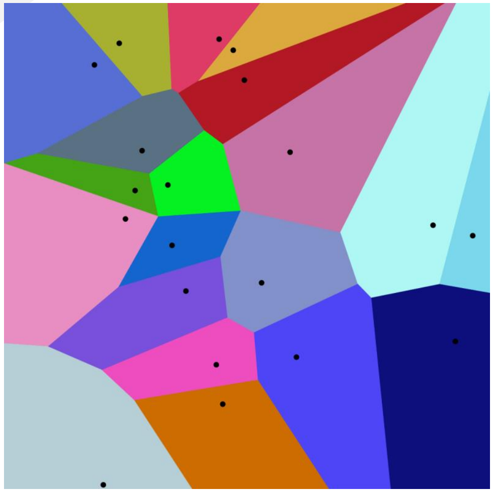


K Nearest Neighbor

Instance Based Learning

- Also known as "Lazy Learning"
- Store the given training data and don't learn any model
- During query time, retrieve a set of "similar" instances from the training data and use them to classify/predict the new instance
- Essentially construct only local approximations to the target function
- There is no global model learnt to perform well across all instances

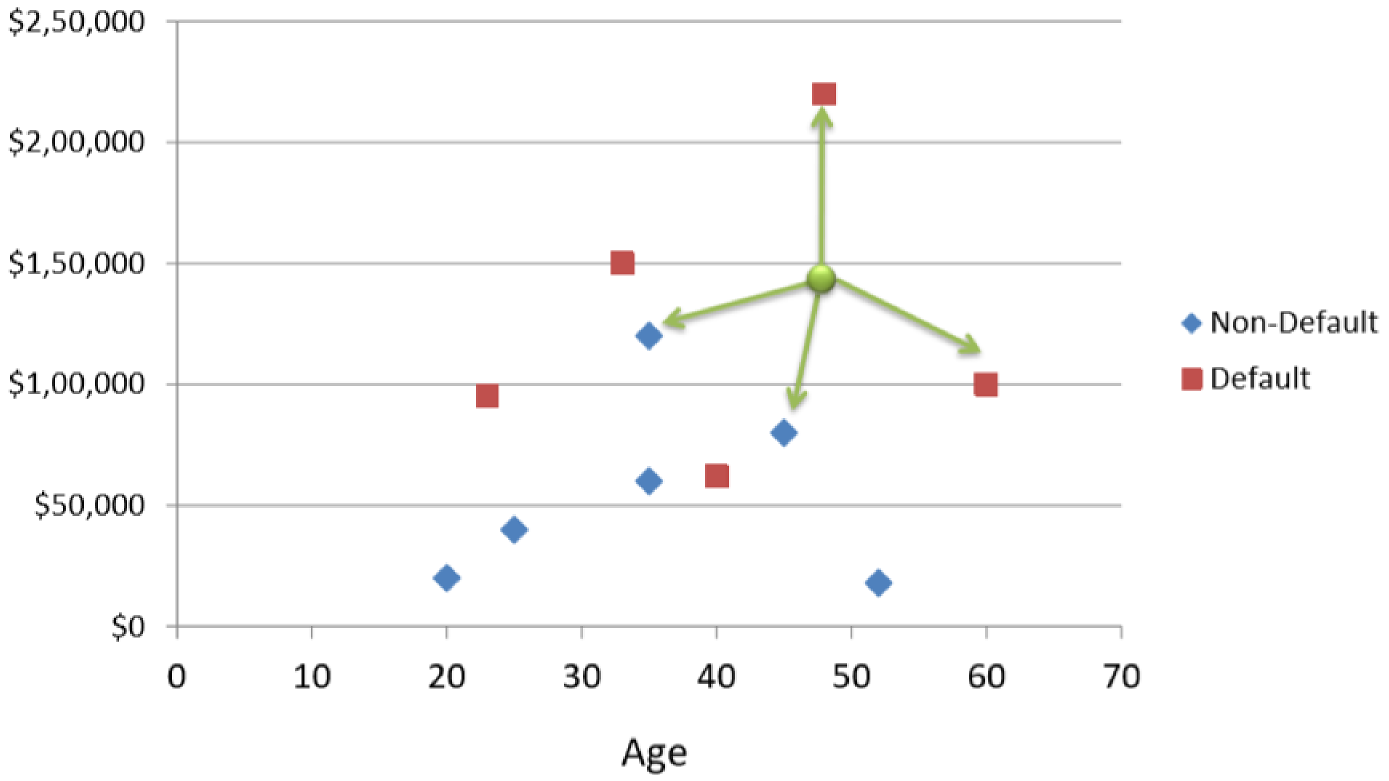
K-NN (K-Nearest Neighbours)

- One of the most basic forms of instance learning
- K-NN Algorithm for Classification
- Training method:
  - Save the training examples
- At prediction time:
  - Find the k training examples (x1,y1),…(xk,yk) that are closest to the test example x
  - Predict the most frequent class among those yi’s.

