CS145 Discussion
Week 5

Junheng, Shengming, Yunsheng

11/2/2018

Roadmap

- Announcement
- K-NN
- Similarity Metrics
- ROC
- K-Means
- Homework 3
- Course Project Midterm Report

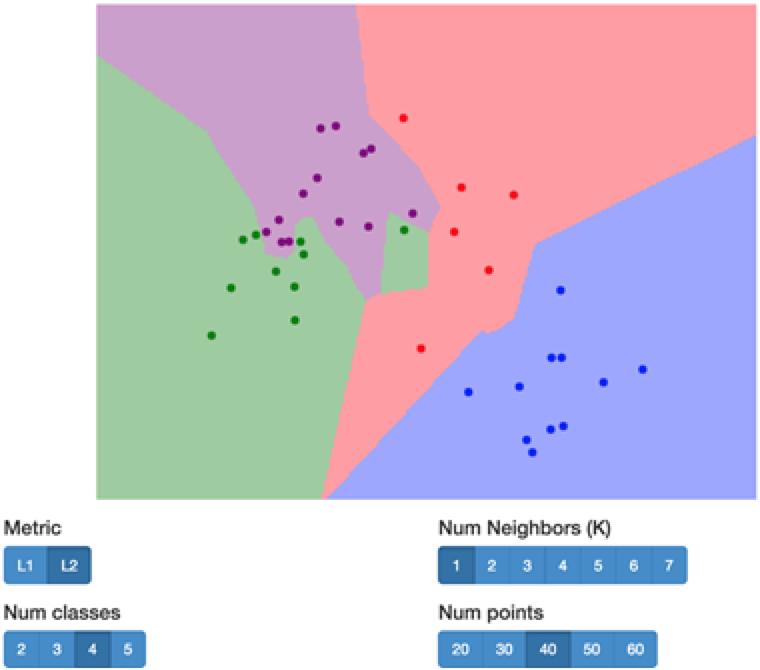
Announcement

- Homework 3 out
  - Deadline: 11/09 Friday 11:59 pm
  - KNN(30%) and Neural Network(70%)
  - Environment Requirement: Jupyter + Python 3
- Course Project Midterm Report (about to) out
  - Deadline: 11/12 Monday 11:59 pm
  - According to the guildline file
- Midterm date out
  - 11/14 Wednesday 12:00-1:50 pm(in class)
  - Remember to carry: one-page reference paper(letter size), simple calculator

KNN

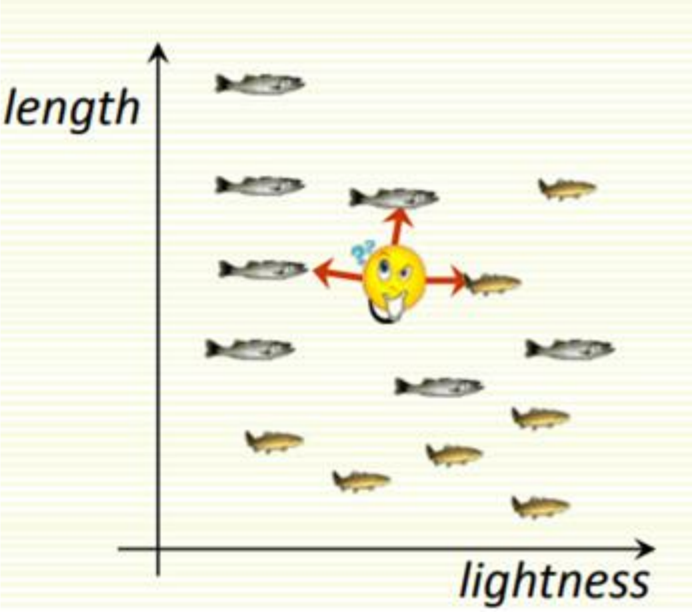
KNN

- Demo: http://vision.stanford.edu/teaching/cs231n-demos/knn/



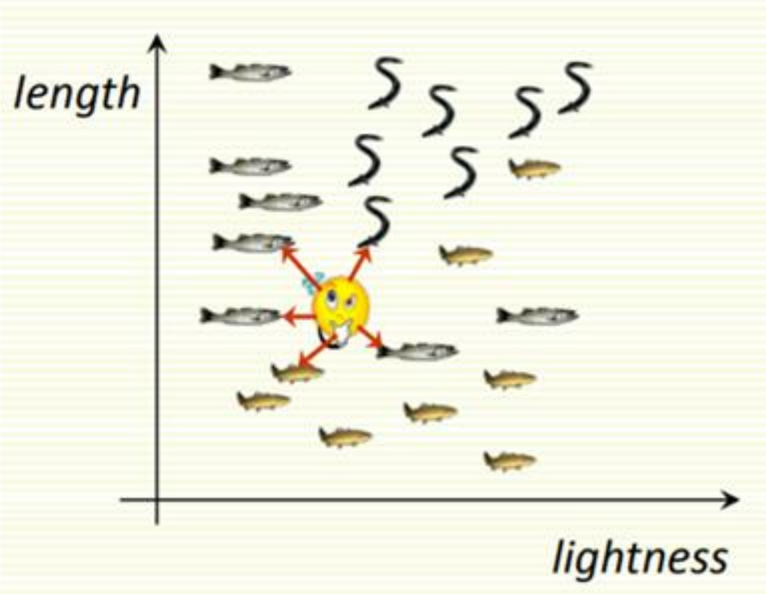
KNN

- Classify an unknown example with the most common class among K nearest examples
  - “Tell me who your neighbors are, and I’ll tell you who you are”
- Example
  - K = 3
  - 2 sea bass, 1 salmon
  - Classify as sea bass



