Occupant-Centric Grid-Interactive Buildings

3. Machine Learning II

CE397

Spring 2024

Prof. Dr. Zoltan Nagy

Tentative Course Outline / Schedule

Week	Class	Topic	Guest Lecture
1	01/17	Introduction / Overview / Python	
2	01/24	Machine Learning I	
3	01/31	Machine Learning II	
4	02/07	Machine Learning III	Justin Hill (Southern)
5	02/14	Occupant Behavior Modeling	
6	02/21	Occupant Behavior Modeling	Tanya Barham (CEL)
7	02/28	Occupant Behavior Modeling	Jessica Granderson (LBNL)
8	03/06	Occupant Behavior Modeling	Hussain Kazmi (KU Leuven)
9	03/13	Spring Break	
10	03/20	Advanced Control & Calibration	Ankush Chakrabarty (MERL)
11	04/27	Calibration	Donghun Kim (LBNL)
12	04/03	Introduction to CityLearn	
13	04/10	Project Work	Siva Sankaranarayanan (EPRI)
14	04/17	Project work	
15	04/24	Project work	

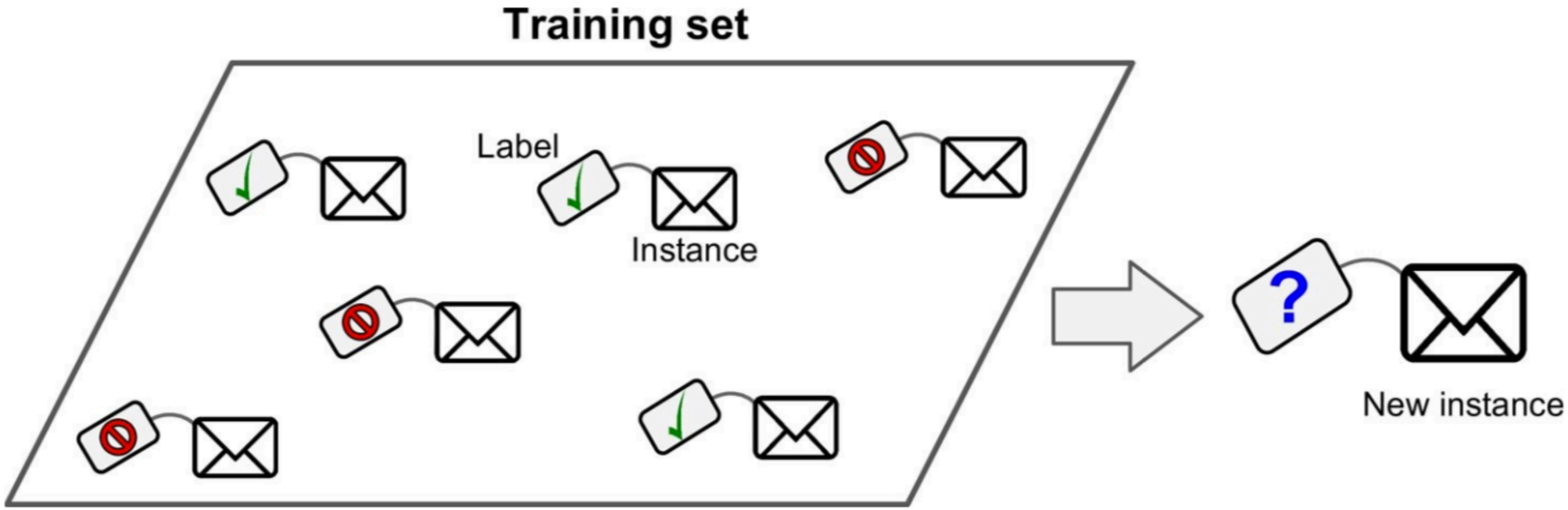
The Plan for Today

- Algorithms Overview
- Distance Metrics
- Tutorial & Homework: Database

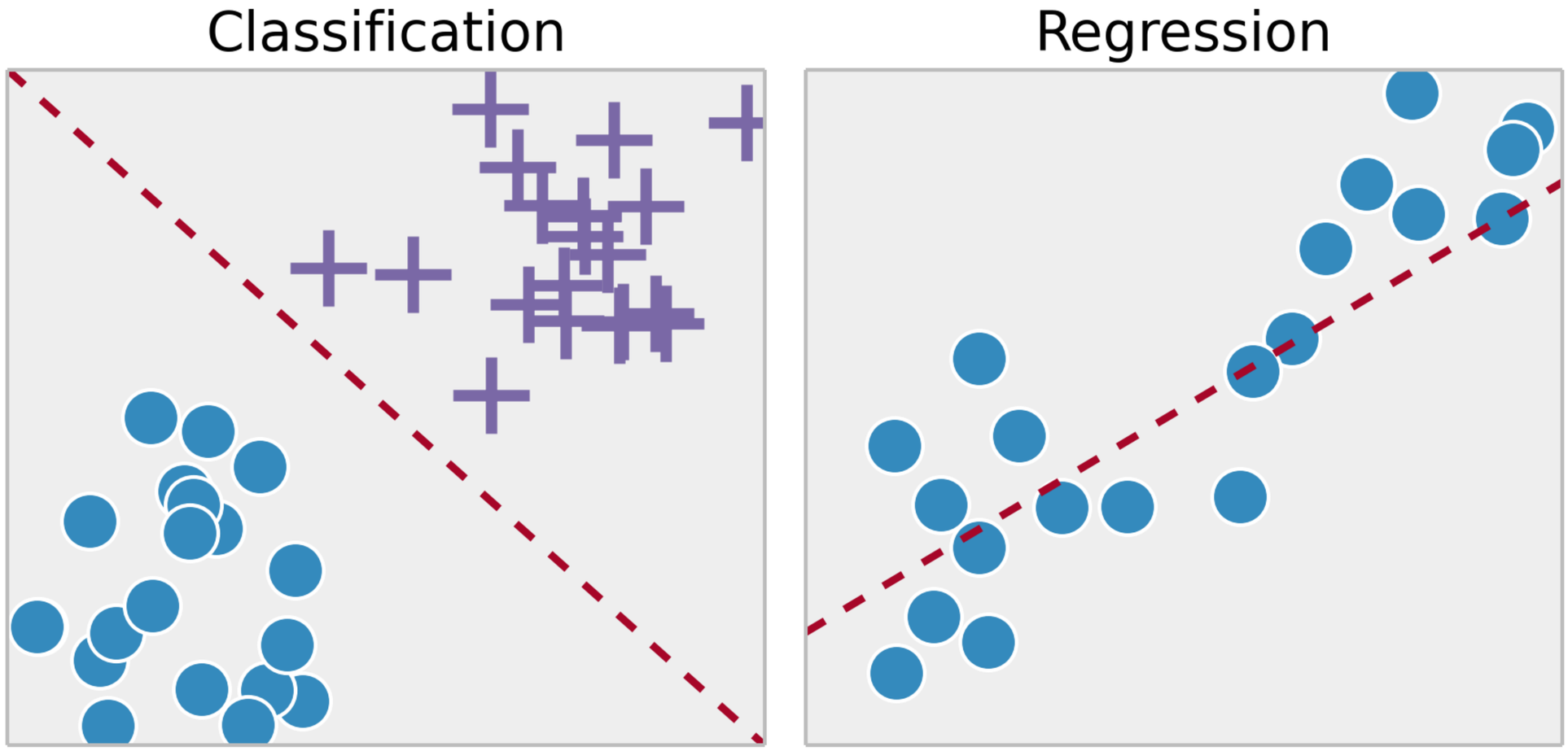
Recap: What are the three types of ML algorithms ?

Supervised Learning

Modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data.



