K-Nearest Neighbors Algorithm

Instance-based learning

In instance-based learning (lazy learning) the training examples are stored verbatim, and a distance function is used to determine which member of the training set is closest to an unknown test instance

Once the nearest training instance has been located, its class is predicted for the test instance

The only remaining problem is defining the distance function

Distance Function

Euclidean Distance

instance 1 $\quad P_1 = a_1^{(1)}, a_2^{(1)}, \ldots, a_k^{(1)}$

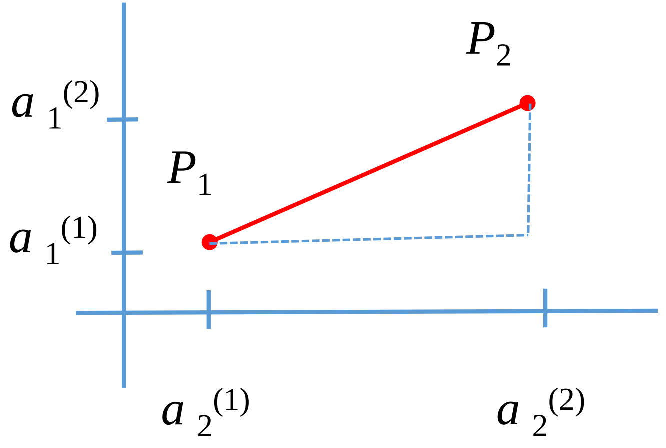
instance 2 $\quad P_2 = a_1^{(2)}, a_2^{(2)}, \ldots, a_k^{(2)}$

k is the number of attributes

$$\sqrt{(a_1^{(1)} - a_1^{(2)})^2 + (a_2^{(1)} - a_2^{(2)})^2 + \cdots + (a_k^{(1)} - a_k^{(2)})^2} = \sqrt{\sum_{i=1}^{k} (a_i^{(1)} - a_i^{(2)})^2}$$

If k=2

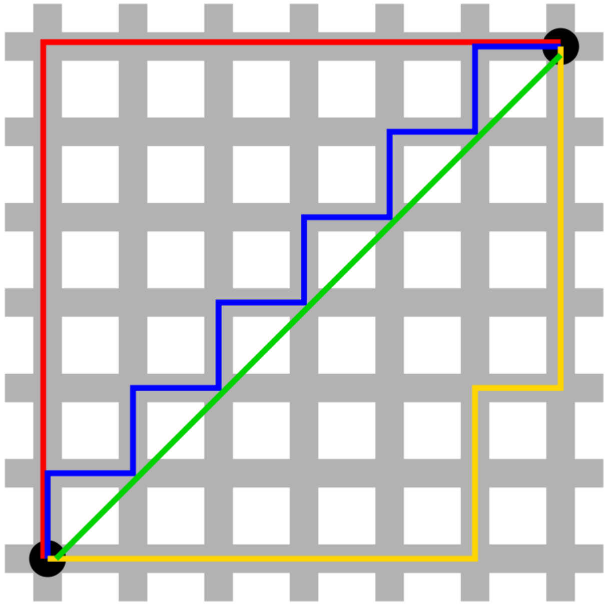
$$\sqrt{(a_1^{(1)} - a_1^{(2)})^2 + (a_2^{(1)} - a_2^{(2)})^2}$$



There are other possible distances

Manhattan Distance

$$\sum_{i=1}^{k} \left| a_i^{(1)} - a_i^{(2)} \right|$$



Minkowski Distance

$$\left(\sum_{i=1}^{k} \left(\left| a_i^{(1)} - a_i^{(2)} \right| \right)^{q} \right)^{\frac{1}{q}}$$