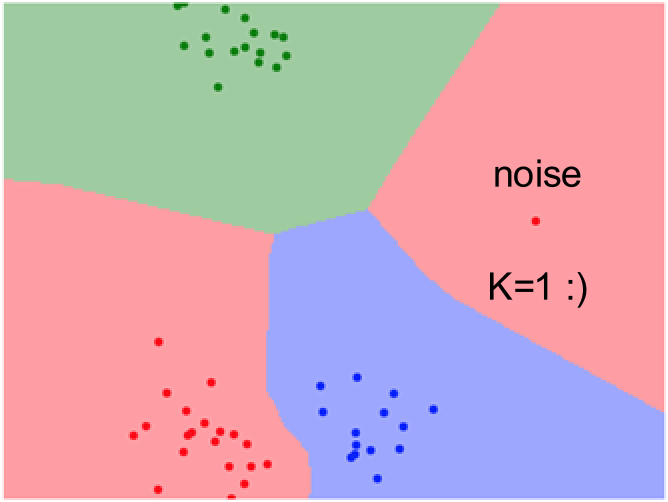
KNN: Multiple Classes

- Easy to implement for multiple classes
- Example for K = 5
  - 3 fish species: salmon, sea bass, eel
  - 3 sea bass, 1 eel, 1 salmon → classify as sea bass



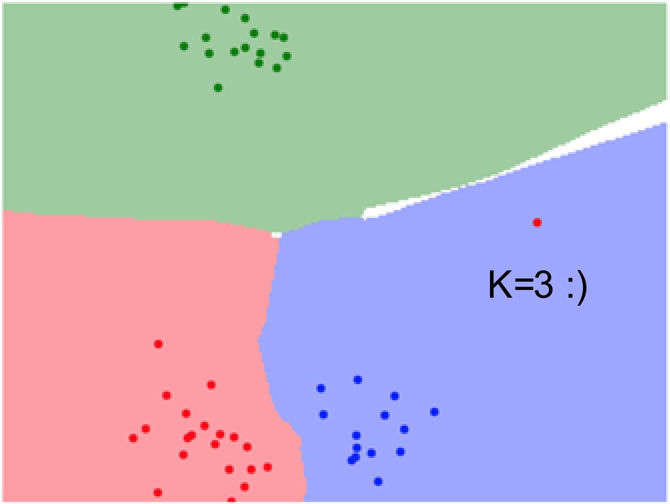
KNN: How to Choose K?

- In theory, if infinite number of samples available, the larger k, the better classification result you'll get.
- Caveat: all K neighbors have to be close
  - Possible when infinite # samples available
  - Impossible in practice since # samples if finite
- Should we "tune" K on training data?
  - Overfitting
- K = 1 → sensitive to "noise" (e.g. see right)

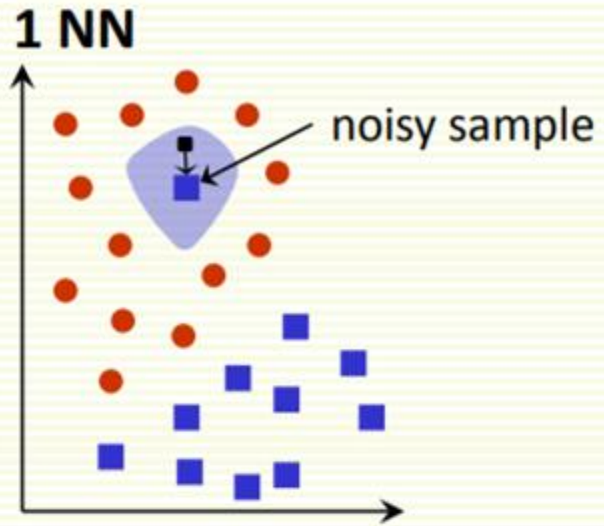


KNN: How to Choose K?

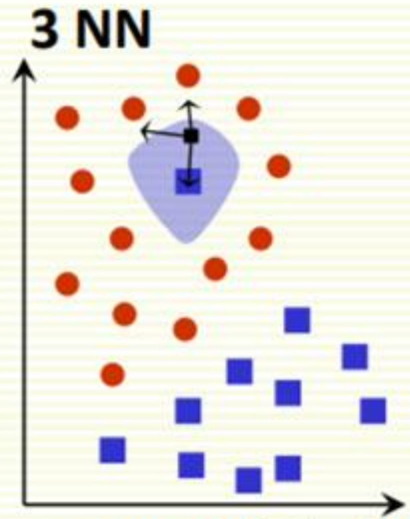
- In theory, if infinite number of samples available, the larger k, the better classification result you'll get.
- Caveat: all K neighbors have to be close
  - Possible when infinite # samples available
  - Impossible in practice since # samples if finite
- Should we "tune" K on training data?
  - Overfitting
- K = 1 → sensitive to "noise" (e.g. see right)



KNN: How to Choose K?