Supervised Learning

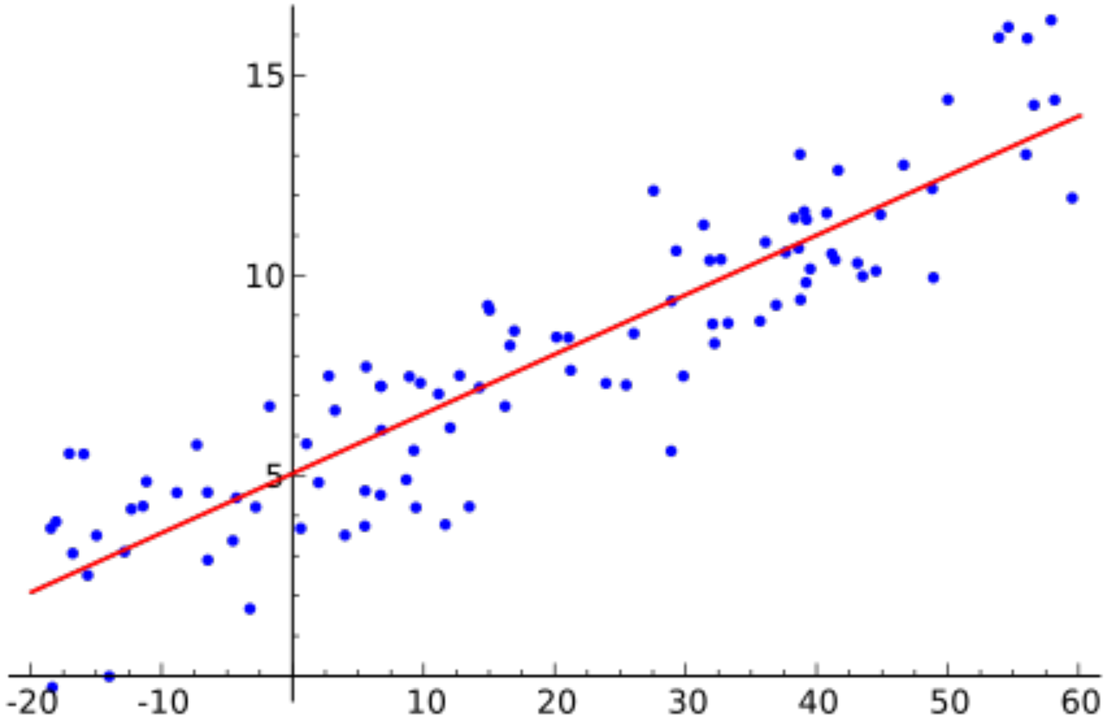


http://ipython-books.github.io/images/ml.png

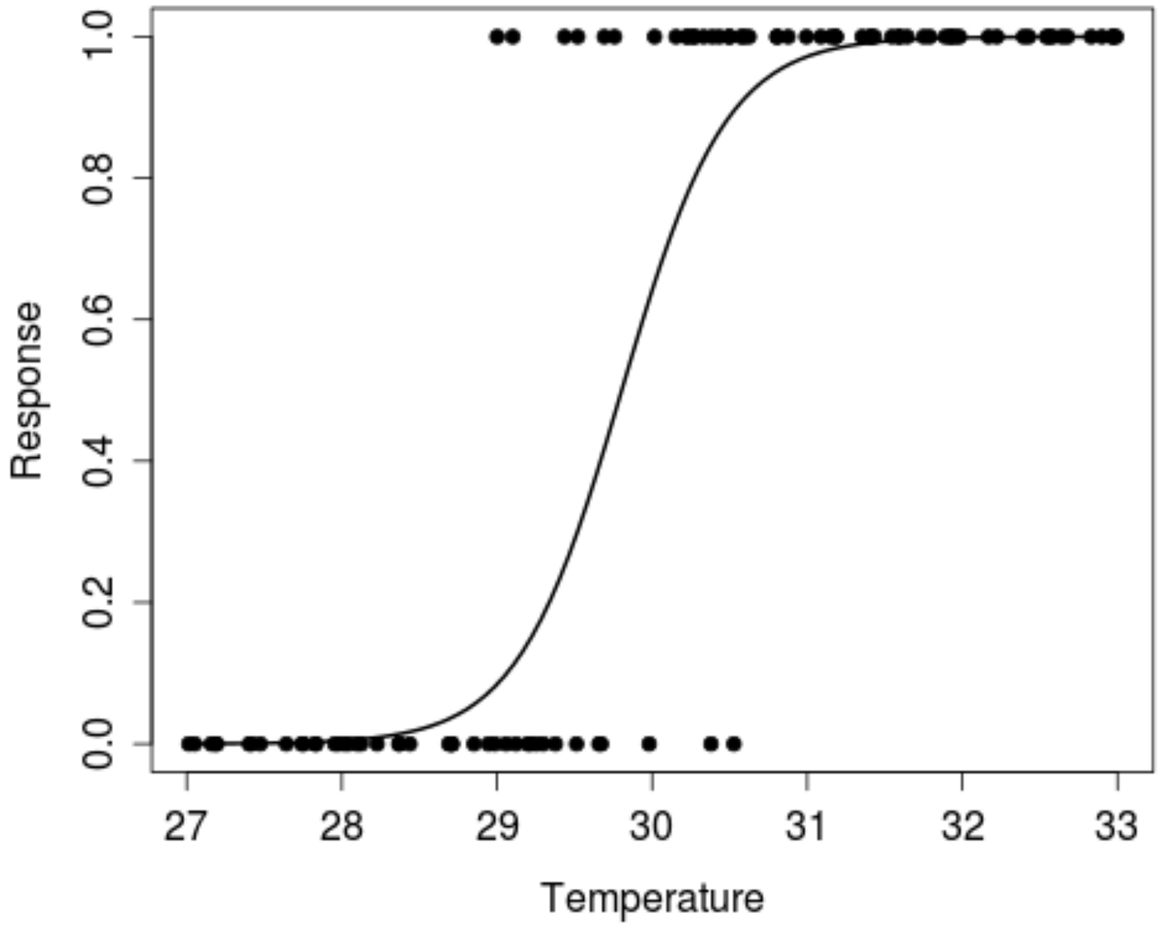
Supervised Learning - Algorithms

- Linear Regression
- Logistic Regression
- KNN (*k* nearest neighbor)
- Naïve Bayes Classifier
- Decision Trees
- Support Vector Machines
- Ensemble Methods

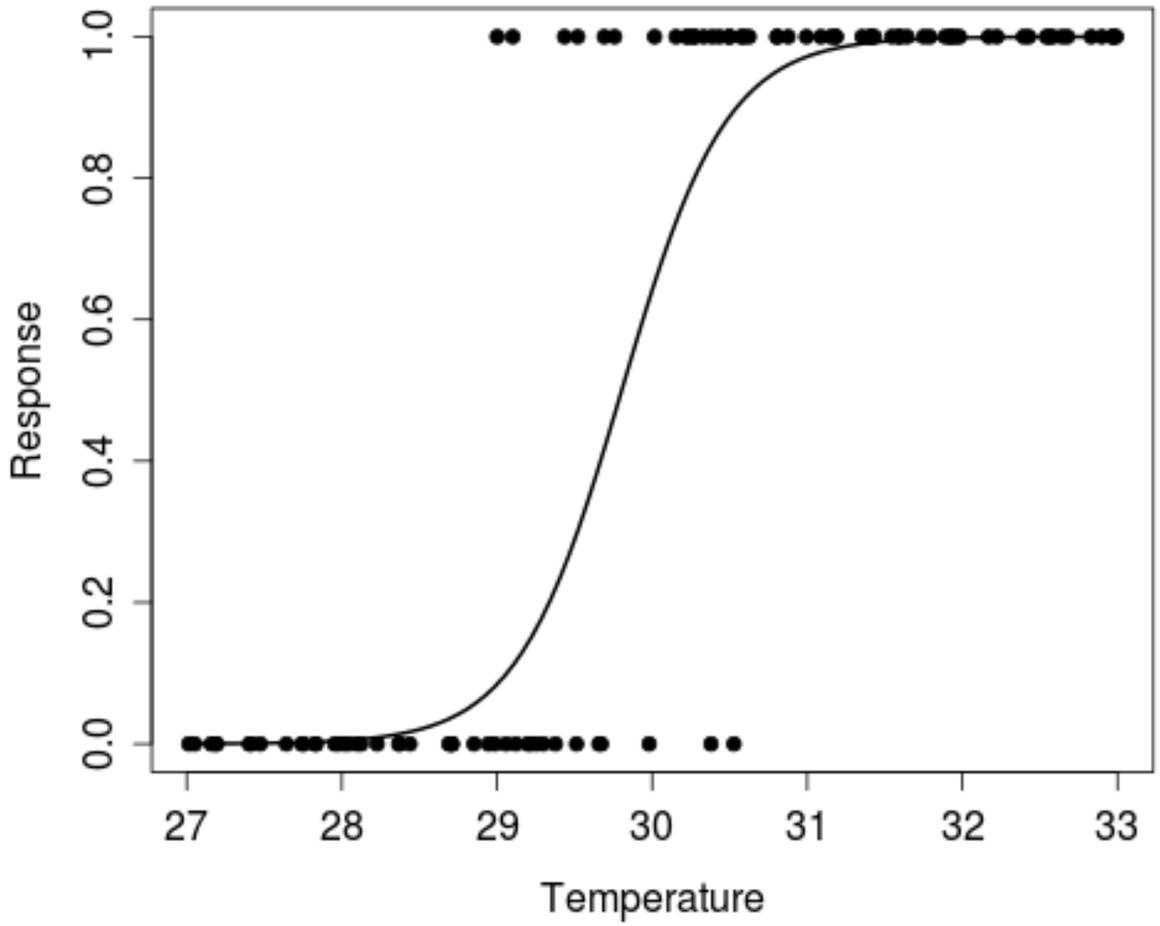
Linear Regression



Logistic Regression

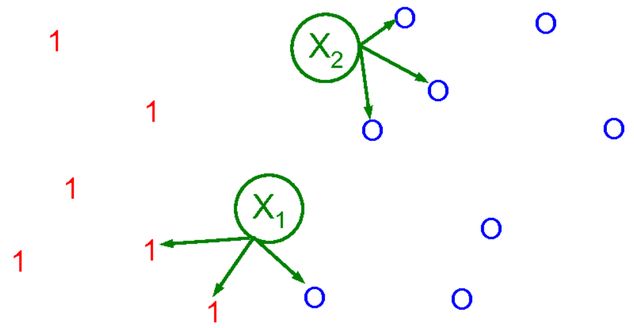


Logistic Regression



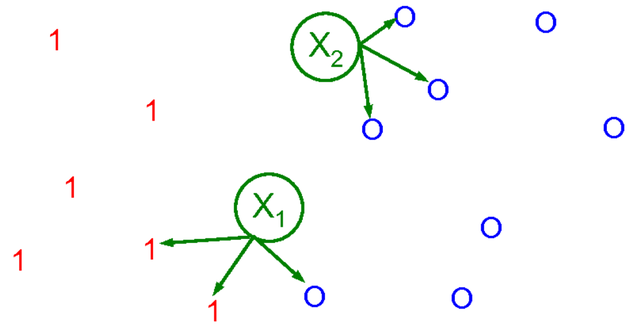
k Nearest Neighbors

Example 1 (k=3)

