




Inspire…Educate…Transform.

Instance Based Learning

Dr. Manoj Chinnakotla
Senior Applied Scientist, Microsoft
Adjunct Professor, IIIT Hyderabad

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:

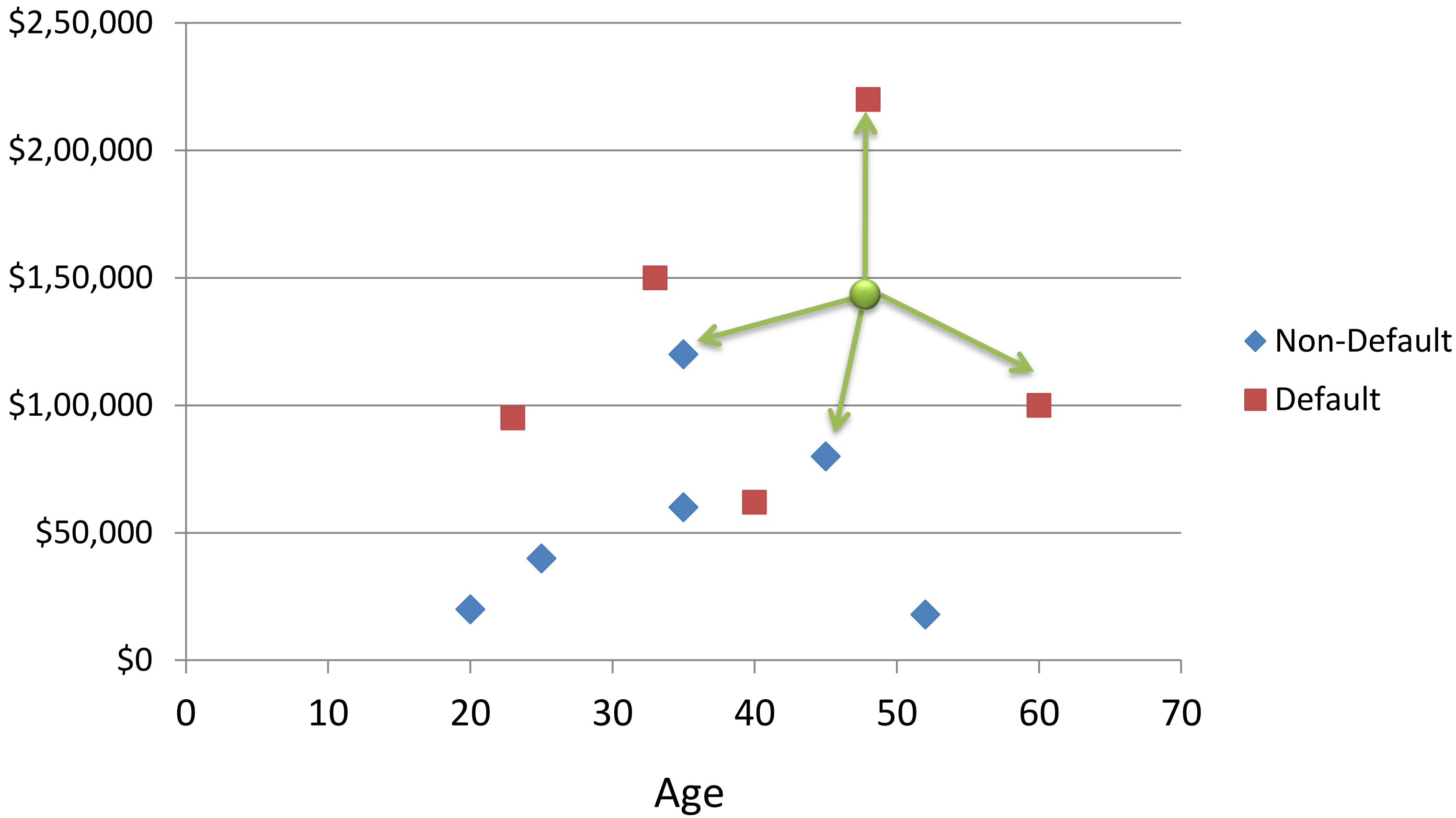
Save the training examples

At prediction time:

Find the k training examples $(x_1,y_1),\ldots(x_k,y_k)$ that are closest to the test example x

Predict the most frequent class among those y_i'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

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

CSE 7306c

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

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

CSE 7306c

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..)

