Supervised Learning (KNN)

Muhammad Bilal Shahnawaz

bilal.shahnawaz@numl.edu.pk

Outline

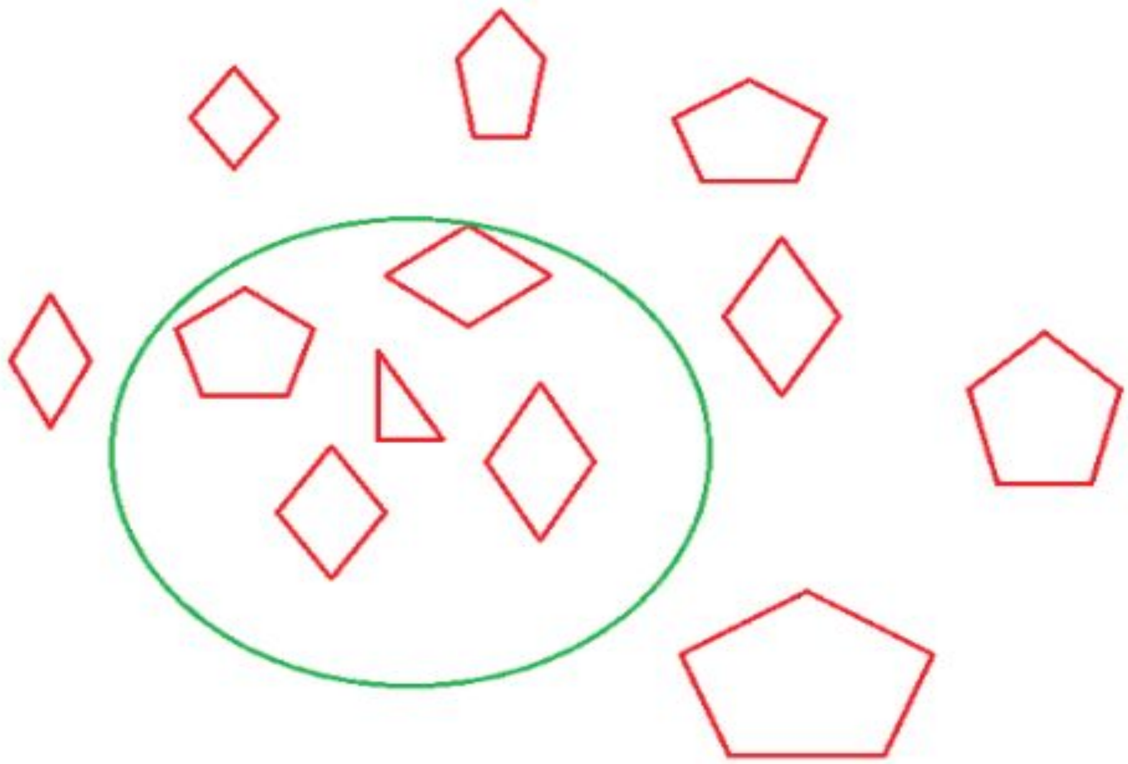
- Last Class
  - ML Life cycle
  - Performance Measure Parameters
    - *Classification Accuracy (mostly used)*
    - *Confusion Matrix*
    - *Area under Curve*
    - *Mean Absolute Error*
    - *Mean Squared Error*
  - Common ML Problems
- Today
  - KNN

KNN

- One of the simplest ML algorithms based on Supervised Learning technique.
- At the training phase, K-NN algorithm stores all the available data (dataset)
- When a new case/data point appears, it classifies the new data point based on the similarity
  - Assigns the category from the available categories that is most similar among from the nearest neighbors.
- **a lazy learner algorithm**

Contd..

- K in KNN represents the number of the nearest neighbors used to classify new data points.
- Different k values can and will produce different classifications
- K=4



KNN algorithm

- The K-NN working can be explained on the basis of the below algorithm:
  - **Step-1:** Determine parameter k= number of nearest neighbors.
  - **Step-2:** Calculate the distance between the query instance (new data point) and all the training samples.
  - **Step-3:** Take the K nearest neighbors as per the calculated distance. (Sort the distance and determine the nearest neighbors)
  - **Step-4:** Gather the category of the nearest neighbors
  - **Step-5:** Use simple majority of the category of nearest neighbors as the prediction value of the query instance.

1. Selection of K value

- Choosing the optimal value for K is best done by first inspecting the data.
- Choosing the right value of K is called parameter tuning and it's necessary for better results
- In general, a large K value is more precise as it reduces the overall noise but there is no guarantee.
  - Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN.
  - The most preferred value for K is 5.