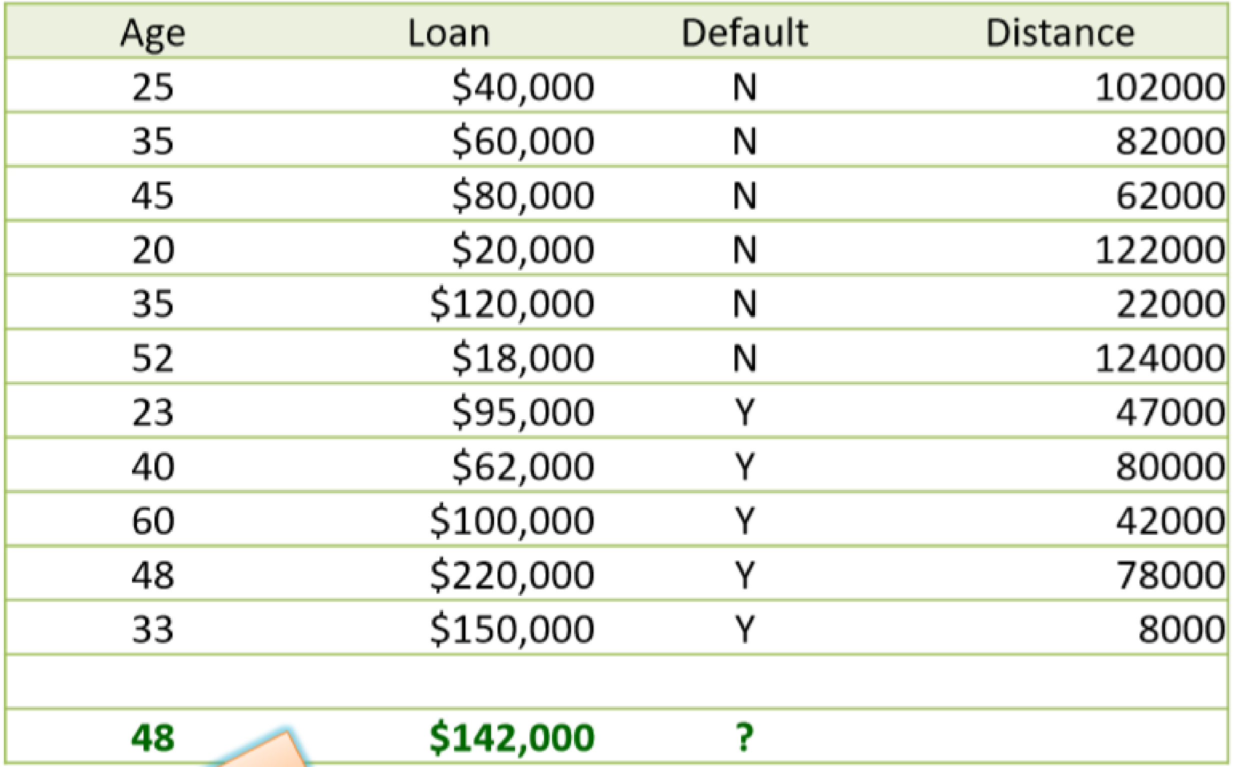
K-NN - Classification

- One of the most basic forms of instance learning
- K-NN Algorithm for Classification
- Training method:
  - Save the training examples
- At prediction time:
  - Find the k training examples (x1,y1),…(xk,yk) that are closest to the test example x
  - Predict the most frequent class among those yi’s.

K-NN - Classification



K-NN – Classification (Contd..)