In the instance of categorical variables the Hamming distance must be used

Hamming Distance

$$D_H = \sum_{i=1}^{k} \left| a_i^{(1)} - a_i^{(2)} \right|$$

$$a^{(1)} = a^{(2)} \Rightarrow D = 0$$

$$a^{(1)} \neq a^{(2)} \Rightarrow D = 1$$

Example

a_1	a_2	*Distance*
Male	Male	0
Male	Female	1

K Nearest Neighbors - Classification

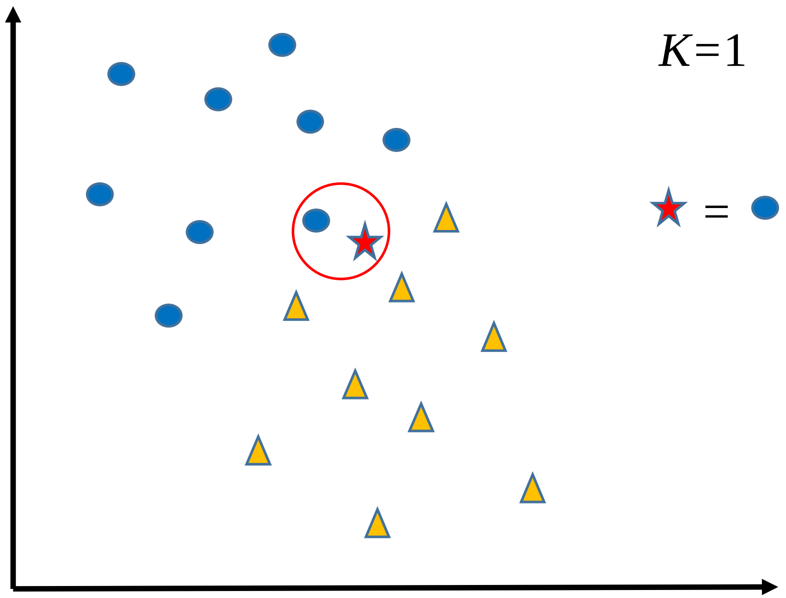
It is a supervised learning algorithm

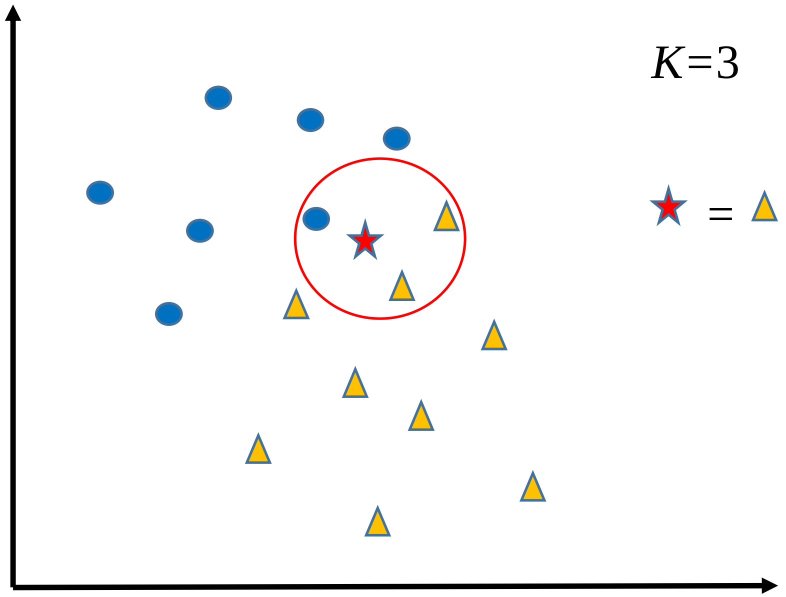
K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure

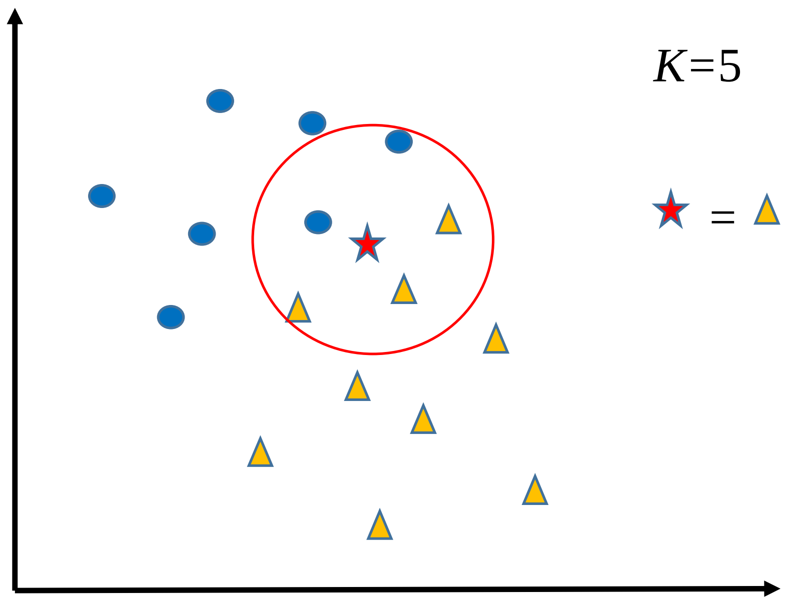
Algorithm

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.


