCI384/HW7-TromD/HW7-TromDQ1.py

1

2 A gain : 0.0049586776859504855

3 B gain : 0.0049586776859504855

4 P gain : 0.0023612750885478317

5 R gain : 0.03526170798898076

6 G gain : 0.04132231404958686

7 V gain : 0.04132231404958686

8 1 gain : 0.22314049586776868

9 2 gain : 0.0049586776859504855

10 3 gain : 0.15495867768595045

Best Gain: 0.22314049586776868 Best question: Is

Odor "== 1?

11

12 A gain : 0.0

13 B gain : 0.0

14 P gain : 0.0

15 R gain : 0.0

16 Best Gain: 0.0 Best question: Is Color "== R?

17 A gain : 0.0

18 B gain : 0.0

19 P gain : 0.008333333333333415

20 R gain : 0.008333333333333443

21 G gain : 0.01785714285714285

22 V gain : 0.01785714285714285

23 2 gain : 0.07500000000000001

24 3 gain : 0.07500000000000001

Best Gain: 0.07500000000000001 Best question: Is

Odor "== 3?

25

26 A gain : 0.0

27 B gain : 0.0

28 P gain : 0.0

29 R gain : 0.0

30 Best Gain: 0.0 Best question: Is Color "== R?

31 A gain : 0.013333333333333308

32 B gain : 0.013333333333333308

33 P gain : 0.17999999999999994

34 R gain : 0.013333333333333308

35 G gain : 0.07999999999999996

36 V gain : 0.07999999999999996

Best Gain: 0.17999999999999994 Best question: Is

Color "== P?

37

38 Best Gain: 0 Best question: None

39 A gain : 0.125

40 B gain : 0.125

File - HW7-TromDQ1

Page 2 of 2

41 R gain : 0.125

42 G gain : 0.041666666666666685

43 V gain : 0.041666666666666685

44 Best Gain: 0.125 Best question: Is Color "== R?

45 A gain : 0.5

46 B gain : 0.5

47 Best Gain: 0.5 Best question: Is Shape "== B?

48 Best Gain: 0 Best question: None

49 Best Gain: 0 Best question: None

50 A gain : 0.0

51 B gain : 0.0

52 G gain : 0.0

53 V gain : 0.0

54 Best Gain: 0.0 Best question: Is Color "== V?

55 Is Odor "== 1?

56 ""--> True:

57 Predict {'Yes': 3}

58 ""--> False:

59 Is Odor "== 3?

60 ""--> True:

61 Predict {'No': 3}

62 ""--> False:

63 Is Color "== P?

64 ""--> True:

65 Predict {'Yes': 1}

66 ""--> False:

67 Is Color "== R?

68 ""--> True:

69 Is Shape "== B?

70 ""--> True:

71 Predict {'No': 1}

72 ""--> False:

73 Predict {'Yes': 1}

74 ""--> False:

75 Predict {'No': 2}

76

77 Process finished with exit code 0