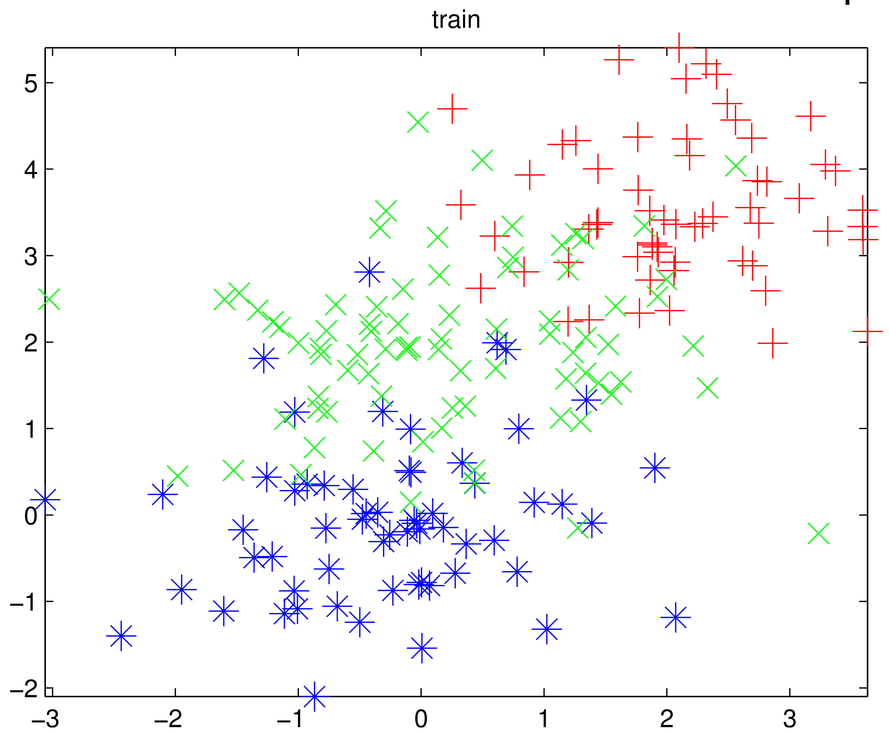


k Nearest Neighbors

Example 1 (k=3)

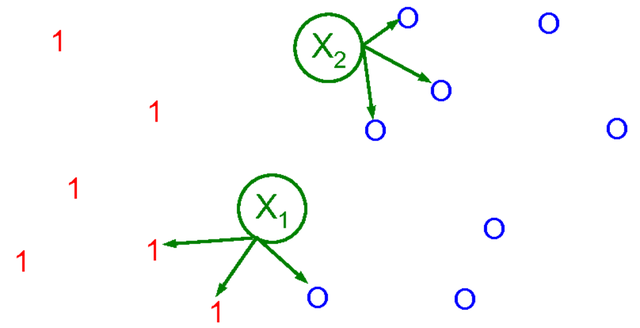


Example 2 (k=10)

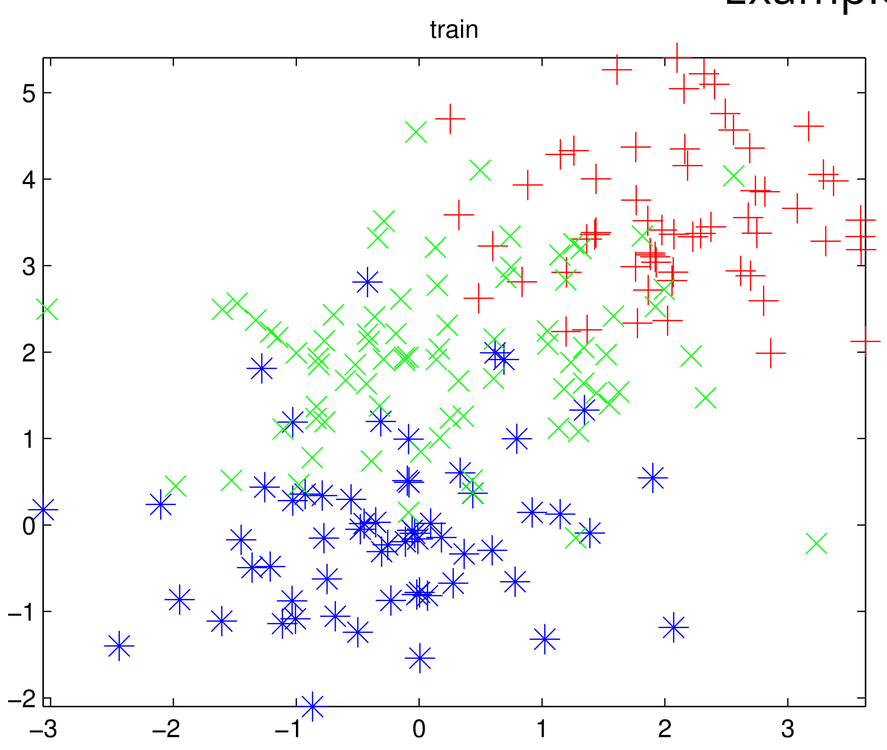


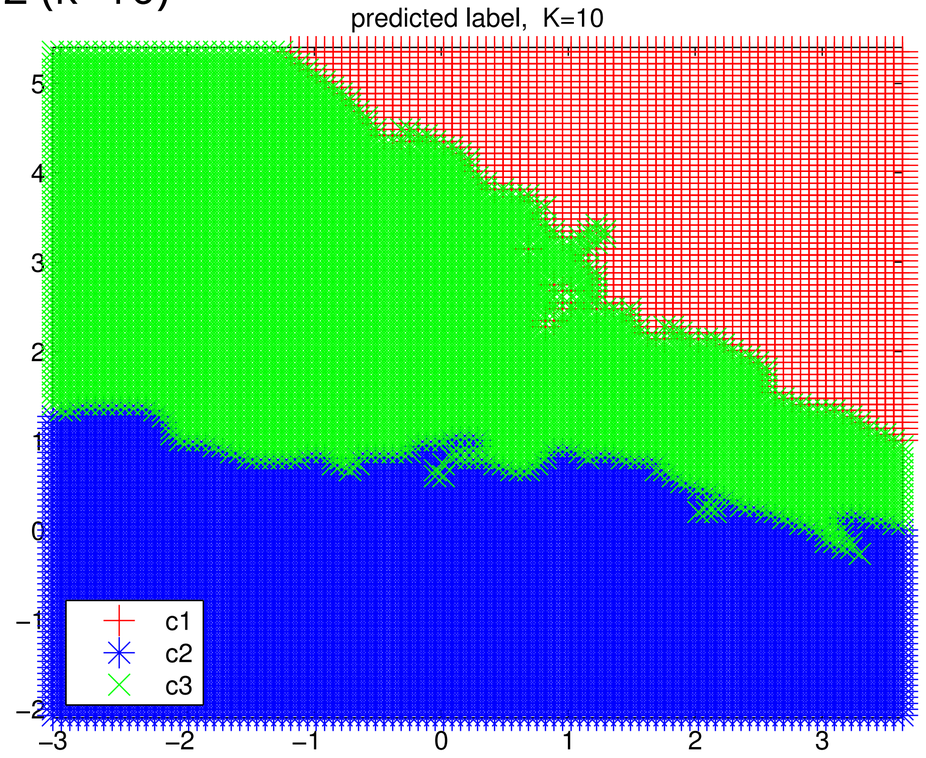
k Nearest Neighbors

Example 1 (k=3)

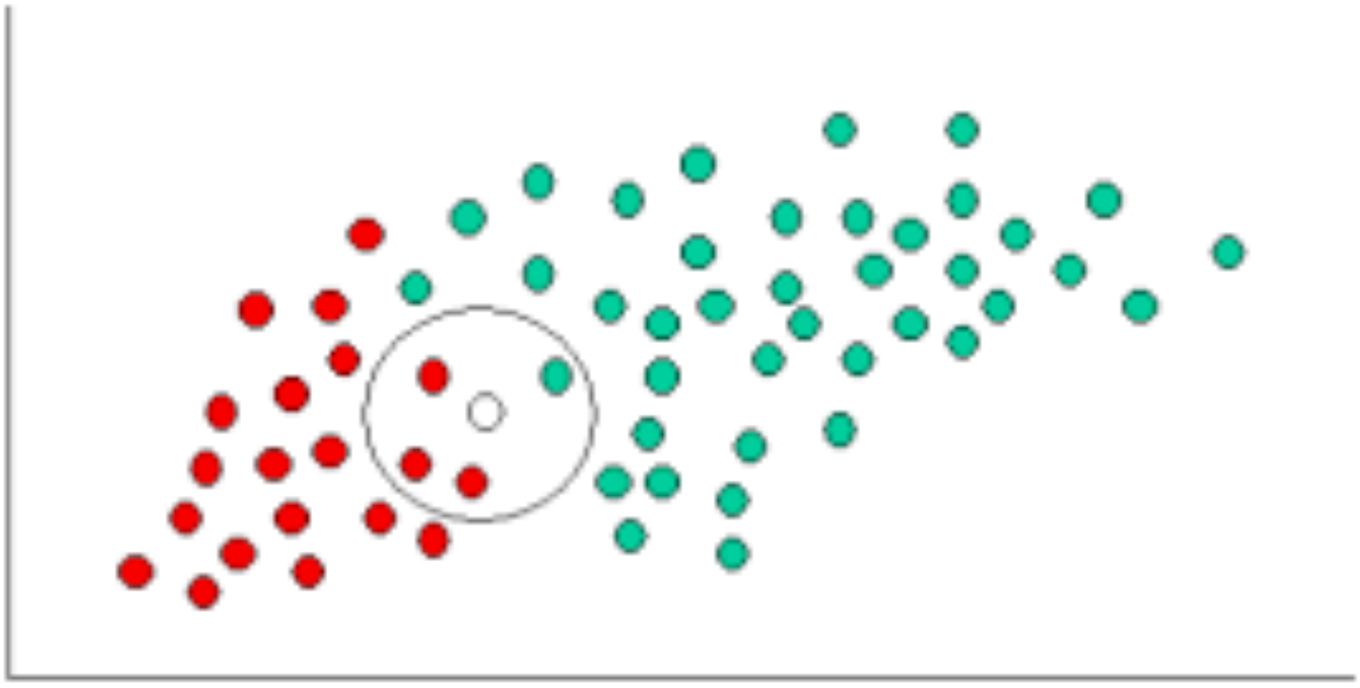


Example 2 (k=10)