K=1
=


K=3
=


K=5
=

Example

New Instance = (Age = 48, Loan=$142,000)

$$D = \sqrt{\sum_{i=1}^{k} (a_i^{(1)} - a_i^{(2)})^2}$$

$$D = \sqrt{(Age_i - Age)^2 + (Loan_i - Loan)^2}$$

Diference Age	Age	Loan	Default	Diference Loan	D
-23	25	$40,000	N	-$102,000	102000.00
-13	35	$60,000	N	-$82,000	82000.00
-3	45	$80,000	N	-$62,000	62000.00
-28	20	$20,000	N	-$122,000	122000.00
-13	35	$120,000	N	-$22,000	22000.00
4	52	$18,000	N	-$124,000	124000.00
-25	23	$95,000	Y	-$47,000	47000.01
-8	40	$62,000	Y	-$80,000	80000.00
12	60	$100,000	Y	-$42,000	42000.00
0	48	$220,000	Y	$78,000	78000.00
-15	33	$150,000	Y	$8,000	8000.01

K= 1

New Instance = (Age = 48, Loan=$142,000, **Default = Y**)

UPAEP

Example

New Instance = (Age = 48, Loan=$142,000)

$$D = \sqrt{\sum_{i=1}^{k} (a_i^{(1)} - a_i^{(2)})^2}$$

$$D = \sqrt{(Age_i - Age)^2 + (Loan_i - Loan)^2}$$

Diference Age	Age	Loan	Default	Diference Loan	D	
-23	25	$40,000	N	-$102,000	102000.00	
-13	35	$60,000	N	-$82,000	82000.00	
-3	45	$80,000	N	-$62,000	62000.00	
-28	20	$20,000	N	-$122,000	122000.00	
-13	35	$120,000	N	-$22,000	22000.00	2
4	52	$18,000	N	-$124,000	124000.00	
-25	23	$95,000	Y	-$47,000	47000.01	
-8	40	$62,000	Y	-$80,000	80000.00	
12	60	$100,000	Y	-$42,000	42000.00	3
0	48	$220,000	Y	$78,000	78000.00	
-15	33	$150,000	Y	$8,000	8000.01	1

K= 3

New Instance = (Age = 48, Loan=$142,000, **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.

Age	Loan	Default
25	$40,000	N
35	$60,000	N
45	$80,000	N
20	$20,000	N
35	$120,000	N
52	$18,000	N
23	$95,000	Y
40	$62,000	Y
60	$100,000	Y
48	$220,000	Y
33	$150,000	Y

$$X_s = \frac{X - Min_{a_i}}{Max_{a_i} - Min_{a_i}}$$

$$\frac{25-20}{60-20} = \frac{5}{40} = 0.125$$

$$\frac{40{,}000-18{,}000}{220{,}000-18{,}000} = \frac{22{,}000}{202{,}000} = 0.10891$$

Age	Loan	Default
0.125	0.11	N
0.375	0.21	N
0.625	0.31	N
0	0.01	N
0.375	0.50	N
0.8	0.00	N
0.075	0.38	Y
0.5	0.22	Y
1	0.41	Y
0.7	1.00	Y
0.325	0.65	Y

New Instance = (Age = 48, Loan=$142,000)

New Instance = (Age = 0.7, Loan=0.61)

Diferenc e Age	Age	Loan	Default	Diference Loan	D
-0.575	0.125	0.11	N	-0.50	0.76
-0.325	0.375	0.21	N	-0.40	0.52
-0.075	0.625	0.31	N	-0.30	0.31
-0.7	0	0.01	N	-0.60	0.92
-0.325	0.375	0.50	N	-0.11	0.34
0.1	0.8	0.00	N	-0.61	0.62
-0.625	0.075	0.38	Y	-0.23	0.67
-0.2	0.5	0.22	Y	-0.39	0.44
0.3	1	0.41	Y	-0.20	0.36
0	0.7	1.00	Y	0.39	0.39
-0.375	0.325	0.65	Y	0.04	0.38

