Machine Learning

Similarity Functions
K Nearest Neighbors (KNN)

K Nearest Neighbors

- K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions).
- KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique.

KNN –different names

- K-Nearest Neighbors
- Memory-Based Reasoning
- Example-Based Reasoning
- Instance-Based Learning
- Case-Based Reasoning
- Lazy Learning

KNN Algorithm

- A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function.
- If K = 1, then the case is simply assigned to the class of its nearest neighbor.

Distance Functions

Distance functions

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

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

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

- It should also be noted that all three distance measures are only valid for continuous variables.
- In the instance of categorical variables the Hamming distance must be used.
- It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

Hamming distance

Hamming Distance

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

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

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

X	Y	Distance
Male	Male	0
Male	Female	1

Similarity –Hamming Distance

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

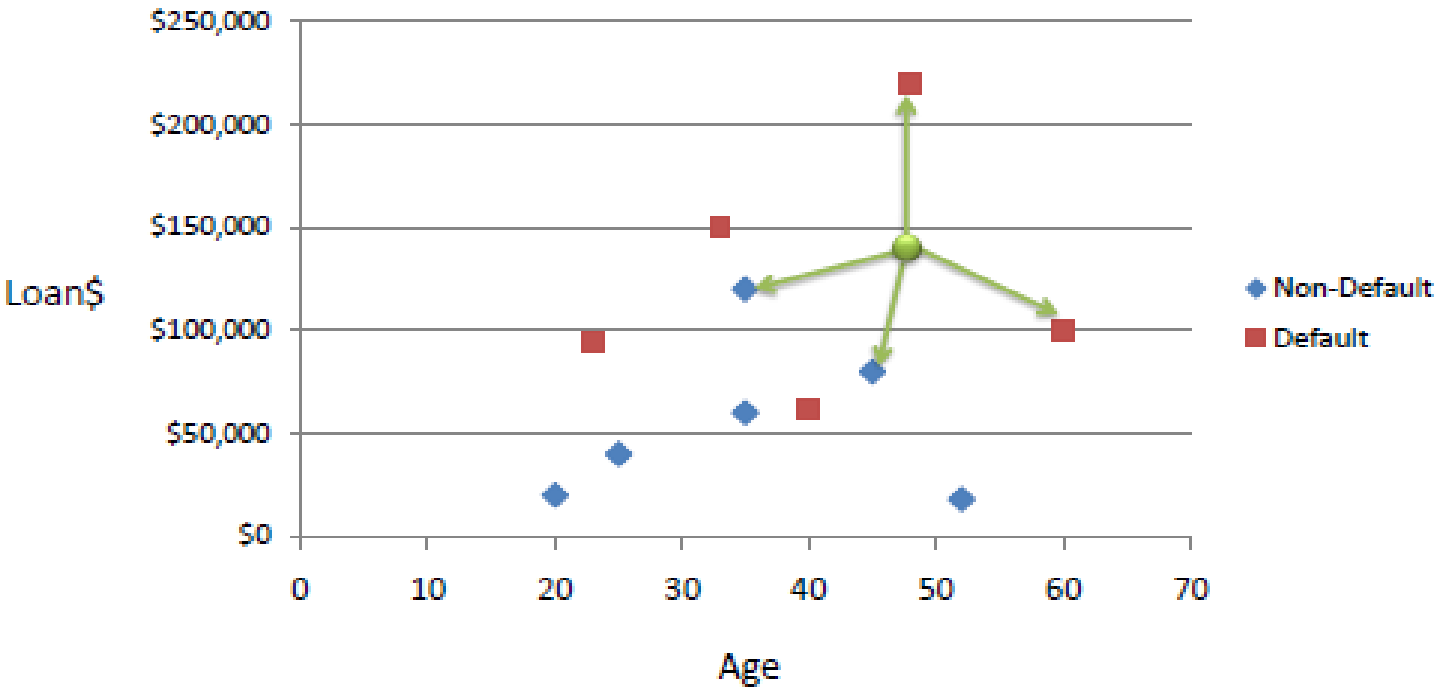
Gene 1	A	A	T	C	C	A	G	T
Gene 2	T	C	T	C	A	A	G	C
Hamming Distance	1	1	0	0	1	0	0	1

Optimal K-value

- Choosing the optimal value for K is best done by first inspecting the data.
- In general, a large K value is more precise as it reduces the overall noise but there is no guarantee.
- Cross-validation is another way to retrospectively determine a good K value by using an independent dataset to validate the K value.
- Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN.

Example

- Consider the following data concerning credit default. Age and Loan are two numerical variables (predictors) and Default is the target.



Con…

- We can now use the training set to classify an unknown case (Age=48 and Loan=$142,000) using Euclidean distance.

- If K=1 then the nearest neighbor is the last case in the training set with Default=Y.

Con…

- D = Sqrt[(48-33)^2 + (142000-150000)^2] = 8000.01 >> Default=Y

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	2
52	$18,000	N	124000	
23	$95,000	Y	47000	
40	$62,000	Y	80000	
60	$100,000	Y	42000	3
48	$220,000	Y	78000	
33	$150,000	Y	8000	1
48	**$142,000**	**?**		

Euclidean Distance

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

- With K=3, there are two Default=Y and one Default=N out of three closest neighbors. The prediction for the unknown case is again Default=Y.