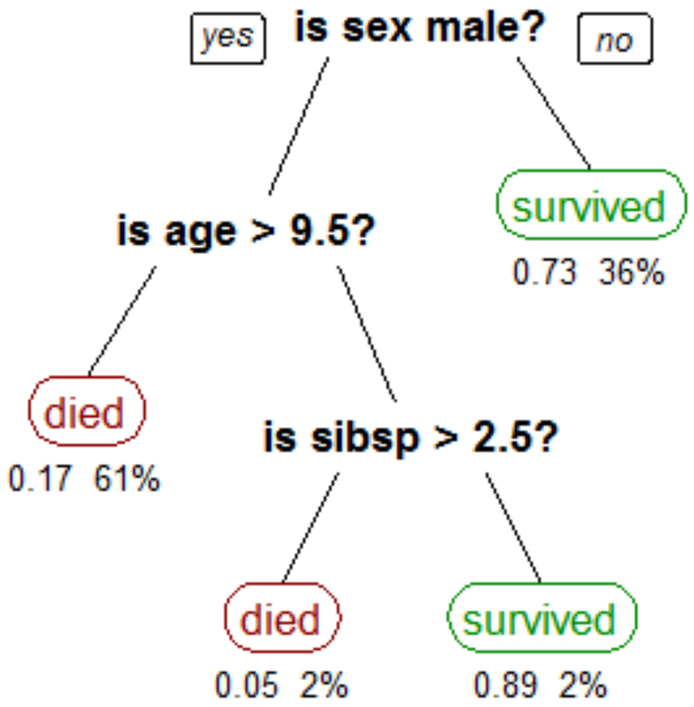
Naïve Bayes Classifier



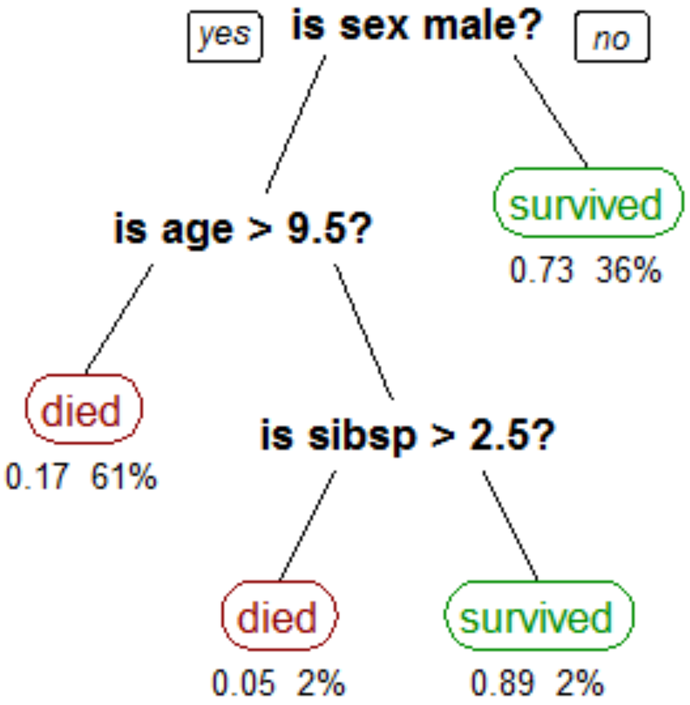
$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

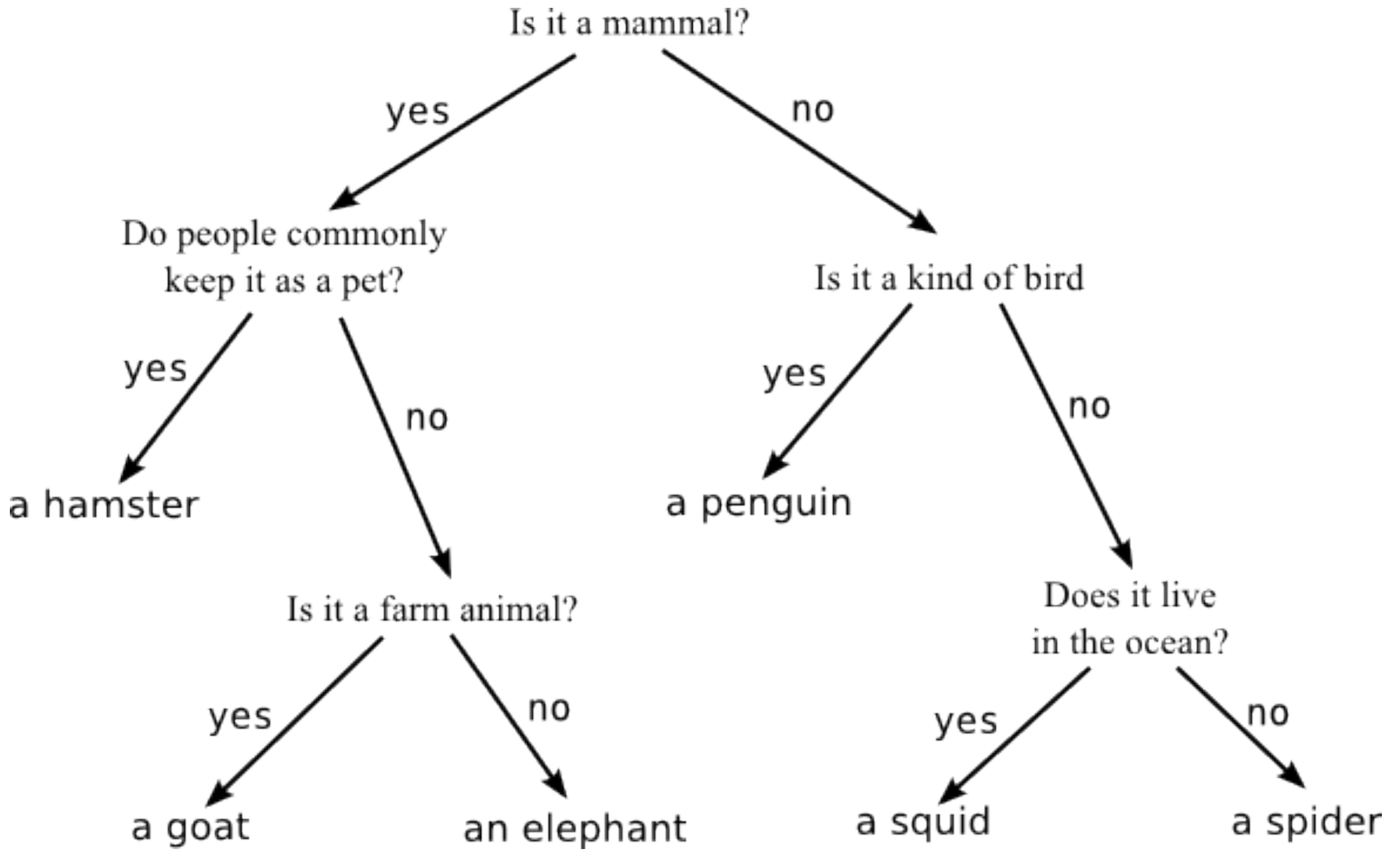
$$P(X \mid Y) = \frac{P(Y \mid X)P(X)}{P(Y)}$$