K= 1

New Instance = (Age = 48, Loan=$142,000, **Default = N**)

New Instance = (Age = 48, Loan=$142,000)

New Instance = (Age = 0.7, Loan=0.61)

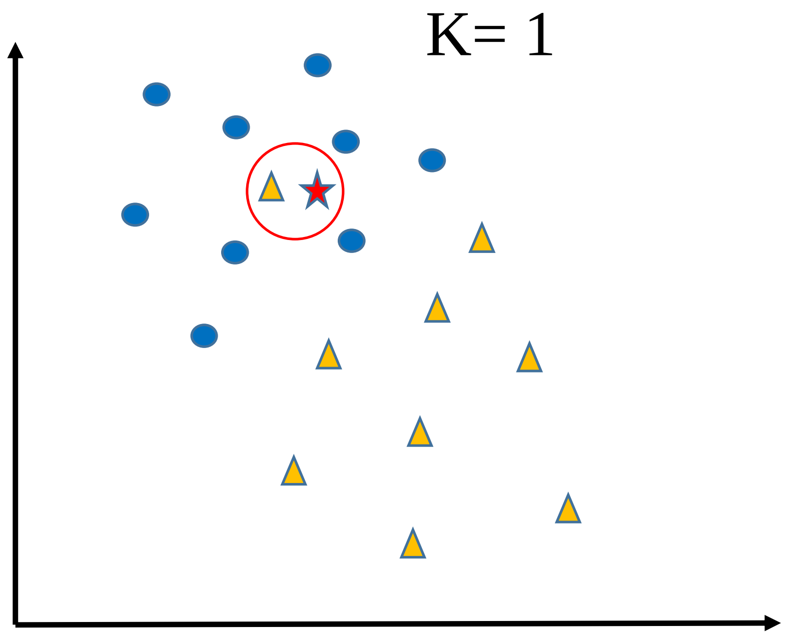
Diference Age	Age	Loan	Default	Diference Loan	D	
-0.575	0.125	0.11	N	-0.50	0.76	
-0.325	0.375	0.21	N	-0.40	0.52	
-0.075	0.625	0.31	N	-0.30	0.31	1
-0.7	0	0.01	N	-0.60	0.92	
-0.325	0.375	0.50	N	-0.11	0.34	2
0.1	0.8	0.00	N	-0.61	0.62	
-0.625	0.075	0.38	Y	-0.23	0.67	
-0.2	0.5	0.22	Y	-0.39	0.44	
0.3	1	0.41	Y	-0.20	0.36	3
0	0.7	1.00	Y	0.39	0.39	
-0.375	0.325	0.65	Y	0.04	0.38	