Standardized Distance

- 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:

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**	**?**	

Standardized Variable

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

Using the standardized distance on the same training set, the unknown case returned a different neighbor which is not a good sign of robustness.

KNN Regression - Distance

Age	Loan	House Price Index	Distance
25	$40,000	135	102000
35	$60,000	256	82000
45	$80,000	231	62000
20	$20,000	267	122000
35	$120,000	139	22000
52	$18,000	150	124000
23	$95,000	127	47000
40	$62,000	216	80000
60	$100,000	139	42000
48	$220,000	250	78000
33	$150,000	264	8000
48	**$142,000**	**?**	

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

KNN Regression – Standardized Distance

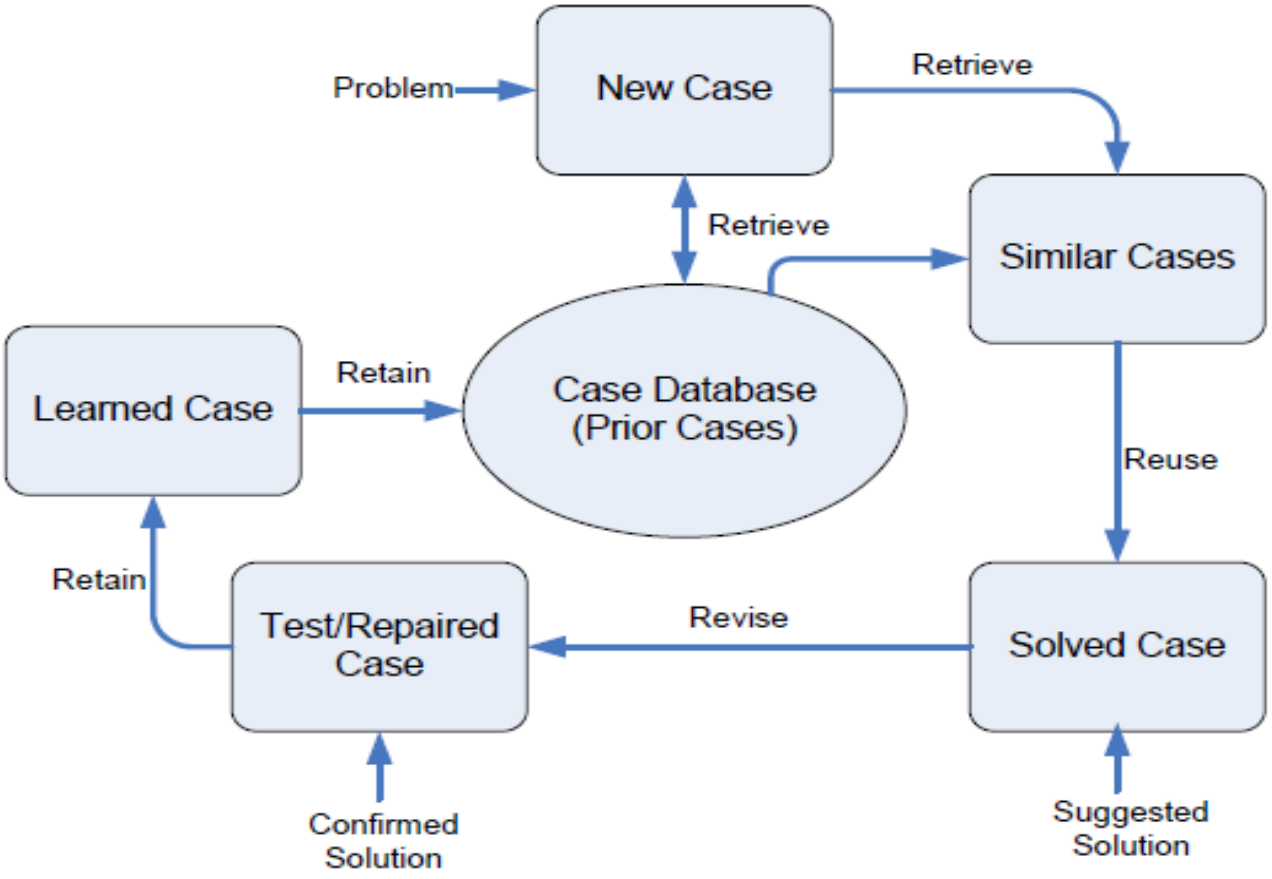
Age	Loan	House Price Index	Distance
0.125	0.11	135	0.7652
0.375	0.21	256	0.5200
0.625	0.31	231	0.3160
0	0.01	267	0.9245
0.375	0.50	139	0.3428
0.8	0.00	150	0.6220
0.075	0.38	127	0.6669
0.5	0.22	216	0.4437
1	0.41	139	0.3650
0.7	1.00	250	0.3861
0.325	0.65	264	0.3771
0.7	**0.61**	**?**	

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

Instance Based Reasoning

- IB1is based on the standard KNN
- IB2 is incremental KNN learner that only incorporates misclassified instances into the classifier.
- IB3discards instances that do not perform well by keeping success records.

Case Based Reasoning



Summary

- KNN is conceptually simple, yet able to solve complex problems
- Can work with relatively little information
- Learning is simple (no learning at all!)
- Memory and CPU cost
- Feature selection problem
- Sensitive to representation