COMP 2211 Exploring Artificial Intelligence

Practice Problems: Naive Bayes, KNN, K-Means, Perceptron, and MLP

Dr. Desmond Tsoi

Department of Computer Science & Engineering
HKUST, Hong Kong SAR, China

Naïve Bayes 1

- Given the following:
  - A doctor knows cold causes fever 50% of the time.
  - The probability of any patient having a cold is 1/50000.
  - The probability of any patient having a fever is 1/20.

  If a patient has a fever, what is the probability he/she has a cold?

Naïve Bayes 2

- Given the following dataset:

Animal	Give Birth	Can Fly	Live in Water	Have Legs	Class
Human	Yes	No	No	Yes	Mammals
Python	No	No	No	No	Non-mammals
Salmon	No	No	Yes	No	Non-mammals
Whale	Yes	No	Yes	No	Mammals
Frog	No	No	Sometimes	Yes	Non-mammals
Komodo	No	No	No	Yes	Non-mammals
Bat	Yes	Yes	No	Yes	Mammals
Pigeon	No	Yes	No	Yes	Non-mammals
Cat	Yes	No	No	Yes	Mammals
Leopard Shark	Yes	No	Yes	No	Non-mammals
Turtle	No	No	Sometimes	Yes	Non-mammals
Penguin	No	No	Sometimes	Yes	Non-mammals
Porcupine	Yes	No	No	Yes	Mammals
Eel	No	No	Yes	No	Non-mammals
Salamander	No	No	Sometimes	Yes	Non-mammals
Gila Monster	No	No	No	Yes	Non-mammals
Platypus	No	No	No	Yes	Mammals
Owl	No	Yes	No	Yes	Non-mammals
Dolphin	Yes	No	Yes	No	Mammals
Eagle	No	Yes	No	Yes	Non-mammals

- Is the animal with the attribute values (Give Birth = Yes, Can Fly = No, Live in Water = Yes, Have Legs = No) a mammal?

K-Nearest Neighbors

- Given a dataset of the speed and agility ratings for 20 athletes and whether they were drafted by a professional team.

ID	Speed	Agility	Draft
1	2.50	6.00	No
2	3.75	8.00	No
3	2.25	5.50	No
4	3.25	8.25	No
5	2.75	7.50	No
6	4.50	5.00	No
7	3.50	5.25	No
8	3.00	3.25	No
9	4.00	4.00	No
10	4.25	3.75	No

ID	Speed	Agility	Draft
11	2.00	2.00	No
12	5.00	2.50	No
13	8.25	8.50	No
14	5.75	8.75	Yes
15	4.75	6.25	Yes
16	5.50	6.75	Yes
17	5.25	9.50	Yes
18	7.00	4.25	Yes
19	7.50	8.00	Yes
20	7.25	5.75	Yes

- Suppose an athlete with speed $= 6.75$ and agility $= 3.00$, classify him into one of the two classes (Draft $=$ Yes, Draft $=$ No) using KNN with Euclidean distance metric and $K = 3$.

K-Means Clustering

- Consider 4 data points A, B, C and D as follows:

	x_1	x_2
A	2	3
B	6	1
C	1	2
D	3	0

- Form two clusters for the above datapoints by picking two initial centroids, $c_1 = (4, 2)$ and $c_2 = (2, 1)$.
- Assume Euclidean distance is used as the metric. Show all the calculation steps and the final cluster assignments for the 4 data points.

Perceptron

- Suppose we have the following data points:
  - $\mathbf{x}$ = (1, -2), T = 1
  - $\mathbf{x}$ = (0, -1), T = 0
- Train a perceptron with the initial weights ($w_1 = 0$, $w_2 = -2$), zero bias, learning rate $\eta = 0.5$, and a unit-step activation function:

$$f(x) = \begin{cases} 1 & \textit{if } x > 0 \\ 0 & \textit{otherwise} \end{cases}$$

  until it converges.
- Show all the steps, i.e. the change of weights and bias in each iteration.

Multilayer Perceptron

- Given a multilayer perceptron with two inputs x_1, x_2, one hidden unit and one output unit. Both the hidden unit and output use sigmoid activation function. Altogether, the network has 3 weights, w_1, w_2, w_3, and 2 biases, θ_1, θ_2.
- All weights are initialized with 0.1, and all the biases are initialized with -0.1.
- Use sigmoid as the activation function for all units, i.e.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

- Let the training set be as follows:

x1	x2	T
1	0	1
0	1	0

Determine the weights after the first epoch two iterations of the backpropagation algorithm, given a learning rate of $\eta = 0.3$.