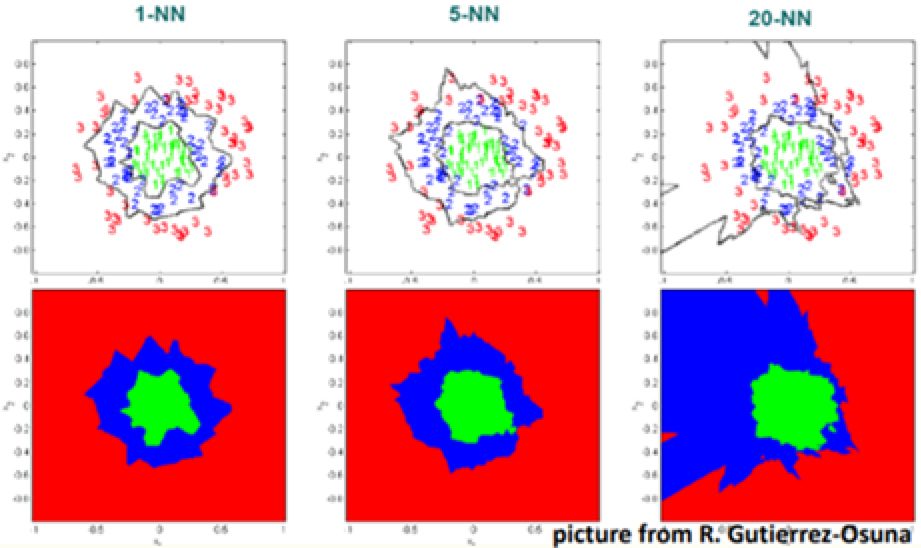
every example in the blue shaded area will be misclassified as the blue class



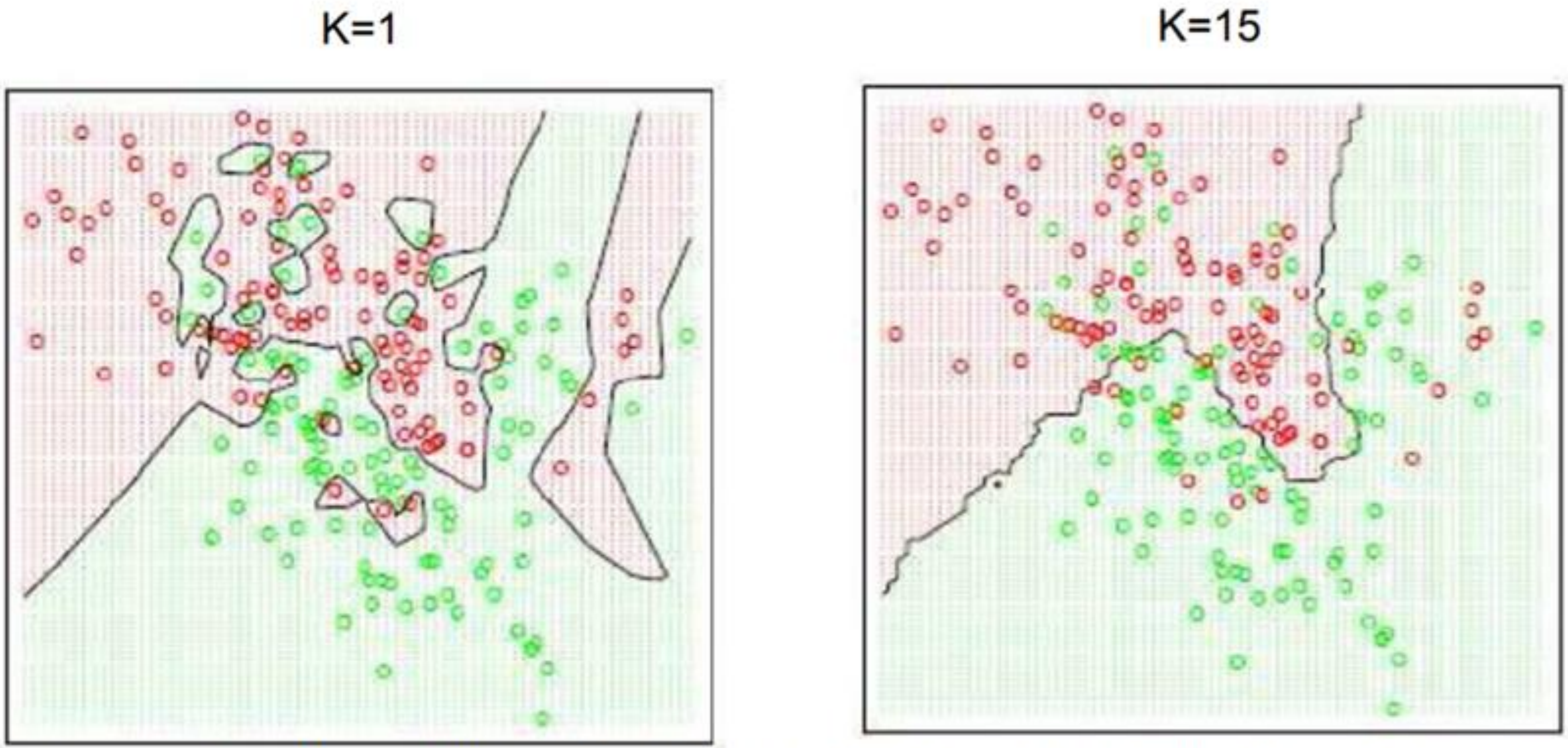
every example in the blue shaded area will be classified correctly as the red class

KNN: How to Choose K?