KNN – Number of Neighbors

- If K=1, select the nearest neighbour
- If K>1,
  - For classification select the most frequent neighbour.
  - For regression calculate the average of K neighbours.

- **Rule of thumb is k < sqrt(n),** n is number of examples
  - Odd value of K is always selected to avoid confusion between 2 classes.

2. Distance Calculation

- There are various methods for calculating the distance between the new point and each training point
- The most commonly known methods are:
  - Euclidian (for continuous)
  - Manhattan (for continuous)

  - Hamming distance (for categorical)

Distance functions

Euclidean

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Manhattan

$$\sum_{i=1}^{k}\left|x_i - y_i\right|$$

Minkowski

$$\left(\sum_{i=1}^{k}(\left|x_i - y_i\right|)^q\right)^{1/q}$$

Hamming Distance

- It is used for categorical variables
- If the value (x) and the value (y) are the same, the distance D will be equal to 0 . Otherwise D=1.

$$D_H = \sum_{i=1}^{k} |x_i - y_i|$$

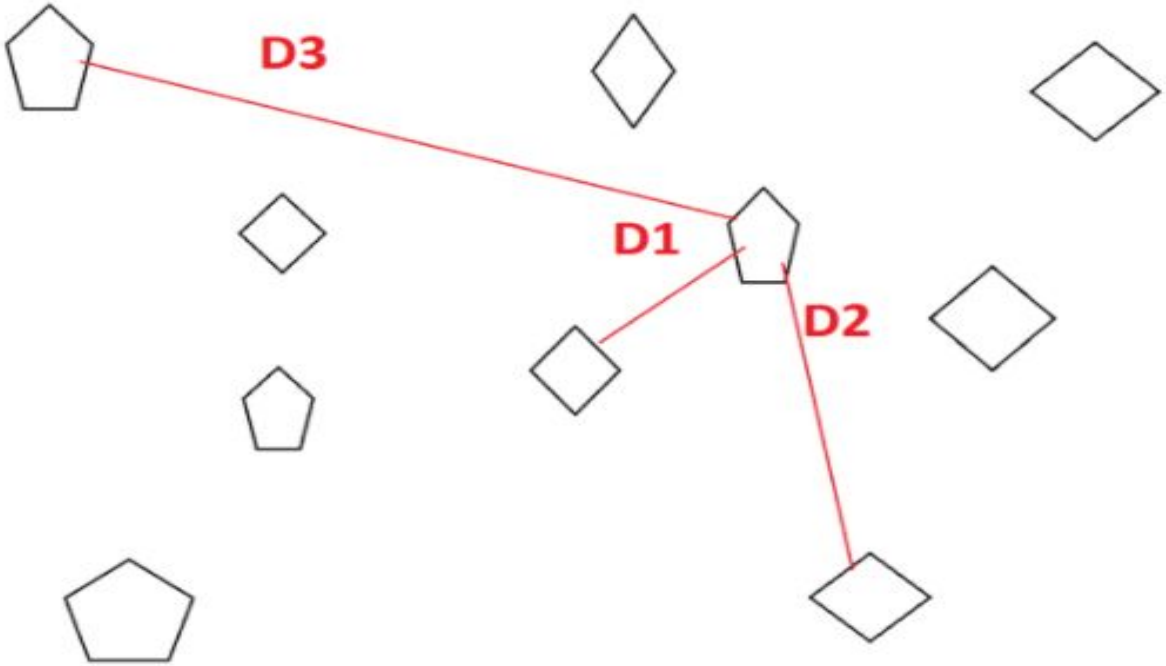
$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

Contd..

- We usually use **Euclidean distance** to calculate the nearest neighbor.
- If we have two points (x, y) and (a, b). The formula for Euclidean distance (d) will be

$$d = \sqrt{(x-a)^2 + (y-b)^2}$$



KNN Classification – Distance

Age	Loan	Default	Distance
25	$40,000	N	102000
35	$60,000	N	82000
45	$80,000	N	62000
20	$20,000	N	122000
35	$120,000	N	22000
52	$18,000	N	124000
23	$95,000	Y	47000
40	$62,000	Y	80000
60	$100,000	Y	42000
48	$220,000	Y	78000
33	$150,000	Y	8000
48	**$142,000**	**?**	