Decision Trees

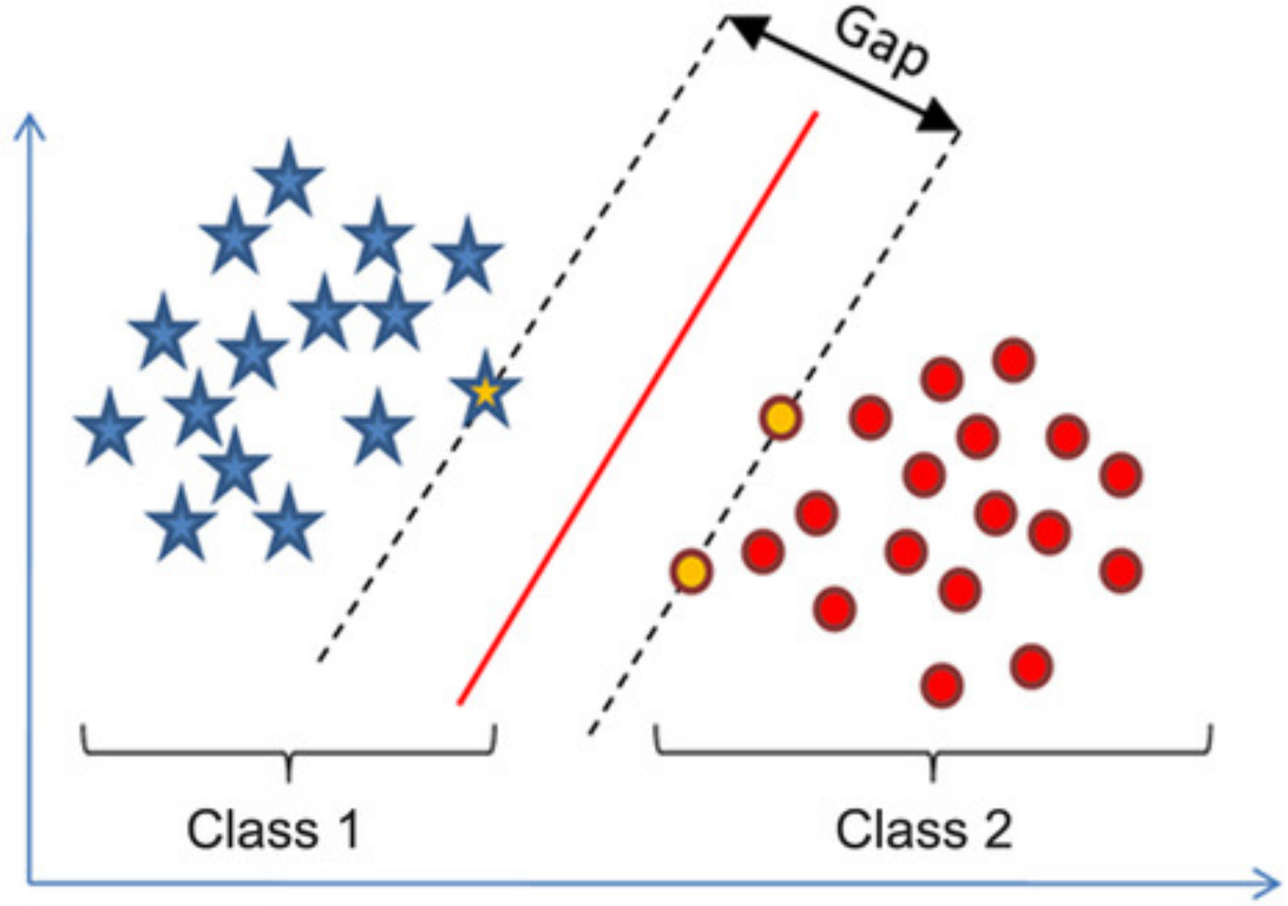


Decision Trees

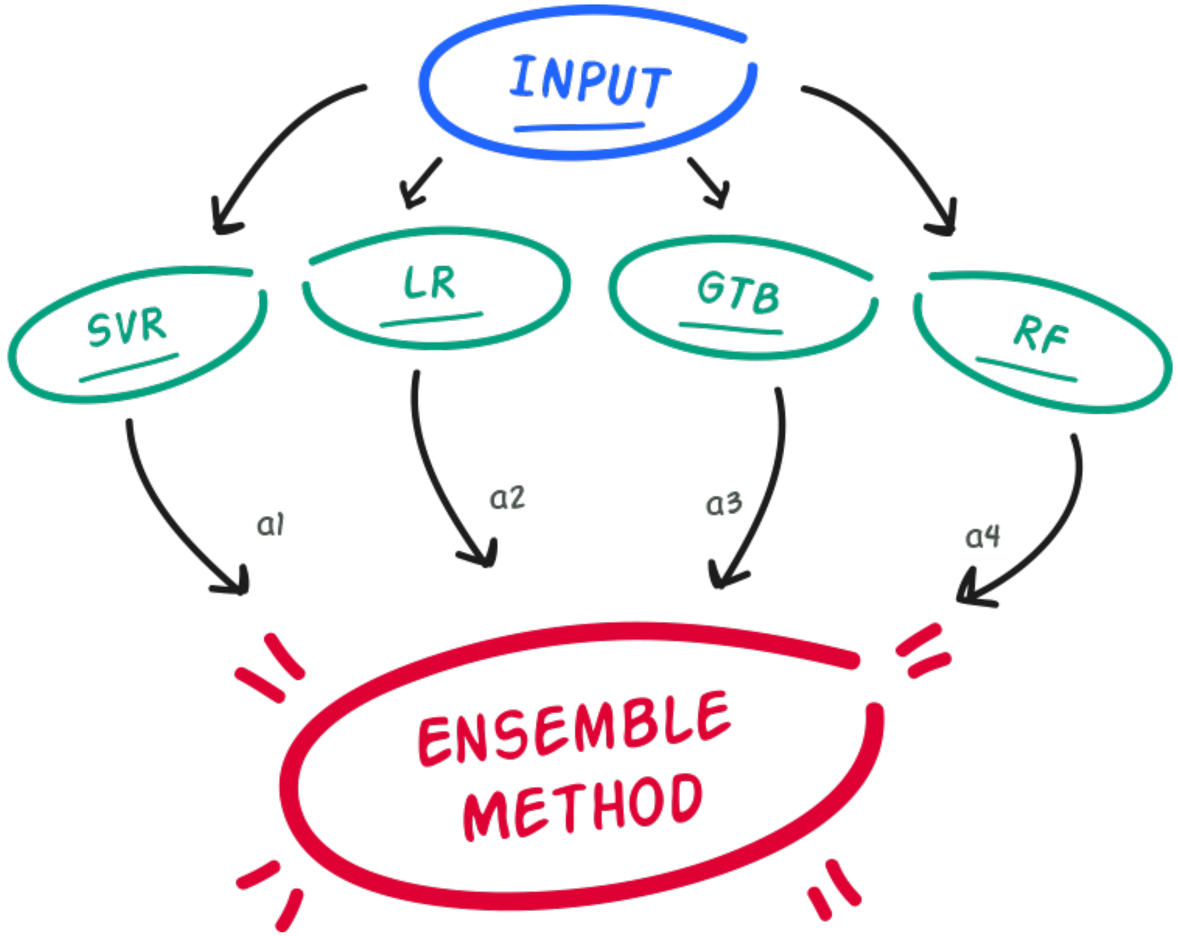




Support Vector Machines

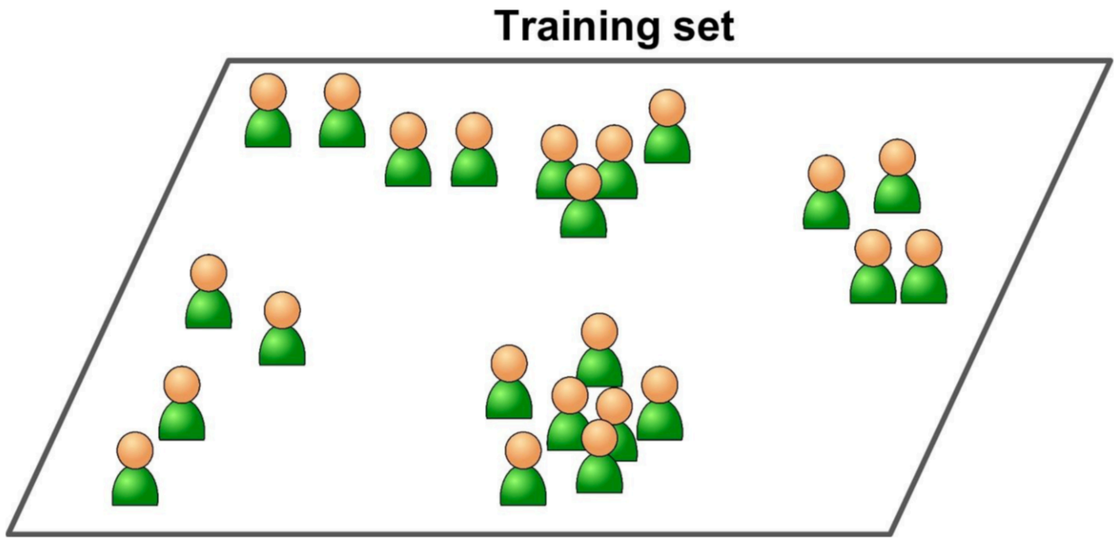


Ensemble Methods



Unsupervised Learning

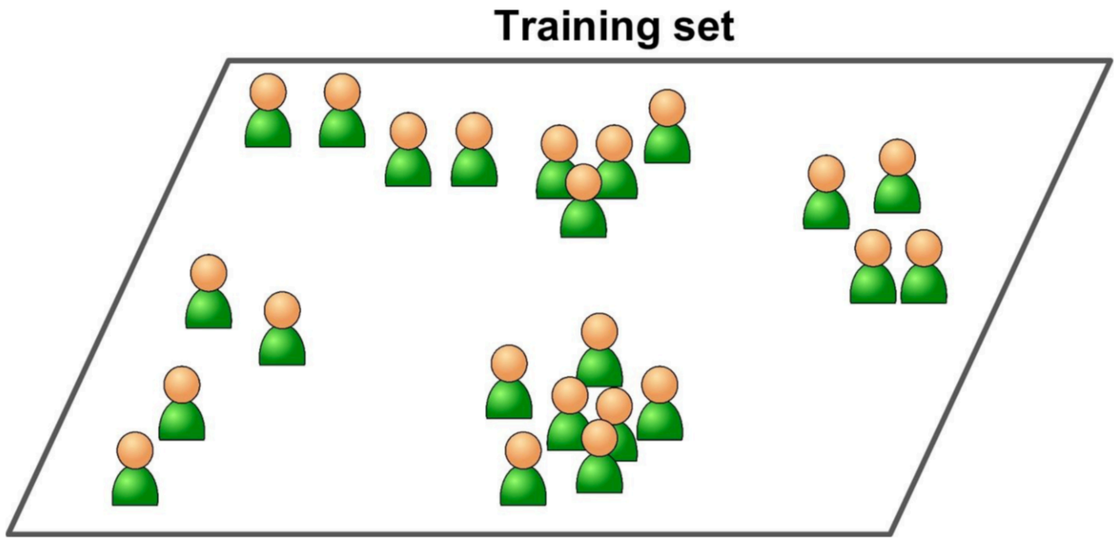
Modeling the features of a dataset *without* reference to any label, and is often described as “letting the dataset speak for itself.”