Suggested Solutions

Naïve Bayes 1

- Let C be the a patient having a cold, F be a patient having a fever
- According to the question, we have:
  - $P(F|C) = 0.5$
  - $P(C) = 1/50000$
  - $P(F) = 1/20$
- Calculation:

$$\begin{aligned} P(C|F) &= \frac{P(F|C)P(C)}{P(F)} \\ &= \frac{0.5 \times (1/50000)}{1/20} \\ &= 0.0002 \end{aligned}$$

Naïve Bayes 2

- Let GB be "Give Birth", CF be "Can Fly", LIW be "Live in Water", HL be "Have Legs", M be "Mammals", NM be "Non-mammals".
- Apply Naïve Bayes, we have:

$$
\begin{aligned}
&P(GB = Yes, CF = No, LIW = Yes, HL = No|M)P(M)\\
=&P(GB = Yes|M)P(CF = No|M)P(LIW = Yes|M)P(HL = No|M)P(M)\\
=&\frac{6}{7} \times \frac{6}{7} \times \frac{2}{7} \times \frac{2}{7} \times \frac{7}{20} = 0.021
\end{aligned}
$$

$$
\begin{aligned}
&P(GB = Yes, CF = No, LIW = Yes, HL = No|N)P(N)\\
=&P(GB = Yes|N)P(CF = No|N)P(LIW = Yes|N)P(HL = No|N)P(N)\\
=&\frac{1}{13} \times \frac{10}{13} \times \frac{3}{13} \times \frac{4}{13} \times \frac{13}{20} = 0.0027
\end{aligned}
$$

- As $P(GB = Yes, CF = No, LIW = Yes, HL = No|M)P(M) > P(GB = Yes, CF = No, LIW = Yes, HL = No|N)P(N)$, it is mammals.

K-Nearest Neighbors

- Compute the Euclidean distance between each training data point and the test point, and find the 3-nearest neighbors.

ID	Speed	Agility	Draft	Speed (Test)	Agility (Test)	Distance
1	2.5	6	No	6.75	3	5.202163
2	3.75	8	No	6.75	3	5.830952
3	2.25	5.5	No	6.75	3	5.147815
4	3.25	8.25	No	6.75	3	6.309715
5	2.75	7.5	No	6.75	3	6.020797
6	4.5	5	No	6.75	3	3.010399
7	3.5	5.25	No	6.75	3	3.952847
8	3	3.25	No	6.75	3	3.758324
9	4	4	No	6.75	3	2.926175
10	4.25	3.75	No	6.75	3	2.610077
11	2	2	No	6.75	3	4.854122
12	5	2.5	No	6.75	3	1.820027
13	8.25	8.5	No	6.75	3	5.700877
14	5.75	8.75	Yes	6.75	3	5.836309
15	4.75	6.25	Yes	6.75	3	3.816084
16	5.5	6.75	Yes	6.75	3	3.952847
17	5.25	9.5	Yes	6.75	3	6.670832
18	7	4.25	Yes	6.75	3	1.274755
19	7.5	8	Yes	6.75	3	5.055937
20	7.25	5.75	Yes	6.75	3	2.795085

- Among the 3-nearest neighbors, 2 of them with "Draft = No" and 1 with "Draft = Yes". So, based on majority voting, we classify the test point as "Draft = No".

K-Means Clustering

- Find the distances between each data point with the 2 centroids $c_1 = (4, 2)$ and $c_2 = (2, 1)$:

Data Point	A	B	C	D
x1	2	6	1	3
x2	3	1	2	0
DC1	2.236068	2.236068	3	2.236068
DC2	2	4	1.414214	1.414214
Cluster	2	1	2	2

- Re-compute the centroids using the current cluster memberships
  - New 1st centroid:

$$x_1 = 6$$
$$x_2 = 1$$

  - New 2nd centroid:

$$x_1 = (2 + 1 + 3)/3 = 2$$
$$x_2 = (3 + 2 + 0)/3 = 1.66667$$

K-Means Clustering

- Find the distances between each data point with the 2 centroids $c_1 = (6, 1)$ and $c_2 = (2, 1.666667)$:

Data Point	A	B	C	D
x1	2	6	1	3
x2	3	1	2	0
DC1	4.472136	0	5.09902	3.162278
DC2	1.333333	4.055175	1.054092	1.943651
Cluster	2	1	2	2

- As the cluster memberships remain the same, the cluster centers also remain the same.
  - New 1st centroid:

$$x_1 = 6$$
$$x_2 = 1$$

  - New 2nd centroid:

$$x_1 = 2$$
$$x_2 = 1.66667$$

- Also, the algorithm converges.