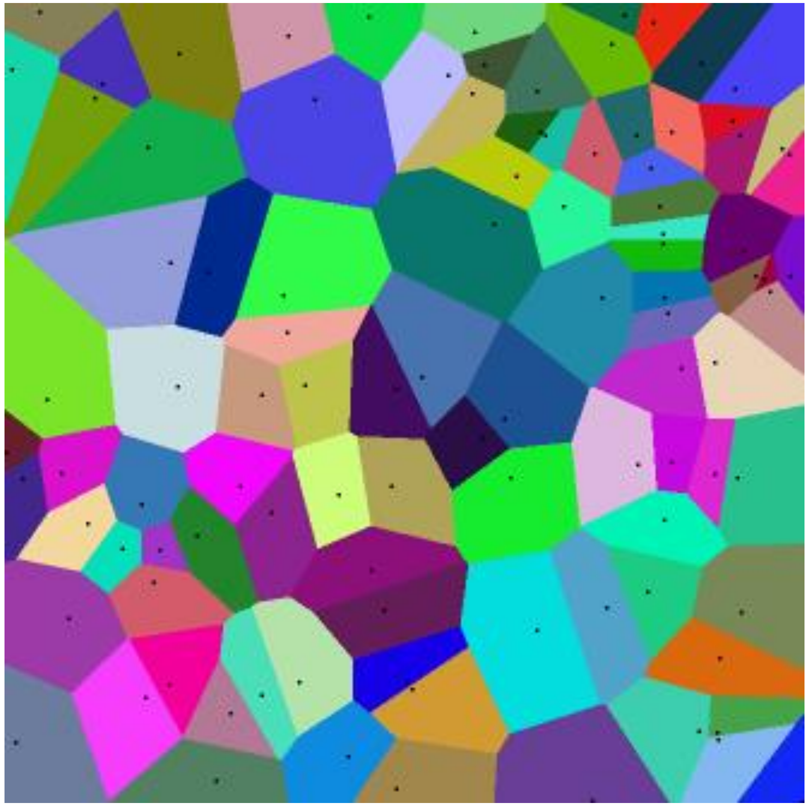
Age	Loan	House Price Index	Distance
0.125	0.11	135	0.7652
0.375	0.21	256	0.5200
0.625	0.31	231	0.3160
0	0.01	267	0.9245
0.375	0.50	139	0.3428
0.8	0.00	150	0.6220
0.075	0.38	127	0.6669
0.5	0.22	216	0.4437
1	0.41	139	0.3650
0.7	1.00	250	0.3861
0.325	0.65	264	0.3771
0.7	**0.61**	**?**	

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

CSE 7306c

K-NN Decision Boundaries

Voronoi Diagram



Let's play with K-NN

- http://sleepyheads.jp/apps/knn/knn.html
- http://scott.fortmann-roe.com/docs/BiasVariance.html

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*

Distance-weighted K-NN

- Refinement to kNN is to weight the contribution of each *k* neighbor according to the distance to the query point x_q
  - Greater weight to closer neighbors
  - For discrete target functions

$$\hat{f}(x_q) \leftarrow \underset{v \in V}{\arg\max} \sum_{i=1}^{k} w_i \delta(v, f(x_i))$$

$$w_i = \begin{cases} \dfrac{1}{d(x_q, x_i)^2} & if \quad x_q \neq x_i \\ 1 & else \end{cases}$$

CSE 7306c

Distance-weighted K-NN (Contd..)