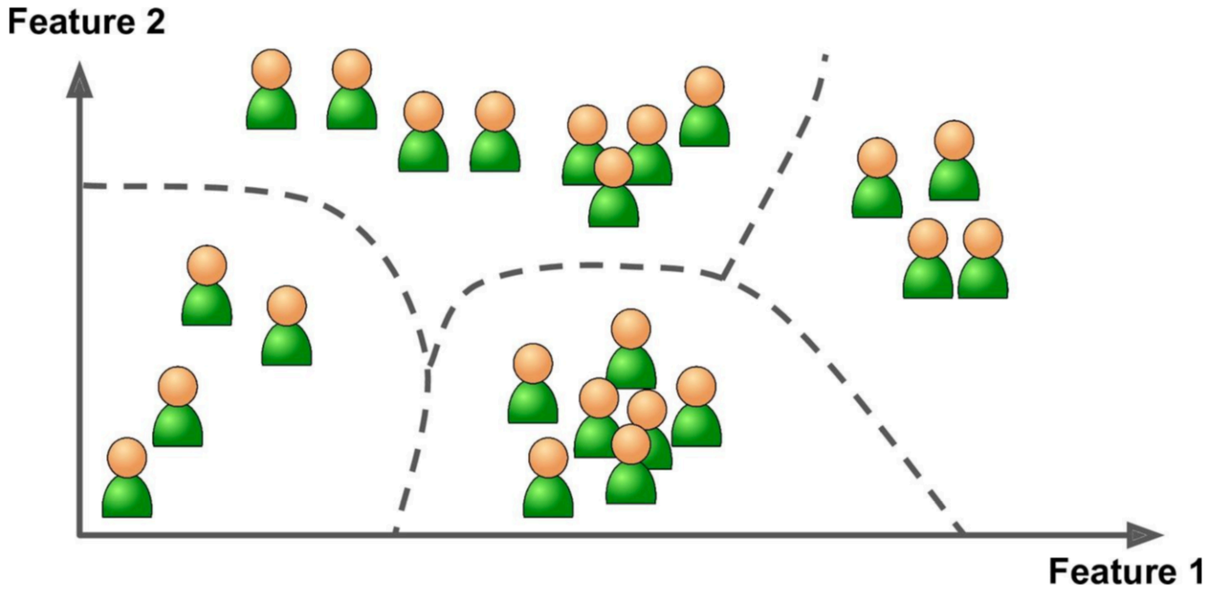


http://www.frankichamaki.com/data-driven-market-segmentation-more-effective-marketing-to-segments-using-ai/

Unsupervised Learning

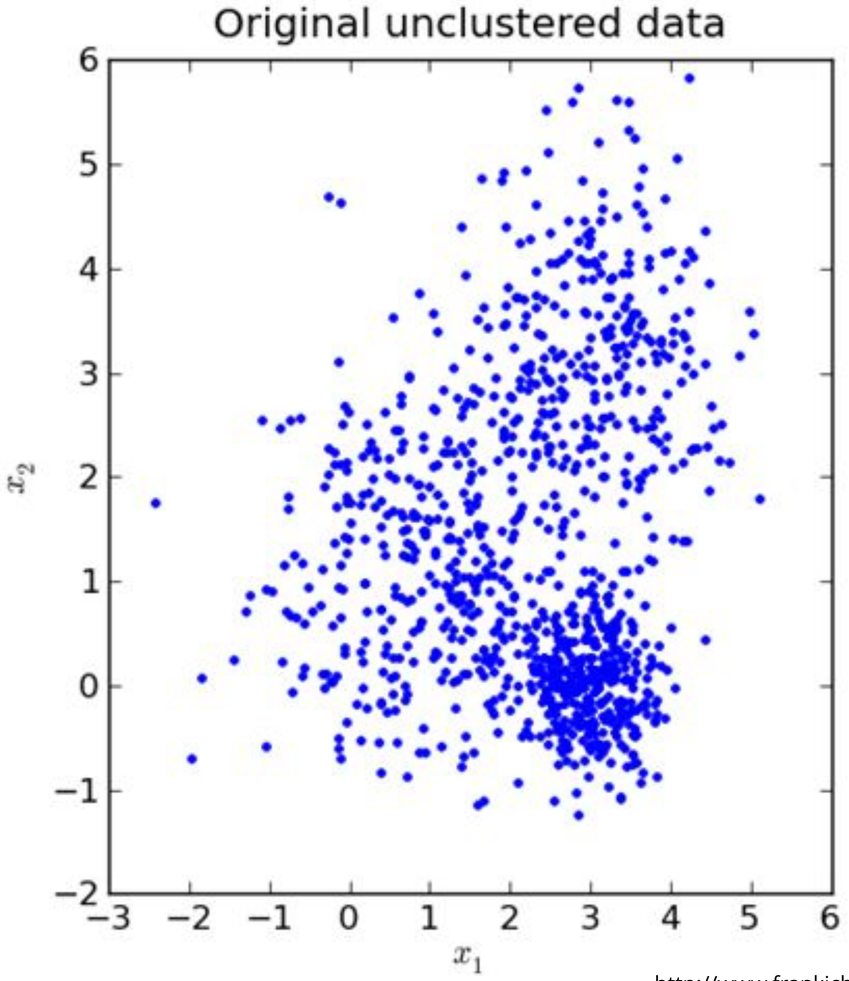
Modeling the features of a dataset *without* reference to any label, and is often described as “letting the dataset speak for itself.”





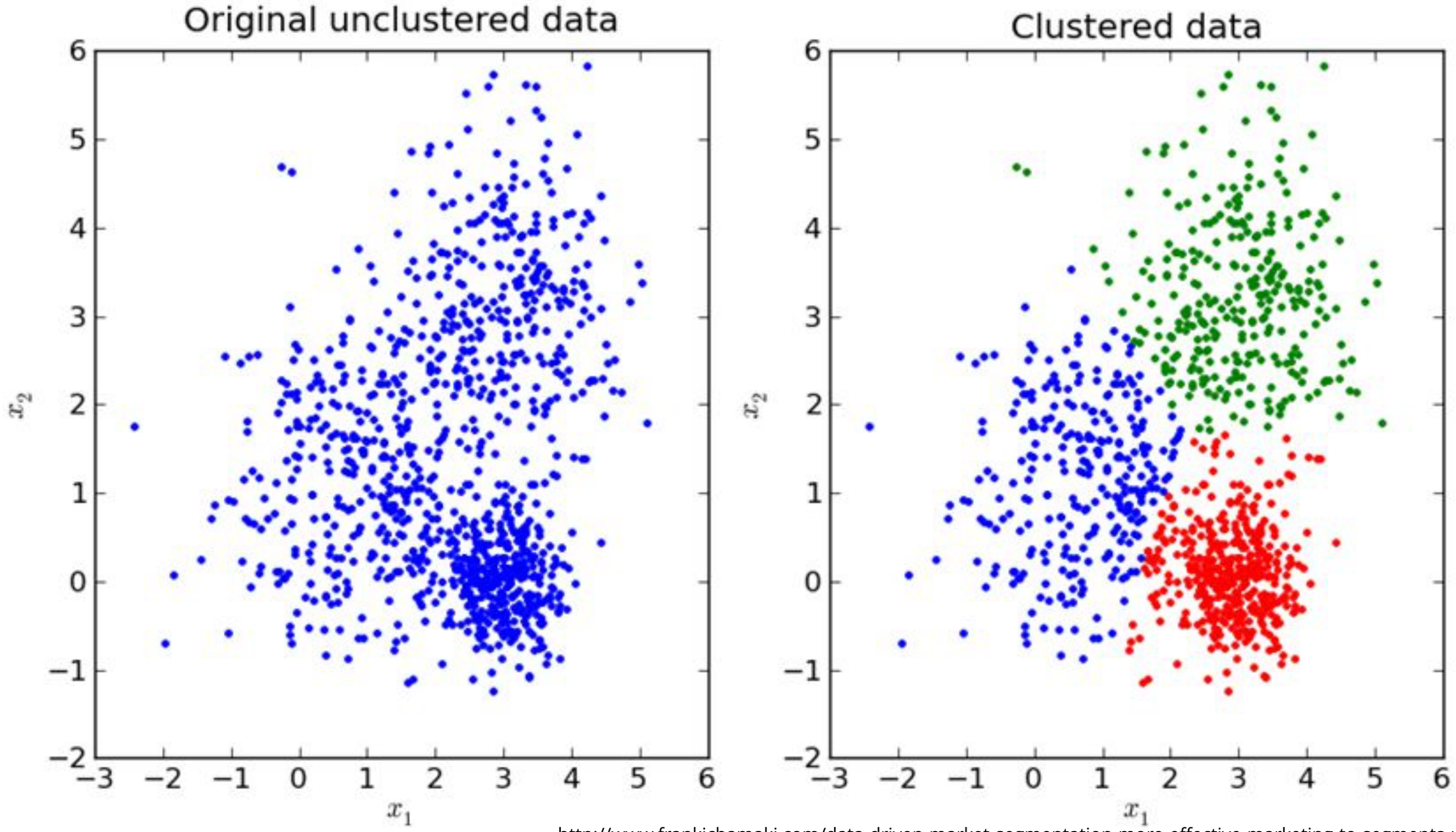
http://www.frankichamaki.com/data-driven-market-segmentation-more-effective-marketing-to-segments-using-ai/

Unsupervised Learning



http://www.frankichamaki.com/data-driven-market-segmentation-more-effective-marketing-to-segments-using-ai/

Unsupervised Learning



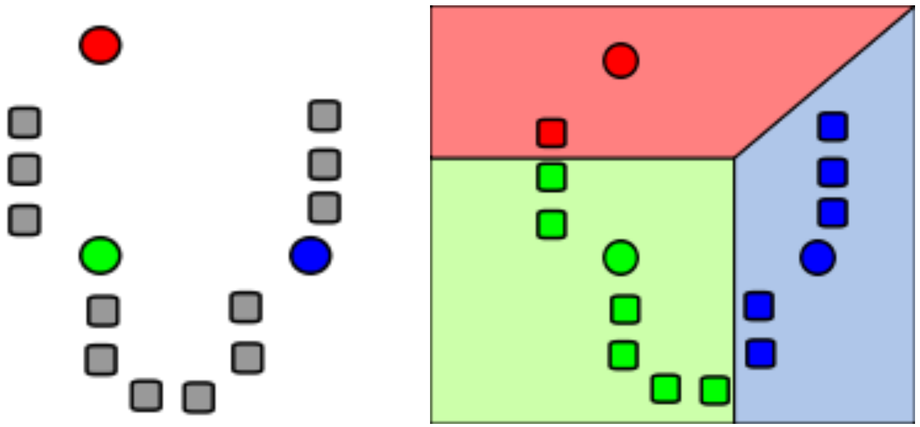
http://www.frankichamaki.com/data-driven-market-segmentation-more-effective-marketing-to-segments-using-ai/

Unsupervised Learning - Algorithms

- k-means clustering
- (self organizing map)