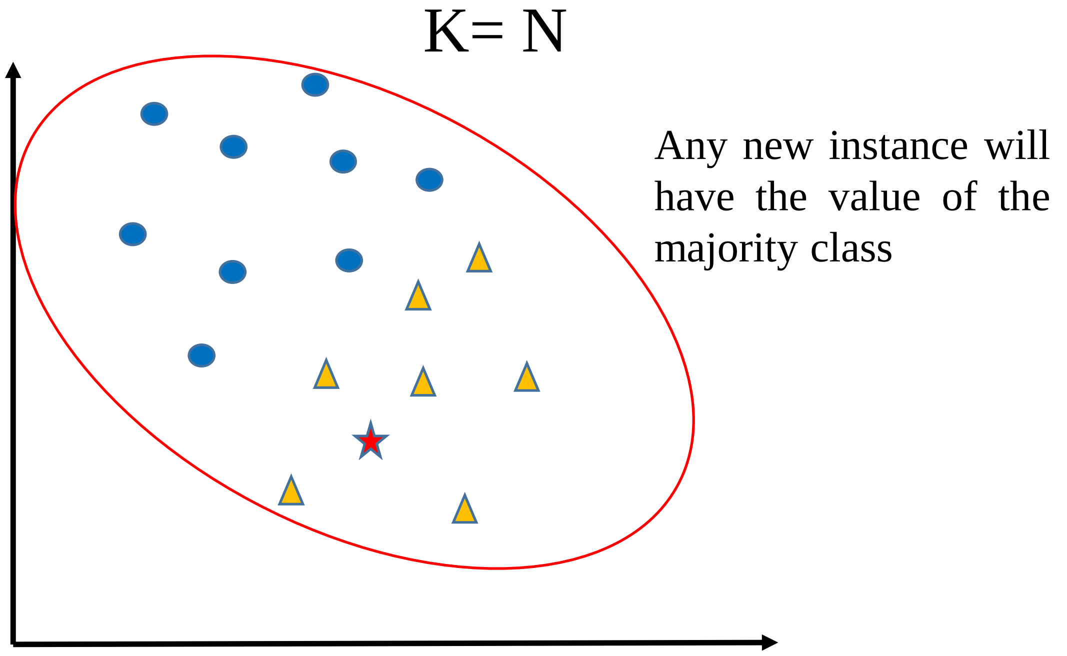
K= 3

New Instance = (Age = 48, Loan=$142,000, **Default = N**)

Choosing the right value for K

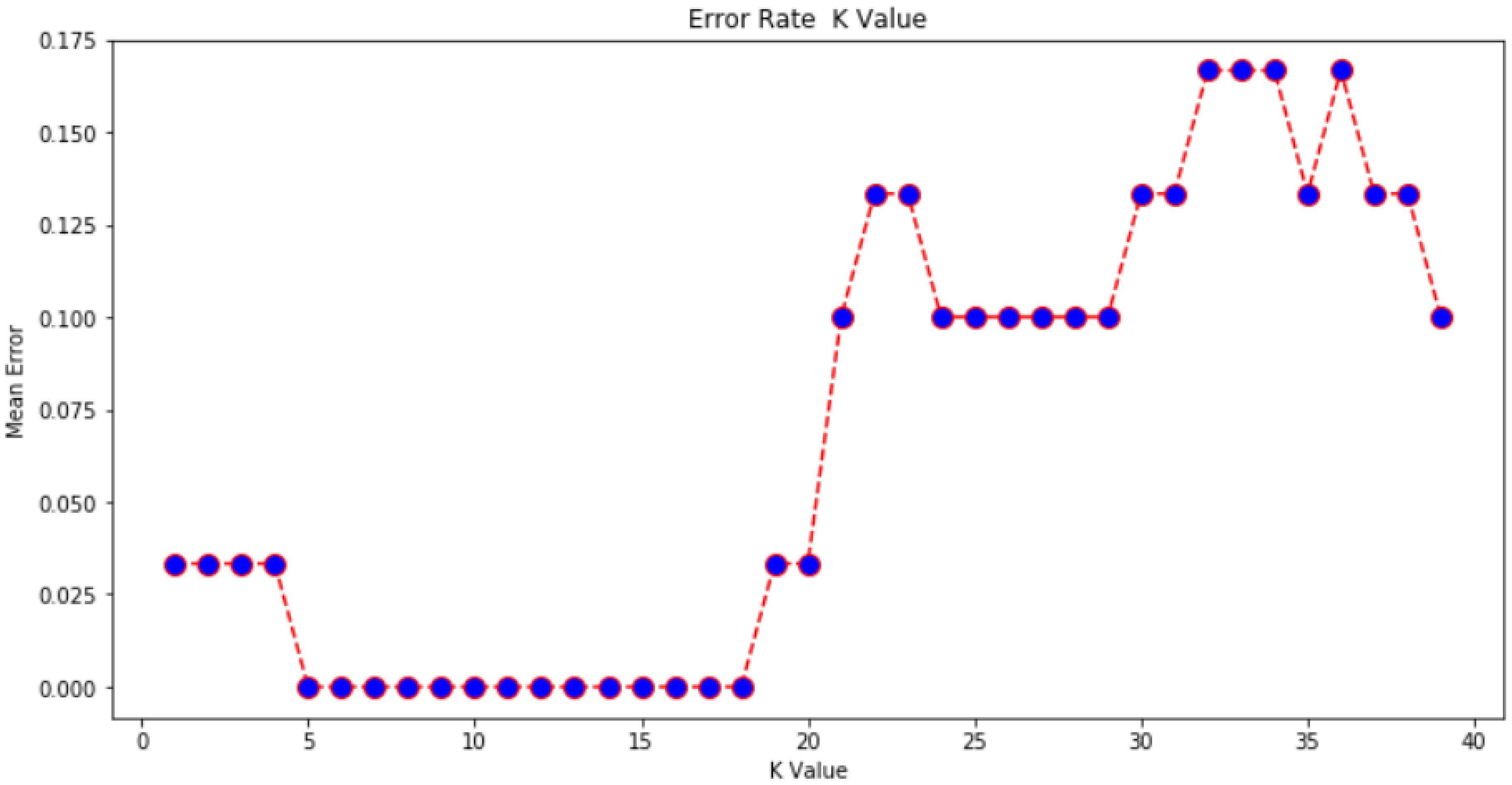
We usually make K an odd number to have a tiebreaker