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

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

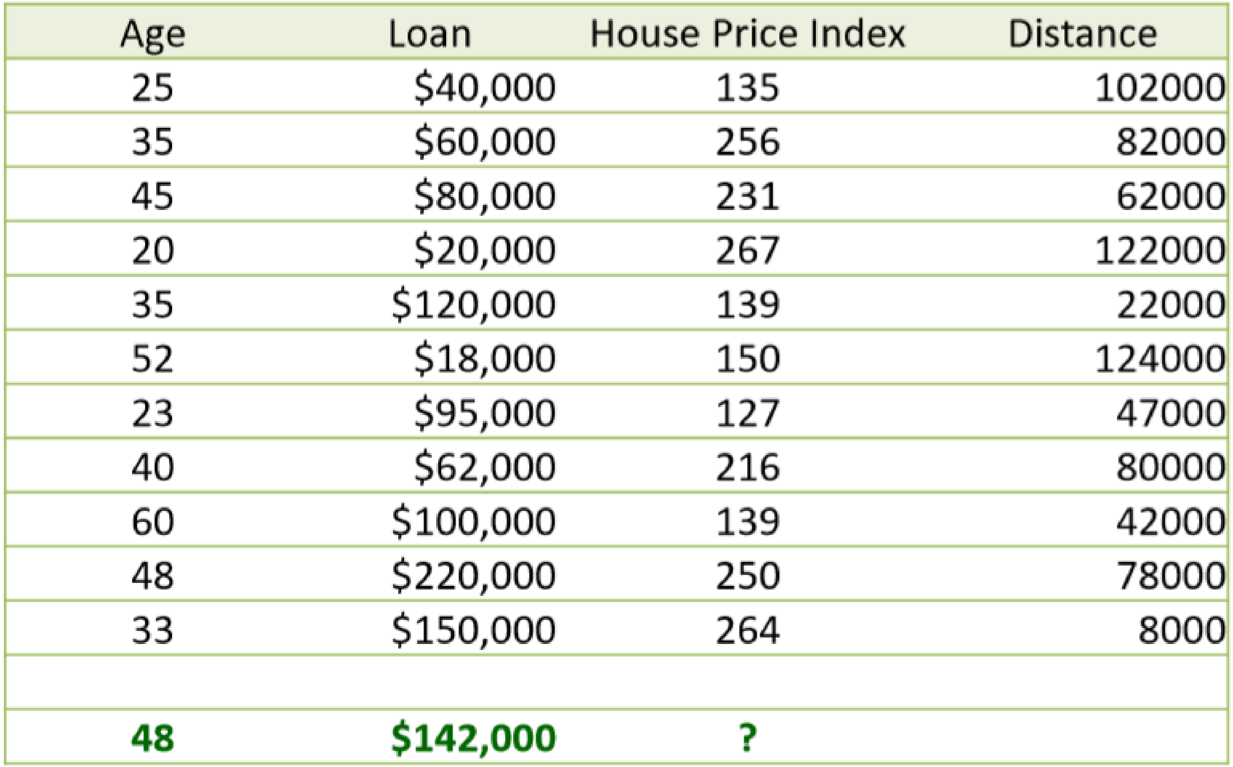
K-NN – Classification (Contd ..)