- Larger K gives smoother boundaries, better for generalization
  - Only if locality is preserved
  - K too large → looking at samples too far away that are not from the same class
- Can choose K through cross-validation



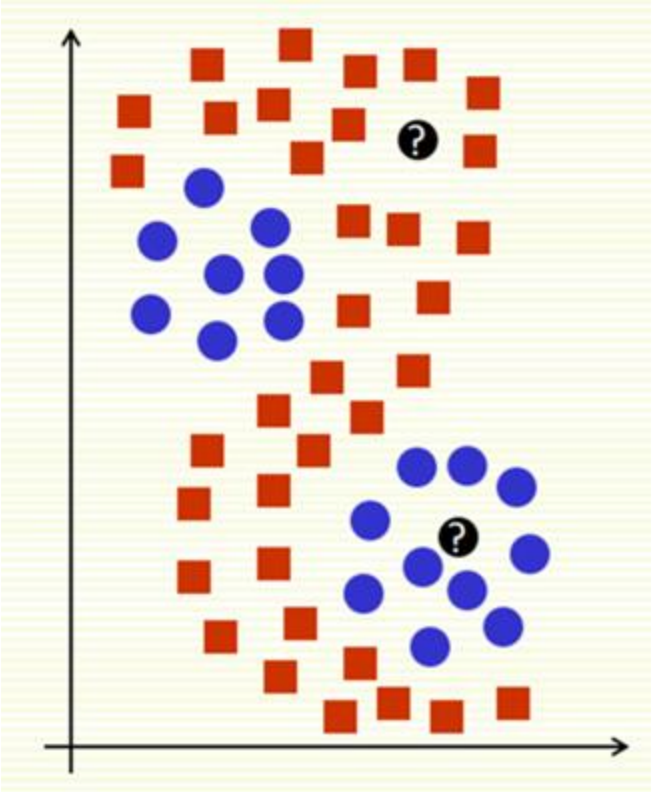
KNN: How to Choose K?



Figures from Hastie, Tibshirani and Friedman (Elements of Statistical Learning)

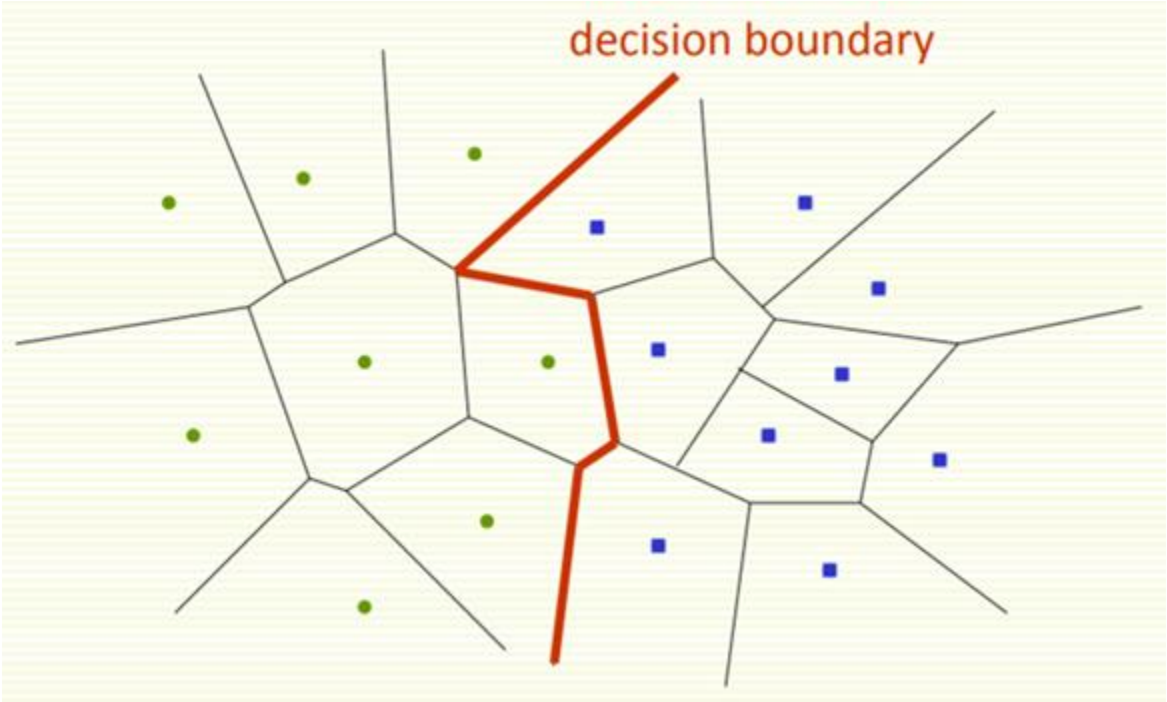
KNN: Multi-Model Distributions

- Many classification models would not work for this 2-class classification problem
    - Linear-R
    - Logistic-R
    - Decision-Tree
    - SVM
- Nearest neighbors will do reasonably well