To select the K that's right for the data, we run the KNN algorithm several times with different values of K and choose the K that reduces the number of errors

Comparing Error Rate with the K Value



The mean error is zero when the value of the K is between 5 and 18

K-Nearest Neighbors - Regression

A simple implementation of KNN regression is to calculate the average of the numerical target of the K nearest neighbors

$$X_{new} = \frac{x_1 + x_2 + \cdots + x_k}{k}$$

Another approach uses an inverse distance weighted average of the K nearest neighbors

$$X_{new} = \frac{w_1 x_1 + w_2 x_2 + \cdots + w_k x_k}{w_1 + w_2 + \cdots + w_k} \quad \text{where} \quad w_i = \frac{1}{d_i} \quad , \quad d_i = \text{distance}$$

KNN regression uses the same distance functions as KNN classification

UPAEP

New Instance = (Age = 48, Loan=$142,000)

Default= (8181+4575+9998)/3=7585

New Instance = (Age = 0.7, Loan=0.61)

Age	Loan	Default
25	$40,000	588
35	$60,000	7616
45	$80,000	8181
20	$20,000	7811
35	$120,000	4575
52	$18,000	9281
23	$95,000	4354
40	$62,000	9011
60	$100,000	9998
48	$220,000	1687
33	$150,000	2104

$$X_s = \frac{X - Min_{a_i}}{Max_{a_i} - Min_{a_i}}$$

Age	Loan
0.125	0.11
0.375	0.21
0.625	0.31
0	0.01
0.375	0.50
0.8	0.00
0.075	0.38
0.5	0.22
1	0.41
0.7	1.00
0.325	0.65

Dist. Age	Dist. Loan	D	
-0.5750	-0.5050	0.7652	
-0.3250	-0.4059	0.5200	
-0.0750	-0.3069	0.3160	1
-0.7000	-0.6040	0.9245	
-0.3250	-0.1089	0.3428	2
0.1000	-0.6139	0.6220	
-0.6250	-0.2327	0.6669	
-0.2000	-0.3960	0.4437	
0.3000	-0.2079	0.3650	3
0.0000	0.3861	0.3861	
-0.3750	0.0396	0.3771	

Inverse distance weighted average

New Instance = (Age = 48, Loan=$142,000)