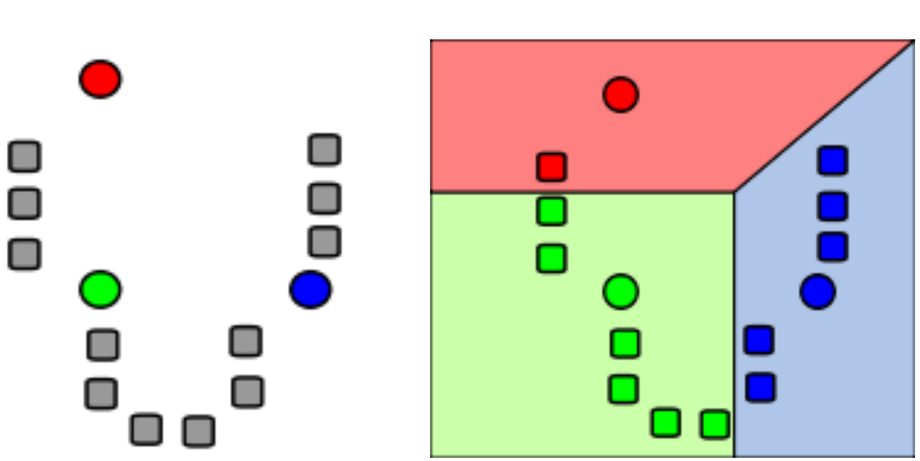
- When we cluster the observations of a data set, we seek to partition them into distinct groups so that the observations **within each group are quite similar to each other**, while observations in **different groups are quite different from each other**.

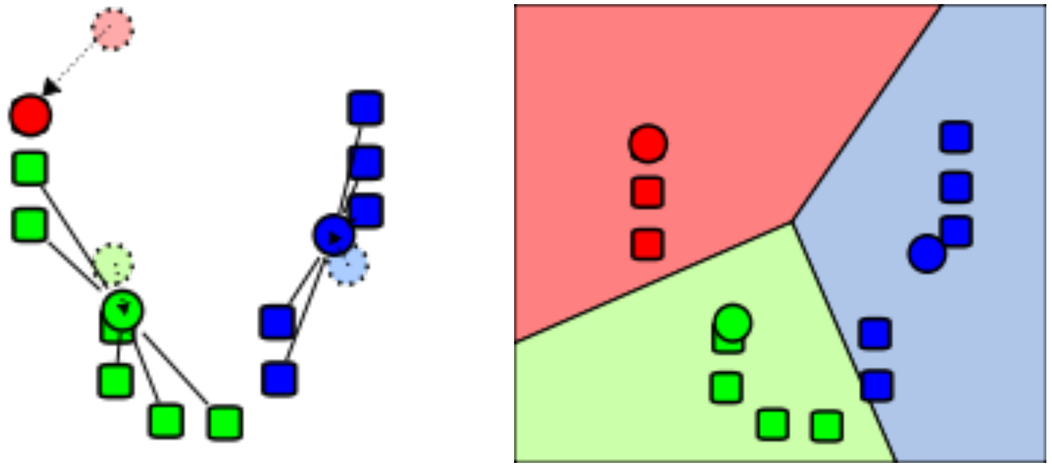
k-means clustering



http://ipython-books.github.io/images/ml.png

k-means clustering





http://ipython-books.github.io/images/ml.png

k-means clustering

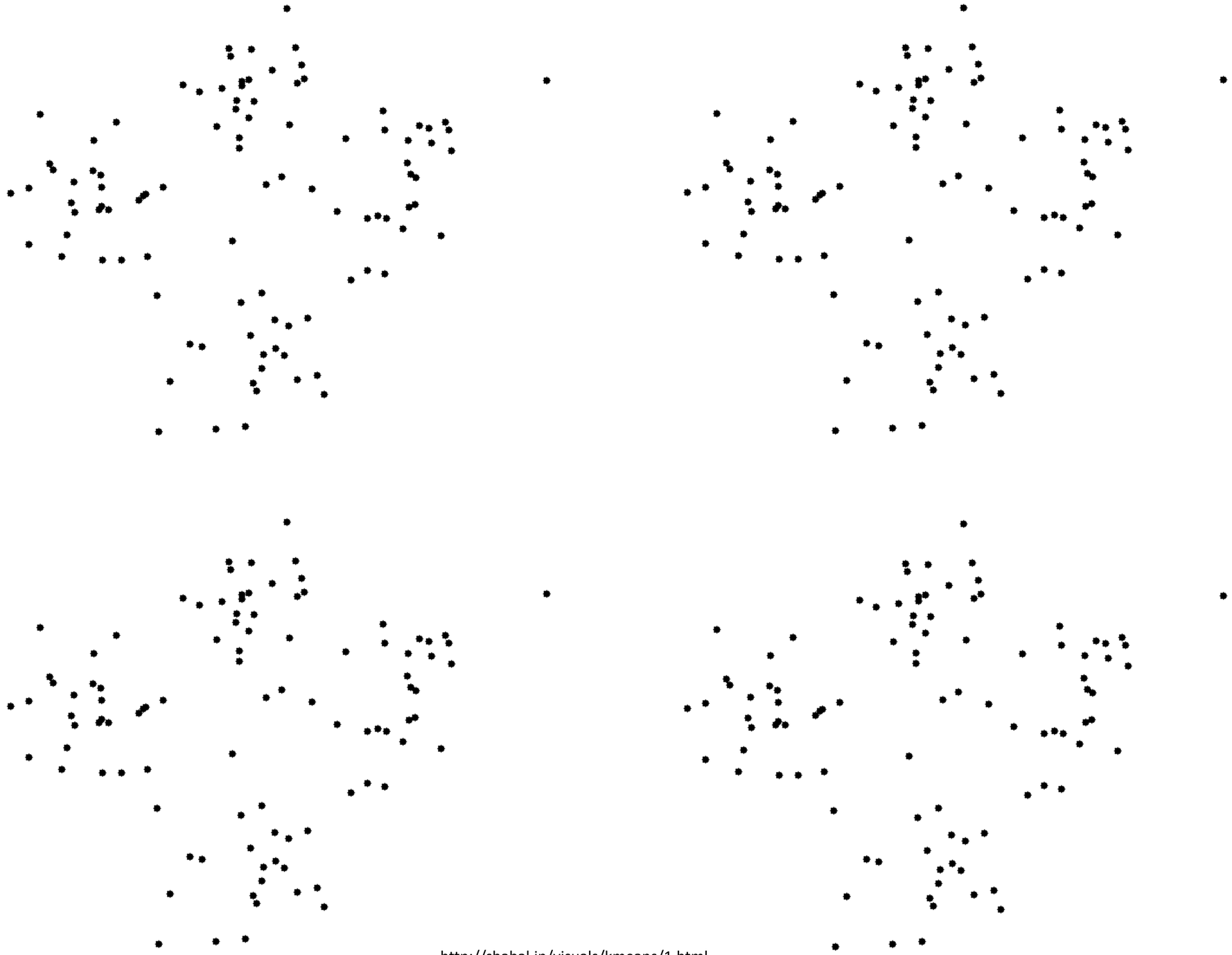
- Simple and elegant
- Partitions a data set into *k* distinct, non-overlapping clusters
- To perform K-means clustering
  1. We must first specify the desired number of clusters *k*
  2. then the algorithm will assign each observation to exactly one of the *k* clusters.

k-means clustering

Algorithm:

1. Pick *k* data points randomly as initial centers/centroids
2. Iterate until clusters stop changing
   (a) compute the cluster centroid as the means of the features of the data in the cluster
   (b) reassign data to the cluster whose centroid is closest.

http://shabal.in/visuals/kmeans/1.html



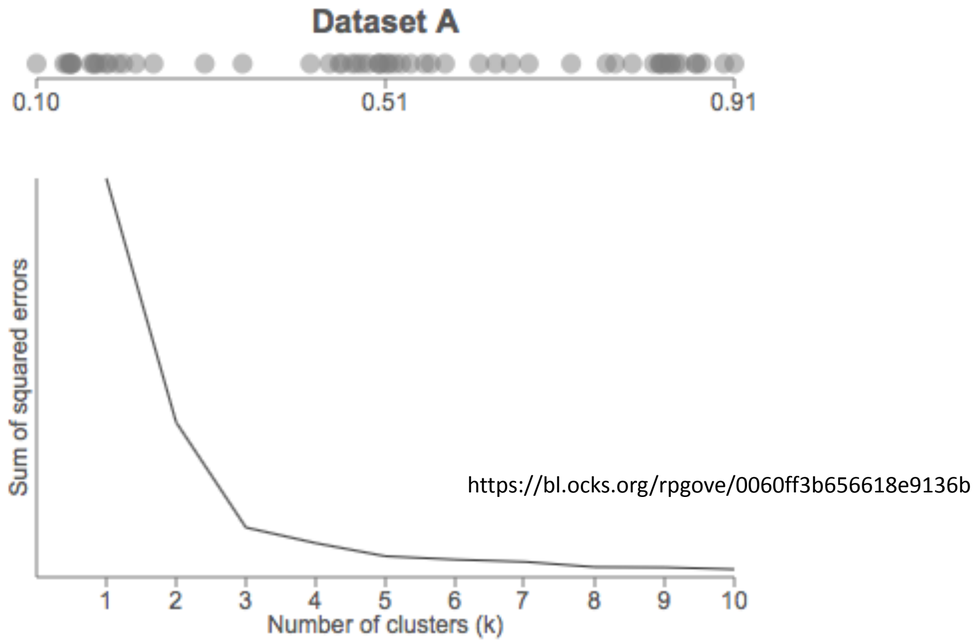
http://shabal.in/visuals/kmeans/1.html

How to choose *k*?

How to choose *k*?

- There is no single right answer
- Sometimes it's also problem dependent. ex: *Looking for the best way to group the data into 3 clusters*
- Some indices attempt to capture quality of clustering
  - 'Elbow'-method using sum of squared error

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$



https://bl.ocks.org/rpgove/0060ff3b656618e9136b

How to choose *k*?