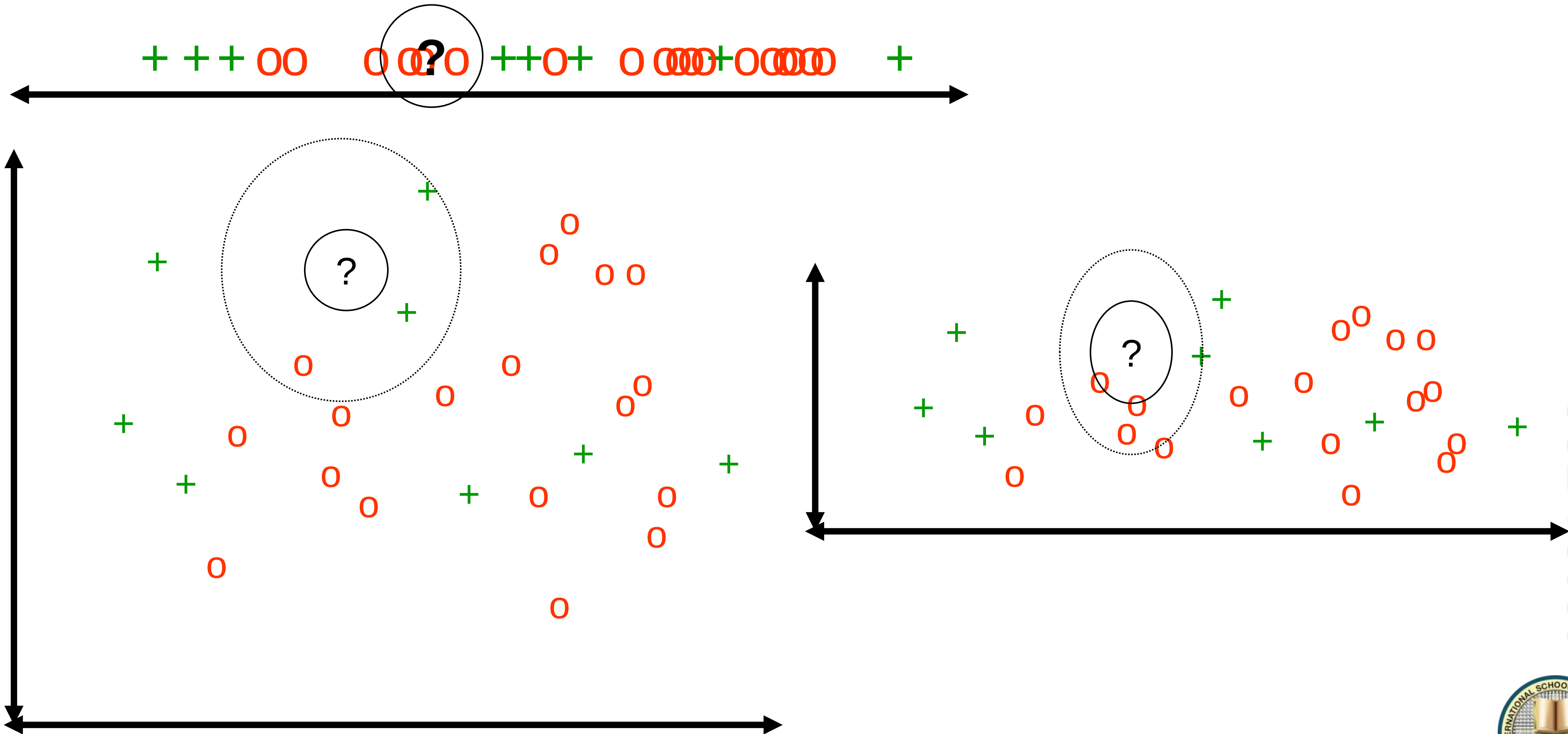
- For real valued functions:

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i}$$

$$w_i = \begin{cases} \dfrac{1}{d(x_q, x_i)^2} & if \quad x_q \neq x_i \\ 1 & else \end{cases}$$

Feature Scaling and Selection

- K-NN is highly sensitive to the scaling and the subset of features selected as it influences the neighbors



Curse of Dimensionality

- Irrelevant features heavily mislead kNN
- Especially true if the dimensionality of the space is high
- Possible Solutions:
  - PCA

A few ways of rescaling distances

- Normalized L1 Distance

$$\Delta(X,Y) = \sum_{i=1}^{n} \delta(x_i, y_i)|$$

where:

$$\delta(x_i, y_i) = \begin{cases} abs(\frac{x_i - y_i}{max_i - min_i}) & \text{if numeric, else} \\ 0 & \text{if } x_i = y_i \\ 1 & \text{if } x_i \neq y_i \end{cases}$$