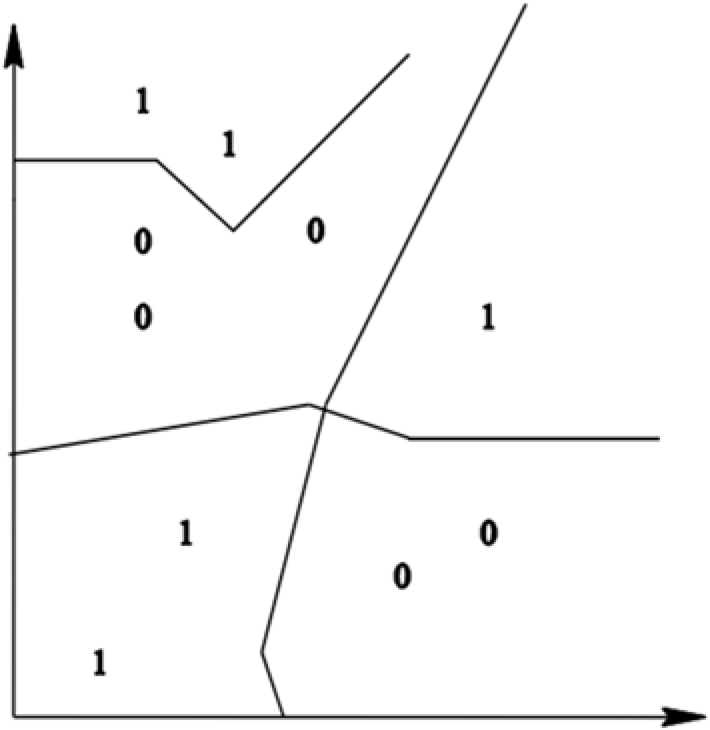
Decision Boundaries

- Voronoi diagram is useful for visualization



Decision Boundaries

- Decision boundaries are formed by a subset of the Voronoi Diagram of the training data
- Each line segment is equidistant between two points of opposite class
- The more examples that are stored, the more fragmented and complex the decision boundaries can be.



KNN: Selection of Distance

- We use Euclidean Distance to find the nearest neighbor:

$$D(a, b) = \sqrt{\sum_k (a_k - b_k)^2}$$

- Euclidean distance treats each feature as equally important
- Sometimes, some features (or dimensions) may be much more discriminative than other features

KNN Distance Selection: Extreme Example

- Feature 1 gives the correct class: 1 or 2
- Feature 2 gives irrelevant number from 100 to 200
- Dataset: [1, 150], [2, 110]
- Classify [1, 100]

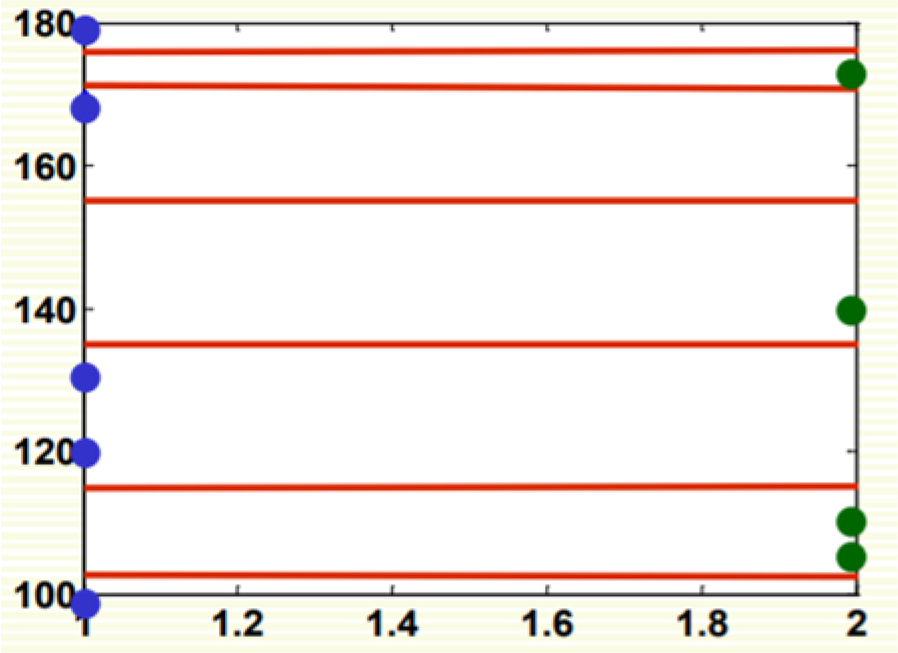
$$D\left(\begin{bmatrix}1\\100\end{bmatrix},\begin{bmatrix}1\\150\end{bmatrix}\right)=\sqrt{(1-1)^2+(100-150)^2}=50$$

$$D\left(\begin{bmatrix}1\\100\end{bmatrix},\begin{bmatrix}2\\110\end{bmatrix}\right)=\sqrt{(1-2)^2+(100-110)^2}=10.5$$

- Use Euclidean distance can result in wrong classification
- Dense Example can help solve this problem