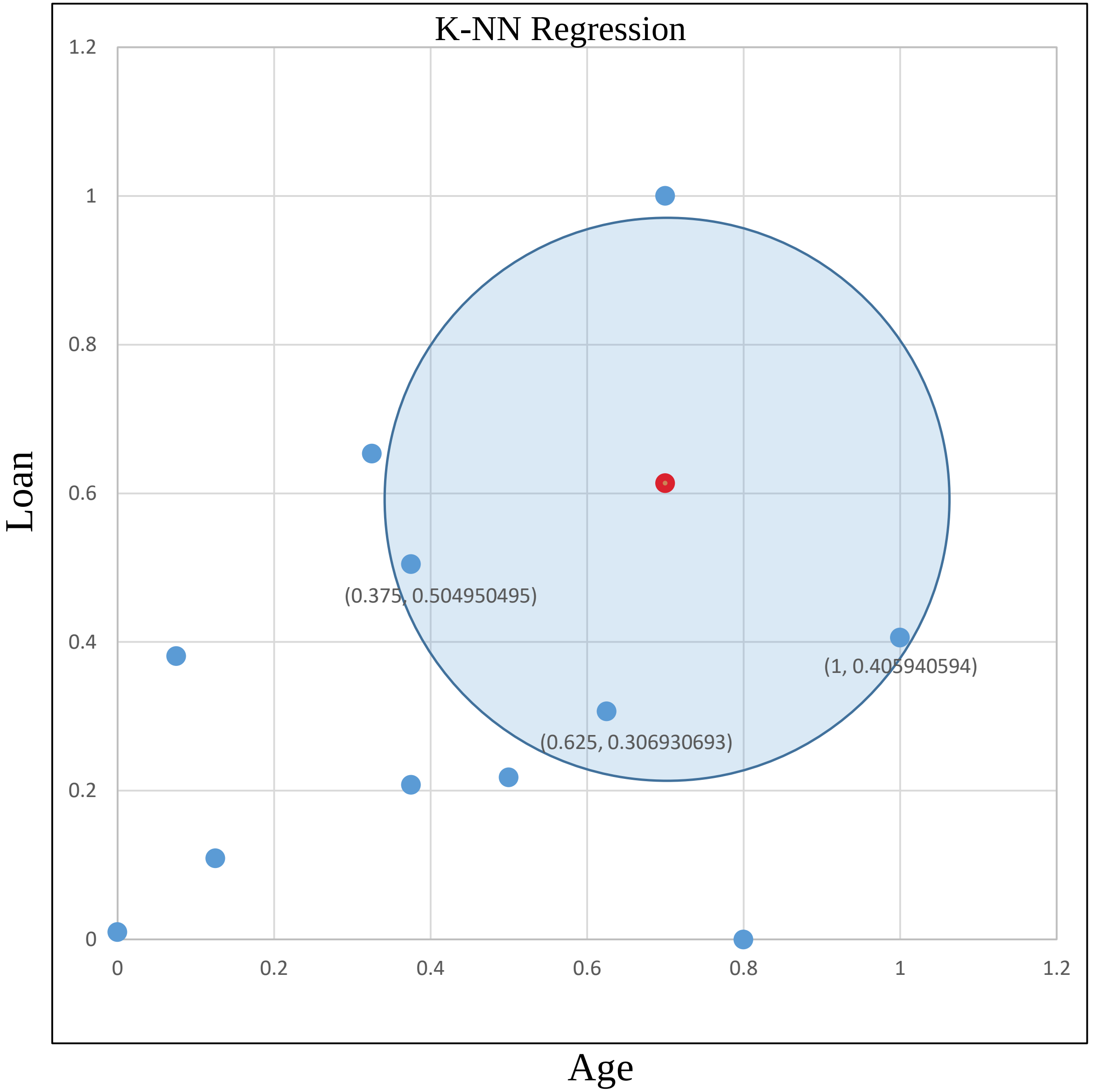
$$Default = \frac{\left(\frac{1}{0.3650}\right)9998 + \left(\frac{1}{0.3428}\right)4575 + \left(\frac{1}{0.3160}\right)8181}{\left(\frac{1}{0.3650}\right) + \left(\frac{1}{0.3428}\right) + \left(\frac{1}{0.3160}\right)} = 7553$$

Age	Loan	Default
25	$40,000	588
35	$60,000	7616
45	$80,000	8181
20	$20,000	7811
35	$120,000	4575
52	$18,000	9281
23	$95,000	4354
40	$62,000	9011
60	$100,000	9998
48	$220,000	1687
33	$150,000	2104

$$X_s = \frac{X - Min_{a_i}}{Max_{a_i} - Min_{a_i}}$$

Age	Loan
0.125	0.11
0.375	0.21
0.625	0.31
0	0.01
0.375	0.50
0.8	0.00
0.075	0.38
0.5	0.22
1	0.41
0.7	1.00
0.325	0.65

Dist. Age	Dist. Loan	D	
-0.5750	-0.5050	0.7652	
-0.3250	-0.4059	0.5200	
-0.0750	-0.3069	0.3160	1
-0.7000	-0.6040	0.9245	
-0.3250	-0.1089	0.3428	2
0.1000	-0.6139	0.6220	
-0.6250	-0.2327	0.6669	
-0.2000	-0.3960	0.4437	
0.3000	-0.2079	0.3650	3
0.0000	0.3861	0.3861	
-0.3750	0.0396	0.3771	


K-NN Regression
Loan
Age
1.2
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
1.2
(0.375, 0.504950495)
(1, 0.405940594)
(0.625, 0.306930693)

Estimate the weight value for the new instance

ID	Height	Age	Weight
1	1.52	45	77
2	1.56	26	47
3	1.71	30	70
4	1.80	34	74
5	1.46	40	54
6	1.77	36	60
7	1.62	19	55
8	1.77	28	60
9	1.68	23	67
10	1.71	32	75

11	1.68	38	?