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

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

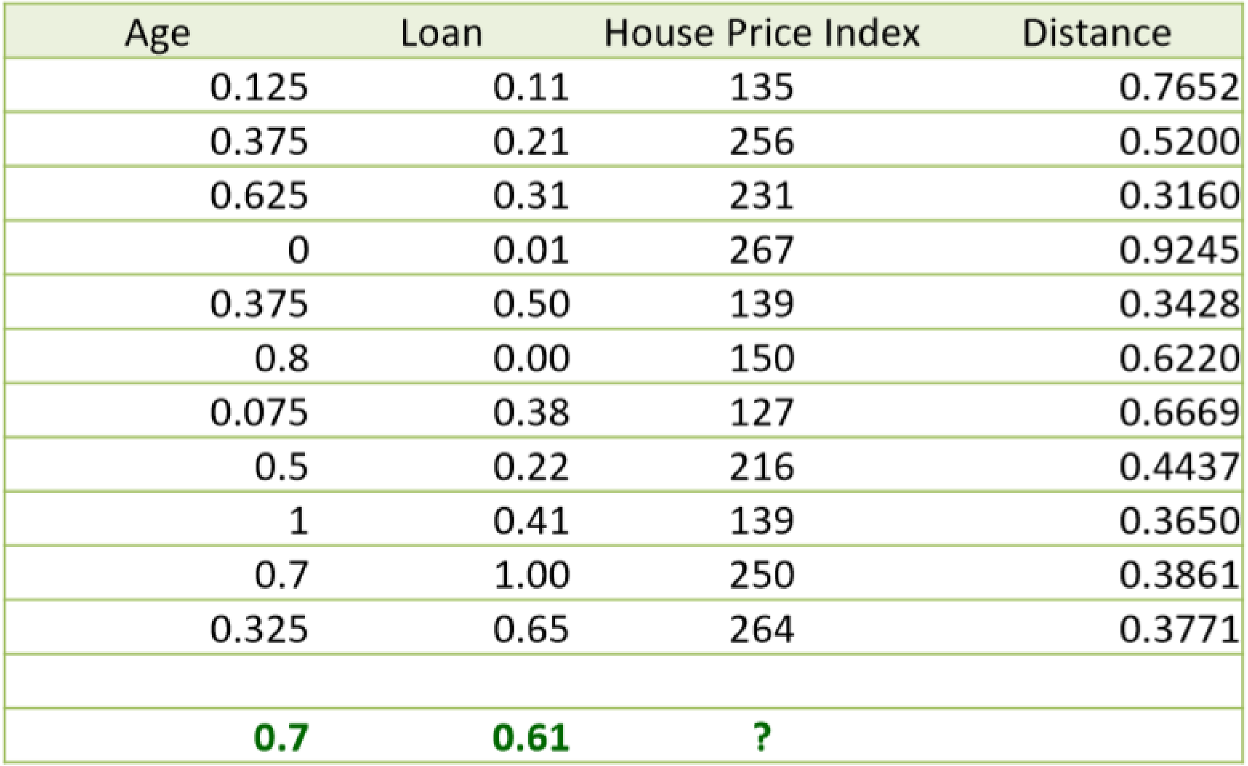
K-NN - Regression



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**	**?**	

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

K-NN Regression (Contd..)

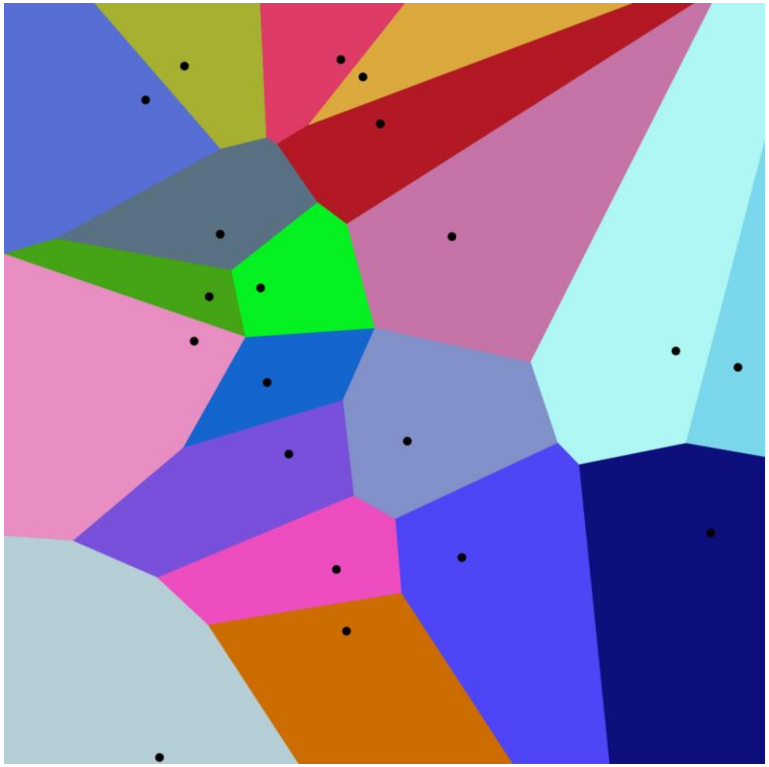


Age	Loan	House Price Index	Distance
0.125	0.11	135	0.7652
0.375	0.21	256	0.5200
0.625	0.31	231	0.3160
0	0.01	267	0.9245
0.375	0.50	139	0.3428
0.8	0.00	150	0.6220
0.075	0.38	127	0.6669
0.5	0.22	216	0.4437
1	0.41	139	0.3650
0.7	1.00	250	0.3861
0.325	0.65	264	0.3771
0.7	**0.61**	**?**	

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

K-NN Decision Boundaries

- Voronoi Diagrams



How to determine a good value of “K”

- Usually tuned using a validation set
- Start with k=1 and test the error rate on validation set
- Repeat with k=k+2
- Choose the value of k which has minimum error rate on validation set
- Note: Odd values of k chosen to avoid ties

Improving K-NN

- Weighting examples from the neighborhood
- Measuring “closeness”
- Finding “close” examples in a large training set quickly

Reference