KNN Distance Selection: Extreme Example

- Decision boundary is in red, and is really wrong because:
  - Feature 1 is discriminative, but its scale is small
  - Feature gives no class information but its scale is large, which dominates distance calculation



KNN: Feature Normalization

- Normalize features that makes them be in the same scale
- Different normalization approaches may reflect the result
- Linear scale the feature in range [0,1]:

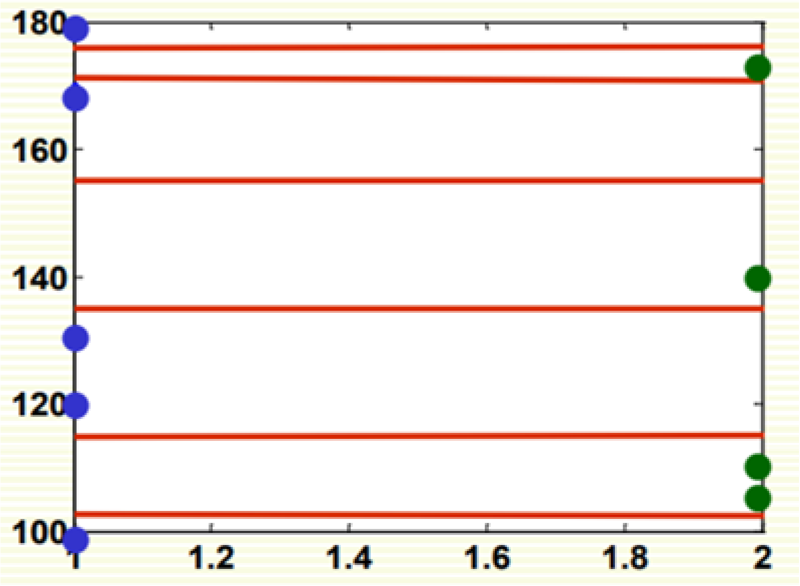
$$f_{new} = \frac{f_{\text{old}} - f_{\text{old}}^{\text{min}}}{f_{\text{old}}^{max} - f_{\text{old}}^{\text{min}}}$$

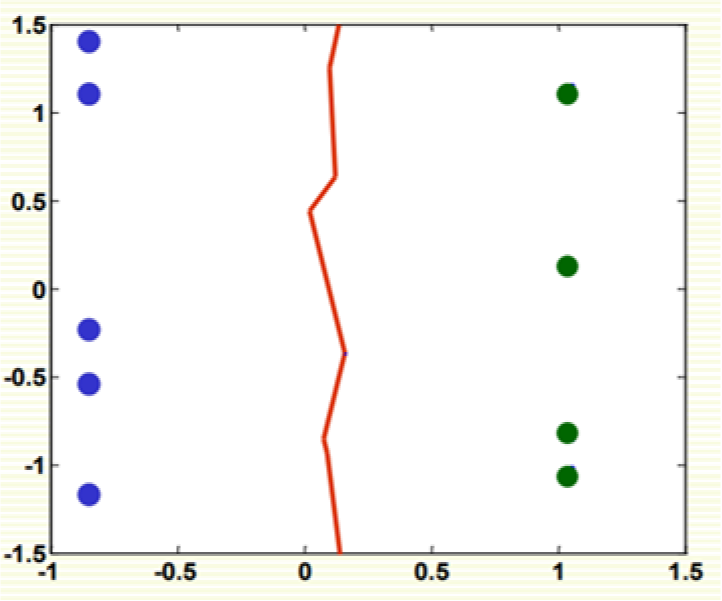
- Linear scale to 0 mean variance 1(Z-score):

