K Nearest Neighbors - Regression

K nearest neighbors is a simple algorithm that stores all available cases and predict the numerical target 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.

Algorithm

A simple implementation of KNN regression is to calculate the average of the numerical target of the K nearest neighbors. Another approach uses an inverse distance weighted average of the K nearest neighbors. KNN regression uses the same distance functions as KNN classification.

Distance functions

Euclidean

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

Manhattan

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

Minkowski

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

The above three distance measures are only valid for continuous variables. In the case of categorical variables you must use the Hamming distance, which is a measure of the number of instances in which corresponding symbols are different in two strings of equal length.

Hamming Distance

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

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

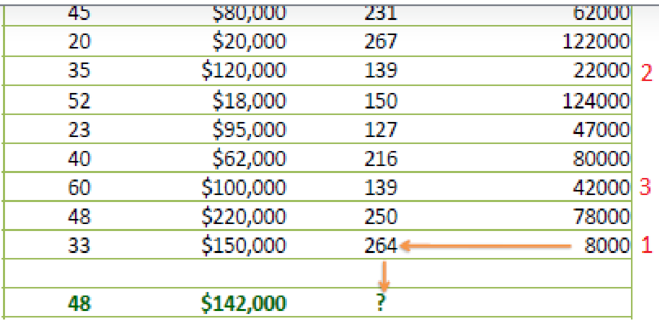
$$x \neq y \Rightarrow D = 1$$

X	Y	Distance
Male	Male	0
Male	Female	1

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; however, the compromise is that the distinct boundaries within the feature space are blurred. Cross-validation is another way to retrospectively determine a good K value by using an independent data set to validate your K value. The optimal K for most datasets is 10 or more. That produces much better results than 1-NN.

Example:

Consider the following data concerning House Price Index or HPI. Age and Loan are two numerical variables (predictors) and HPI is the numerical target.



45	$80,000	231	62000	
20	$20,000	267	122000	
35	$120,000	139	22000	2
52	$18,000	150	124000	
23	$95,000	127	47000	
40	$62,000	216	80000	
60	$100,000	139	42000	3
48	$220,000	250	78000	
33	$150,000	264	8000	1
48	$142,000	?		

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

We can now use the training set to classify an unknown case (Age=33 and Loan=$150,000) using Euclidean distance. If K=1 then the nearest neighbor is the last case in the training set with HPI=264.

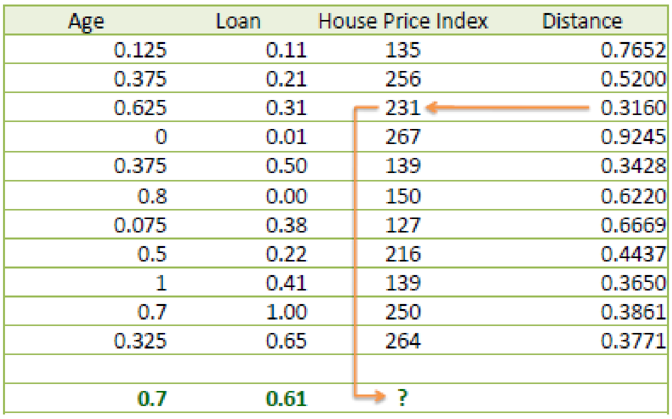
D = Sqrt[(48-33)^2 + (142000-150000)^2] = 8000.01 >> HPI = 264

By having K=3, the prediction for HPI is equal to the average of HPI for the top three neighbors.

HPI = (264+139+139)/3 = 180.7

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 below.



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	?	

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

As mentioned in KNN Classification using the standardized distance on the same training set, the unknown case returned a different neighbor which is not a good sign of robustness.

Exercise