Perceptron

x_1	x_2	T	O	Δw_1	w_1	Δw_2	w_2	$\Delta\theta$	θ
-	-	-	-	-	0	-	-2	-	0
1	-2	1	1	0	0	0	-2	0	0
0	-1	0	1	0	0	0.5	-1.5	-0.5	-0.5
1	-2	1	1	0	0	0	-1.5	0	-0.5
0	-1	0	1	0	0	0.5	-1	-0.5	-1
1	-2	1	1	0	0	0	-1	0	-1
0	-1	0	0	0	0	0	-1	0	-1

Multilayer Perceptron - Round 1 - Step 1, Forward Propagation

- Inputs: $x_1 = 1, x_2 = 0$
- Actual Output: $T = 1$
- Weights: $w_1 = 0.1,\ w_2 = 0.1,\ w_3 = 0.1$
- Biases: $\theta_1 = -0.1,\ \theta_2 = -0.1.$
- Calculations:
  - $\sum_1 = x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot (0.1) + 0 \cdot (0.1) = 0.1$
    Output (O_j): $f(\sum_1 + \theta_1) = f(0.1 - 0.1) = 0.5$
  - $\sum_2 = O_j \cdot w_3 = 0.5 \cdot (0.1) = 0.05$
    Output (O_k): $f(\sum_2 + \theta_2) = f(0.05 - 0.1) = 0.487503$

Multilayer Perceptron - Round 1 - Step 1, Backward Propagation

- Calculations:
  - Output (O_j): $f(\sum_1 + \theta_1) = f(0.1 - 0.1) = 0.5$
  - Output (O_k): $f(\sum_2 + \theta_2) = f(0.05 - 0.1) = 0.487503$
  - $\delta_k = (O_k - T_k)O_k(1 - O_k) = (0.487503 - 1)(0.487503)(1 - 0.487503) = -0.128044$
  - New w_3 = Old w_3 - $\eta \delta_k O_j = 0.1 - 0.3(-0.128044)(0.5) = 0.119207$
  - New θ_2 = Old $\theta_2 - \eta \delta_k = -0.1 - (0.3)(-0.128044) = -0.061587$
  - $\delta_j = O_j(1 - O_j)\delta_k w_{jk} = 0.5(1 - 0.5)(-0.128044)(0.1) = -0.003201$
  - New w_1 = Old $w_1 - \eta \delta_j x_1 = 0.1 - (0.3)(-0.003201)(1) = 0.100960$
  - New w_2 = Old $w_2 - \eta \delta_j x_2 = 0.1 - (0.3)(-0.003201)(0) = 0.1$
  - New θ_1 = Old $\theta_1 - \eta \delta_j = -0.1 - (0.3)(-0.003201) = -0.099040$

Multilayer Perceptron - Round 1 - Step 2, Forward Propagation

- Inputs: $x_1 = 0, x_2 = 1$
- Actual Output: $T = 0$
- Weights: $w_1 = 0.100960$, $w_2 = 0.1$, $w_3 = 0.119207$
- Biases: $\theta_1 = -0.099040$, $\theta_2 = -0.061587$.
- Calculations:
  - $\sum_1 = x_1 \cdot w_1 + x_2 \cdot w_2 = 0 \cdot (0.100960) + 1 \cdot (0.1) = 0.1$
    Output (O_j): $f(\sum_1 + \theta_1) = f(0.1 - 0.099040) = 0.50024$
  - $\sum_2 = O_j \cdot w_3 = 0.50024 \cdot (0.119207) = 0.059632$
    Output (O_k): $f(\sum_2 + \theta_2) = f(0.059632 - 0.061587) = 0.499511$

Multilayer Perceptron - Round 1 - Step 2, Backward Propagation

- Calculations:
  - Output (O_j): $f(\sum_1 + \theta_1) = f(0.1 - 0.099040) = 0.50024$
  - Output (O_k): $f(\sum_2 + \theta_2) = f(0.059632 - 0.061587) = 0.499511$
  - $\delta_k = (O_k - T_k)O_k(1 - O_k) = (0.499511 - 0)(0.499511)(1 - 0.499511) = 0.124878$
  - New w_3 = Old w_3 - $\eta\delta_k O_j = 0.119207 - 0.3(0.124878)(0.50024) = 0.100466$
  - New θ_2 = Old $\theta_2 - \eta\delta_k = -0.061587 - (0.3)(0.124878) = -0.09905$
  - $\delta_j = O_j(1 - O_j)\delta_k w_{jk} = 0.50024(1 - 0.50024)(0.124878)(0.119207) = 0.003722$
  - New w_1 = Old $w_1 - \eta\delta_j x_1 = 0.100960 - (0.3)(0.003722)(0) = 0.100960$
  - New w_2 = Old $w_2 - \eta\delta_j x_2 = 0.1 - (0.3)(0.003722)(1) = 0.098883$
  - New θ_1 = Old $\theta_1 - \eta\delta_j = -0.099040 - (0.3)(0.003722) = -0.100157$