$$f_{new} = \frac{f_{\text{old}} - \mu}{\sigma}$$

KNN: Feature Normalization

- Result comparison non-normalized vs normalized





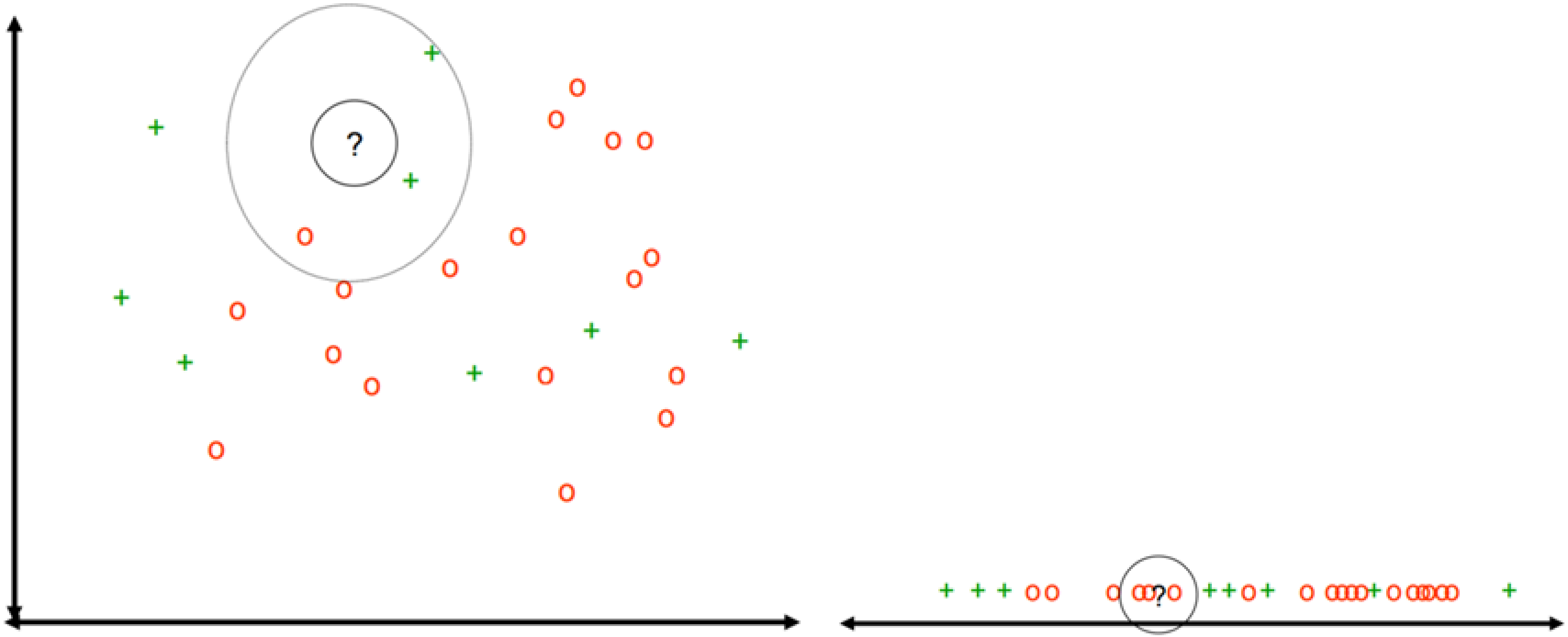
KNN: Selection of Distance

- Feature normalization does not help in high dimensional spaces if most features are irrelevant

$$D(a,b) = \sqrt{\sum_k (a_k - b_k)^2} = \sqrt{\underbrace{\sum_i (a_i - b_i)^2}_{\text{Discriminative features}} + \underbrace{\sum_j (a_j - b_j)^2}_{\text{Noisy features}}}$$

- If the number of useful feature is smaller than the number of noisy features, Euclidean distance is dominated by noise.

KNN: Example of Noise Domination Problem



KNN: Feature Weighting

- Scale each feature by its importance for classification

$$D(a,b) = \sqrt{\sum_k w_k (a_k - b_k)^2}$$

- Use prior/domain knowledge to set the weight w
- Use cross-validation to learn the weight w

KNN: Computational Complexity

- Suppose n examples with dimension d
- For each point to be classified:
    - O(d) to compute distance to one example
    - O(nd) to compute distances to all examples
    - O(nk) time to find k closest examples
    - Total time: O(nk + nd)
- Very expensive for a large number of queries

KNN: Reduce Complexity

- Reduce the dimensionality of the data:
  - Find a projection from high dimensional space to a lower dimensional space so that the distance between samples are approximately the same
  - Use Principal Component Analysis(PCA)
- Apply smart data structure, e.v. k-d trees

KNN: Summary

- Advantages:
  - Can be applied to the data from any distribution
  - The decision boundary is not necessarily to be linear
  - Simple and Intuitive
  - Good Classification with large number of samples
- Disadvantages:
  - Chossing k may be tricky
  - Test stage is computationally expensive
    - No training stage, time-consuming test stage
    - Usually we can afford long training step but fast testing speed
  - Need large number of examples for accuracy

Reference

- http://www.csd.uwo.ca/courses/CS9840a/Lecture2_knn.pdf
- http://classes.engr.oregonstate.edu/eecs/spring2012/cs534/notes/knn.pdf
- http://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture10.pdf

Similarity Metrics

Similarity Metrics

- Dissimilarity
- $\begin{bmatrix} 3 & 5 \\ 6 & 9 \\ 11 & 21 \end{bmatrix}$
- 3 2-d input, x1,x2,x3
- The dissimilarity matrix is a 3*3 lower-triangular matrix:

$$\begin{bmatrix} 0 & 0 & 0 \\ d(2,1) & 0 & 0 \\ d(3,1) & d(3,2) & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ \sqrt{(3-6)^2+(5-9)^2} & 0 & 0 \\ \sqrt{(3-11)^2+(5-21)^2} & \sqrt{(11-6)^2+(21-9)^2} & 0 \end{bmatrix}$$

Dissimilarity: Nominal Atributes

- Student 1: likes Jazz, eats pizza, roots for the cubs, wears socks
- Student 2: likes Rock, eats pizza, roots for the cubs, goes barefoot
- d(Student 1, Student 2):
  - m: # of matches → 2
  - p: total # of variables → 4
  - d(Student 1, Student 2) = (4-2)/4 = 0.5

Binary Attributes

- Symmetric binary attributes:
  - Gender
- Asymmetric attributes:
  - Preference, Character, etc.
- Can be manually defined