Euclidean Distance

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Distance scaling

- One major drawback in calculating distance measures directly from the training set is in the case where variables have different measurement scales or there is a mixture of numerical and categorical variables.
- For example, if one variable is based on annual income in dollars, and the other is based on age in years.
- Then income will have a much higher influence on the distance calculated. One solution is to standardize the training set as shown below

Normalized Distance

Age	Loan	Default	Distance
0.125	0.11	N	0.7652
0.375	0.21	N	0.5200
0.625	0.31	N	0.3160
0	0.01	N	0.9245
0.375	0.50	N	0.3428
0.8	0.00	N	0.6220
0.075	0.38	Y	0.6669
0.5	0.22	Y	0.4437
1	0.41	Y	0.3650
0.7	1.00	Y	0.3861
0.325	0.65	Y	0.3771
0.7	**0.61**	**?**	

Normalized Variable

$$X_s = \frac{X - Min}{Max - Min}$$

Nearest Neighbor Complexity

- Expensive for high dimensional data (d>20?)
- O(Knd) complexity for both storage and query time
  - n is the number of training examples,
  - d is the dimension of each sample
    - O(d) to compute distance to one example.
    - O(nd) to compute distance to n numbers of examples
    - O(knd) to find k closest examples
    - Thus total complexity is O(knd).

Advantages/Disadvantages

- Advantages:
  - Simple to implement
  - Training is very fast
  - Learns very quickly;
  - Provides good generalization accuracy
  - Robust to noisy training data;
  - Don't lose information
- Disadvantages:
  - Slow at query
  - Always needs to determine the value of K which may be complex some time.
  - The computation cost may be high because of calculating the distance between the data points for all the training samples.
    - Expensive, High Storage Requirements

When is KNN?

- Data is **properly labeled**.
  - For example, if we are predicting someone is having diabetes or not the final label can be 1 or 0. It cannot be NaN or -1.
- Data is **noise-free**.
  - For the diabetes data set we cannot have a Glucose level as 0 or 10000. It's practically impossible.
- Comparatively Small dataset.

Example

- We have data from the questionnaires survey (to ask people opinion) and objective testing with two attributes (acid durability and strength) to classify whether a special paper tissue is good or not. Here are four training samples:

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Y = Classification
7	7	Bad
7	4	Bad
3	4	Good
1	4	Good

Contd..

- **Question:** Now the factory produces a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7. Without another expensive survey, can we guess what the classification of this new tissue is?

Solution

- 1.Determine parameter K
  - Suppose k=3
- 2.Calculate the distance between the query-instance and all the training sample

7	7	
7	4	
3	4	
1	4	

Contd..

- 3. Sort the distance and determine nearest neighbors based on the K-th minimum distance

X1= Acid Durability (seconds)	X2=Strength (Kg/square meter)	Distance to query instance (3,7)	Rank minimum distance	Is it included in 3-Nearest neighbors?
7	7		3	Yes
7	4		4	No
3	4		1	Yes
1	4		2	Yes

Contd..

- 4. Gather the category of the nearest neighbors
  - Notice in the second row last column that the category of nearest neighbor (Y) is not included because the rank of this data is more than 3 (=K).

X1= Acid Durability (seconds)	X2=Strength (Kg/square meter)	Distance to query instance (3,7)	Rank minimum distance	Is it included in 3-Nearest neighbors?	Category of nearest Neighbors
7	7		3	Yes	Bad
7	4		4	No	---
3	4		1	Yes	Good
1	4		2	Yes	Good

Contd..

- 5. Use simple majority of the category of nearest neighbors as the prediction value of the query instance
  - We have 2 good and 1 bad, since 2>1 then we conclude that a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7 is **included in Good category**

Example 2

- In order to select the best candidates, an over-subscribed secondary school sets an entrance exam on two subjects of English and Mathematics. Suppose that we know the marks and the classification results of 5 applicants as in the Table below. If an applicant has been accepted, this is denoted as class 1,otherwise class 2. Use the nearest neighbor rule to determine if Andy should be accepted if his marks of **English and Mathematics are 70 and 70** respectively.

Contd..

- .

Candidate No.	English	Math	Class
1	80	85	1
2	70	60	2
3	50	70	2
4	90	70	1
5	85	75	1

KNN in Regression

- For regression calculate the average of K neighbours.

ID	Height	Age	Weight
1	5	45	77
2	5.11	26	47
3	5.6	30	55
4	5.9	34	59
5	4.8	40	72
6	5.8	36	60
7	5.3	19	40
8	5.8	28	60
9	5.5	23	45
10	5.6	32	58
11	5.5	38	?