- Scale using Information Gain (IG)

$$\Delta(X,Y) = \sum_{i=1}^{n} w_i \, \delta(x_i, y_i)$$

$$w_i = H(C) - \sum_{v \in V_i} P(v) \times H(C|v)$$

CSE 7306c

A few ways of rescaling distances

- For text, we can use TF-IDF

#occur. of term i in doc j

$$\mathrm{tf_{i,j}} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

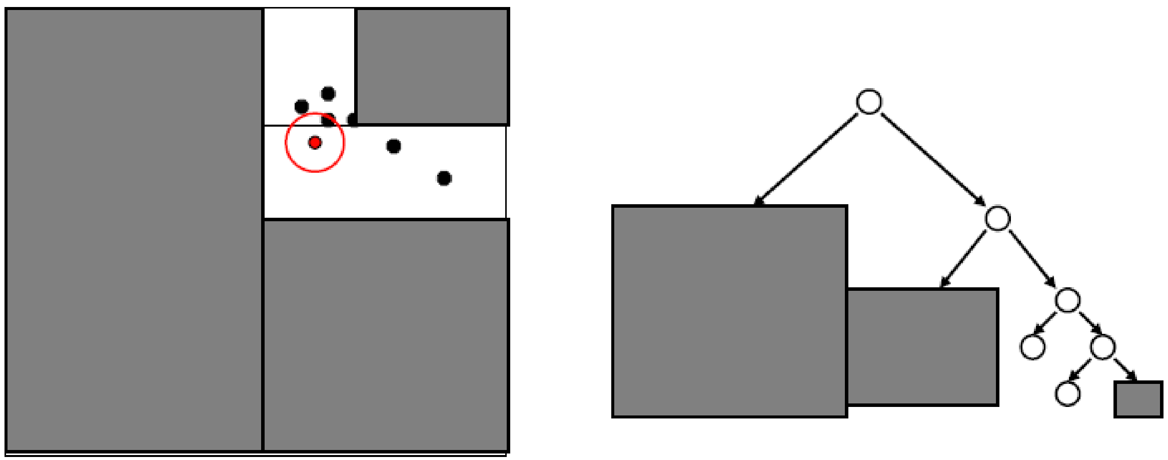
$$\mathrm{idf_i} = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|}$$

#docs in corpus

#docs in corpus that contain term i

Speeding up kNNs

- KD Trees could be used to efficient retrieve closest neighbors for a given query



Using the distance bounds and the bounds of the data below each node, we can prune parts of the tree that could NOT include the nearest neighbor.

kNNs – Pros and Cons

- Storage: All training examples are saved in memory
  - A decision tree or linear classifier is much smaller
- Time: To classify x, you need to loop over all training examples (x',y') to compute distance between x and x'.
  - However, you get predictions for every class y
    - kNN is nice when there are many many classes
  - There are some tricks to speed this up…especially when data is sparse

Case Studies - Discussion

- Handwritten Digit Recognition
- Understanding sign language

Case Studies - Discussion

CSE 7306c

Summary

- kNN is an example of "Instance Based Learning"
- Conceptually simple, yet able to solve complex problems
- Can work with relatively little information
- Learning is simple (no learning at all!)
- Suffers from the curse of dimensionality
  - Sensitive to representation
  - Feature selection and weighting extremely important
- For practical applications, need to use data structures to speed up retrieval of "close" neighbours

References

- Hastie, Tibshirani and Friedman, *"Elements of Statistical Learning: Data Mining, Inference and Prediction"*, Springer
- Duda, Hart and Stork, *"Pattern Classification"*, Wiley Publication
- Tom Mitchell, *"Machine Learning"*, McGraw Hill

International School of Engineering

Plot 63/A, 1st Floor, Road # 13, Film Nagar, Jubilee Hills, Hyderabad - 500 033

For Individuals: +91-9502334561/63 or 040-65743991

For Corporates: +91-9618483483

Web: http://www.insofe.edu.in

Facebook: https://www.facebook.com/insofe

Twitter: https://twitter.com/Insofeedu

YouTube: http://www.youtube.com/InsofeVideos

SlideShare: http://www.slideshare.net/INSOFE

LinkedIn: http://www.linkedin.com/company/international-school-of-engineering