Binary Attributes

- Dissimilarity of Binary Attributes:
  - Define 0 and 1
  - calculate q,s,r,t,p
- Distance measure for symmetric binary variables:

$$d(i, j) = \frac{r + s}{q + r + s + t}$$

		Object j		
		1	0	sum
Object i	1	q	r	$q+r$
	0	s	t	$s+t$
	sum	$q+s$	$r+t$	p

- Distance measure for asymmetric binary variables:

$$d(i, j) = \frac{r + s}{q + r + s}$$

- Jaccard coefficient (similarity measure for asymmetric binary variables):

$$sim_{Jaccard}(i, j) = \frac{q}{q + r + s}$$

Binary Attributes: Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- i = Jack, j = Mary
- Define M,Y,P as 1; Define F,N as 0
- Assume symmetric attributes for all:
  - r = 0, s = 1, q = 2
- Assume symetric for Gender, asymetric for other attributes
  - For Gender: r = 1, s = 0, q = 0, t = 0
  - For others: r = 0, s = 1, q = 2

		Object j		
		1	0	sum
Object i	1	q	r	$q+r$
	0	s	t	$s+t$
	sum	$q+s$	$r+t$	p

Ordinal Attributes

- Order is important
  - Transfer rank into value
  - Freshman, Sophomore, Junior, Senior
  - 1,2,3,4

Mixed Attributes

All Together

$$\begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ 1 & 1 & 0 & \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & & & \\ 1.0 & 0 & & \\ 0.5 & 0.5 & 0 & \\ 0 & 1.0 & 0.5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & & & \\ 0.55 & 0 & & \\ 0.45 & 1.00 & 0 & \\ 0.40 & 0.14 & 0.86 & 0 \end{bmatrix}$$

Object Identifier	*test-1 (nominal)*	*test-2 (ordinal)*	*test-3 (numeric)*
1	code A	excellent	45
2	code B	fair	22
3	code C	good	64
4	code A	excellent	28

- d(3,1)?
  - $\frac{1(1)+1(0.5)+1(0.45)}{3}$

Cosine Similarity

- For vector data
- d1: I like to go to the store
- d2: I like the cubs, go cubs go

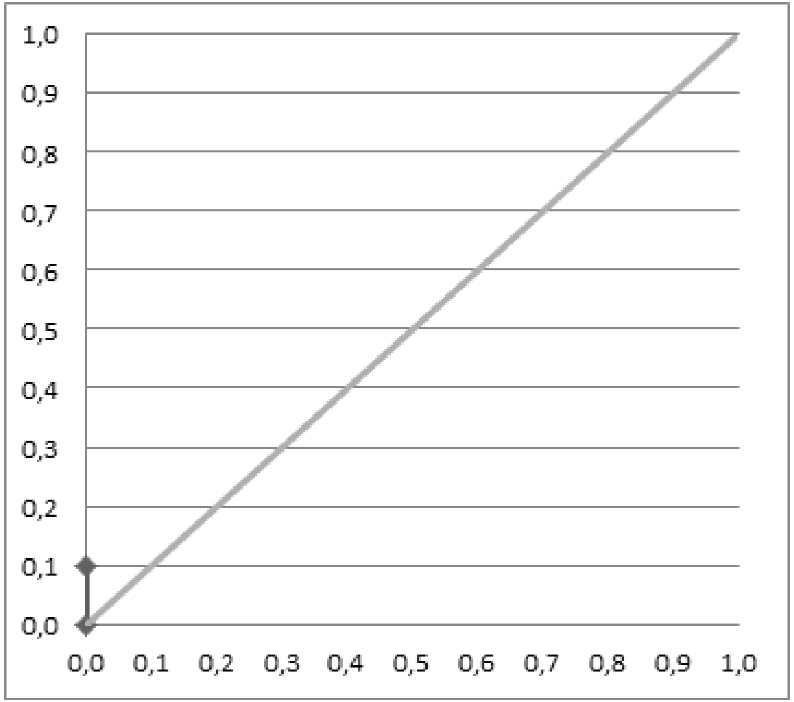
$$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1||\ ||d_2||,$$

Document	I	like	to	go	the	store	cubs
d1	1	1	2	1	1	1	0
d2	1	1	0	2	1	0	2

- cos(d1, d2)?
  - $\dfrac{1\cdot1+1\cdot1+2\cdot0+1\cdot2+1\cdot1+1\cdot0+0\cdot2}{\sqrt{1^2+1^2+2^2+1^2+1^2+1^2+0^2}\cdot\sqrt{1^2+1^2+0^2+2^2+1^2+0^2+2^2}}$

ROC

#	C	Score
1	P	0,9
2	P	0,8
3	N	0,7
4	P	0,6
5	P	0,55
6	P	0,54
7	N	0,53
8	N	0,52
9	P	0,51
10	N	0,505
11	P	0,4
12	N	0,39
13	P	0,38
14	N	0,37
15	N	0,36
16	N	0,35
17	P	0,34
18	N	0,33
19	P	0,3
20	N	0,1



http://mlwiki.org/index.php/ROC_Analysis

K-Means

- Demo 1: http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html
- Demo 2: https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

About Homework 3

- Due Date: Next Friday
- Be prepared (Refer to the doc for hw3)
  - Install Jupyter with Python 3.x(3.6 is preferred)
  - Download the cifar-10 dataset via get_datasets.sh
    - For windows user, there might be problems downloading the dataset
- Live demonstration for installation
- Hints on k-fold cross-validation and matrix-level operations involved in NN

Course Project Midterm Report

- https://docs.google.com/document/d/1xLeBU-n8nuMT6zhLyLL1NIGuu25SJU1OWl9eybdgjXg/edit?usp=sharing