- There is no single right answer
- Sometimes it's also problem dependent. ex: *Looking for the best way to group the data into 3 clusters*
- Some indices attempt to capture quality of clustering
  - 'Silhouette' score $$s = \frac{b - a}{max(a, b)}$$
    - a: mean distance between a sample and all other points in the **same cluster**
    - b: distance between a sample and all other points in the **next nearest cluster**
- **s →1:** clusters appropriate | **s →-1:** clusters not appropriate

How to choose *k*?

- There is no single right answer
- *Cohesion*: measures the similarity of profiles within a cluster

$$\text{Cohesion} = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - c_i||^2$$

where k is the number of clusters, C_i is cluster i, x is a point in cluster C_i and c_i is the centroid of cluster C_i.

How to choose *k*?

- There is no single right answer
- *Cohesion*: measures the similarity of profiles within a cluster
- *Separation*: measures how well dissimilar profiles are grouped into separate clusters

$$\text{Separation} = \sum_{i=1}^{k} |C_i| ||c_i - c||^2$$

where $|C_i|$ is the number of points in each cluster and c is the overall centroid of the data.

How to choose *k*?

- There is no single right answer
- *Cohesion*: measures the similarity of profiles within a cluster
- *Separation*: measures how well dissimilar profiles are grouped into separate clusters
- Calinski-Harabasz score: a trade- off between separation and cohesion by using both the average between- and within- cluster sum of squares

$$\text{CH Score} = \frac{\sum_{i=1}^{k} |C_i| ||c_i - c||^2 / (k-1)}{\sum_{i=1}^{k} \sum_{x \in C_i} ||x - c_i||^2 / (n-k)}$$

where n is the number of data points.

Break

similar implies a distance metric

A distance or metric function *d* is a real valued function that has following properties

- $d(\mathbf{x}_1,\mathbf{x}_2) \geq 0$
- $d(\mathbf{x}_1,\mathbf{x}_2) = 0 \iff \mathbf{x}_1 = \mathbf{x}_2$
- $d(\mathbf{x}_1,\mathbf{x}_2) = d(\mathbf{x}_2,\mathbf{x}_1)$
- $d(\mathbf{x}_1,\mathbf{x}_3) \leq d(\mathbf{x}_1,\mathbf{x}_2) + d(\mathbf{x}_2,\mathbf{x}_3)$

=> You can define your own distance metric as long as it has these properties

similar implies a distance metric

For continuous variables: **Minkowski distance**

$$d(\mathbf{x_1}, \mathbf{x_2}) = \left(\sum_i |x_{1,i} - x_{2,i}|^p \right)^{1/p}$$

p = 2: *Euclidean* distance

default, good for similar type data

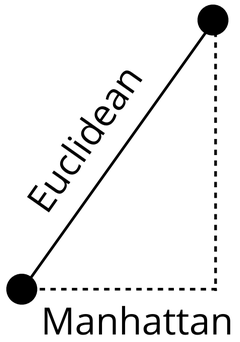
(ex: width, height, depth of parts)

p = 1: *Manhattan* distance

good for dissimilar type data

(ex: age, weight, gender)

p = ∞: *Chebyshev* distance



similar implies a distance metric

- Categorical variables: **Hamming** distance (overlap distance)
  - if categories are coded as 0 & 1(binary):

$$d_H(\mathbf{x_1}, \mathbf{x_2}) = \sum_i |x_{1,i} - x_{2,i}|$$

  - Example:

$\mathbf{x_1}$ = [0,0,1,1,1,0]

$\mathbf{x_2}$ = [1,0,1,1,0,1]

$d_H(\mathbf{x_1},\mathbf{x_2})$ = ?

Problem with distance computation

$\mathbf{x_1}$ = [43, 370]
$\mathbf{x_2}$ = [21, 300]
$\mathbf{x_q}$ = [22, 250]

$$
\begin{aligned}
d(\mathbf{x_1}, \mathbf{x_q}) &= \sqrt{(43-22)^2 + (370-250)^2} \\
&= \sqrt{21^2 + 120^2} \\
&\approx 121
\end{aligned}
$$

$$
\begin{aligned}
d(\mathbf{x_2}, \mathbf{x_q}) &= \sqrt{(21-22)^2 + (300-250)^2} \\
&= \sqrt{1^2 + 50^2} \\
&\approx 50
\end{aligned}
$$

Problem with distance computation

$\mathbf{x_1}$ = [43, 370]
$\mathbf{x_2}$ = [21, 300]
$\mathbf{x_q}$ = [22, 250]

$$d(\mathbf{x_1}, \mathbf{x_q}) = \sqrt{(43-22)^2 + (370-250)^2}$$
$$= \sqrt{21^2 + 120^2}$$
$$\approx 121$$

$$d(\mathbf{x_2}, \mathbf{x_q}) = \sqrt{(21-22)^2 + (300-250)^2}$$
$$= \sqrt{1^2 + 50^2}$$
$$\approx 50$$

The distance is dominated by one feature because it has much larger values

Normalization is necessary

Scaling

$$z_i = a + \frac{(x - x_{\min})(b - a)}{x_{\max} - x_{\min}}$$

$$x_i \in [x_{\min}, x_{\max}] \Rightarrow z_i \in [a, b]$$

Special case
a=0, b=1

Normalization is necessary

Scaling

$$z_i = a + \frac{(x - x_{\min})(b - a)}{x_{\max} - x_{\min}}$$

$$x_i \in [x_{\min}, x_{\max}] \Rightarrow z_i \in [a, b]$$

Special case
a=0, b=1

Standardization

- compute mean μ_i and standard deviation σ_i of the values in each dimension *i*
- use standardized variable $z_i = \frac{x_i - \mu_i}{\sigma_i}$

$$x_i \sim \mathcal{N}(\mu_i, \sigma_i) \Rightarrow z_i \sim \mathcal{N}(0, 1)$$

Other distance/similarity metrics

- Jaccard index (for binary features)

Who is q more similar to d_1 or d_2?

$\mathbf{q} = \langle$PROFILE:1, FAQ:0, HELP FORUM:1, NEWSLETTER:0, LIKED:0, $\rangle$

ID	Profile	FAQ	Help Forum	Newsletter	Liked	Signup
1	1	1	1	0	1	Yes
2	1	0	0	0	0	No

Other distance/similarity metrics

- Jaccard index (for binary features)

Who is q more similar to d_1 or d_2?

$\mathbf{q} = \langle$PROFILE:1, FAQ:0, HELP FORUM:1, NEWSLETTER:0, LIKED:0, $\rangle$

ID	Profile	FAQ	Help Forum	Newsletter	Liked	Signup
1	1	1	1	0	1	Yes
2	1	0	0	0	0	No

		q	
		Pres.	Abs.
d_1	Pres.	CP=2	PA=0
	Abs.	AP=2	CA=1

		q	
		Pres.	Abs.
d_2	Pres.	CP=1	PA=1
	Abs.	AP=0	CA=3

Table: The similarity between the current trial user, **q**, and the two users in the dataset, $\mathbf{d}_1$ and $\mathbf{d}_2$, in terms of co-presence (CP), co-absence (CA), presence-absence (PA), and absence-presence (AP).

Other distance/similarity metrics

- Jaccard index (for binary features)

Jaccard

$$sim_J(\mathbf{q}, \mathbf{d}) = \frac{CP(\mathbf{q}, \mathbf{d})}{CP(\mathbf{q}, \mathbf{d}) + PA(\mathbf{q}, \mathbf{d}) + AP(\mathbf{q}, \mathbf{d})} \quad (14)$$

(Jaccard ignores co-absence)

⟨PROFILE:1, FAQ:0, HELP FORUM:1, NEWSLETTER:0, LIKED:0, ⟩

		q Pres.	Abs.
$\mathbf{d}_1$	Pres.	CP=2	PA=0
	Abs.	AP=2	CA=1

		q Pres.	Abs.
$\mathbf{d}_2$	Pres.	CP=1	PA=1
	Abs.	AP=0	CA=3

Other distance/similarity metrics

- Jaccard index (for binary features)

Jaccard

$$sim_J(\mathbf{q}, \mathbf{d}) = \frac{CP(\mathbf{q}, \mathbf{d})}{CP(\mathbf{q}, \mathbf{d}) + PA(\mathbf{q}, \mathbf{d}) + AP(\mathbf{q}, \mathbf{d})} \quad (14)$$

⟨PROFILE:1, FAQ:0, HELP FORUM:1, NEWSLETTER:0, LIKED:0, ⟩

		q Pres.	**q** Abs.
$\mathbf{d}_1$	Pres.	CP=2	PA=0
	Abs.	AP=2	CA=1

		q Pres.	**q** Abs.
$\mathbf{d}_2$	Pres.	CP=1	PA=1
	Abs.	AP=0	CA=3

$$sim_J(\mathbf{q}, \mathbf{d}_1) = \frac{2}{4} = 0.5$$

$$sim_J(\mathbf{q}, \mathbf{d}_2) = \frac{1}{2} = 0.5$$

Other distance/similarity metrics

- cosine similarity (continuous features)

- Cosine similarity between two instances is the cosine of the inner angle between the two vectors that extend from the origin to each instance.

Cosine

$$sim_{COSINE}(\mathbf{a}, \mathbf{b}) = \frac{(\mathbf{a}[1] \times \mathbf{b}[1]) + \cdots + (\mathbf{a}[m] \times \mathbf{b}[m])}{\sqrt{\sum_{i=1}^{m} \mathbf{a}[i]^2} \times \sqrt{\sum_{i=1}^{m} \mathbf{b}[i]^2}}$$

Other distance/similarity metrics

- cosine similarity (continuous features)

$$\mathbf{d}_1 = \langle \text{SMS} = 97, \text{VOICE} = 21 \rangle$$

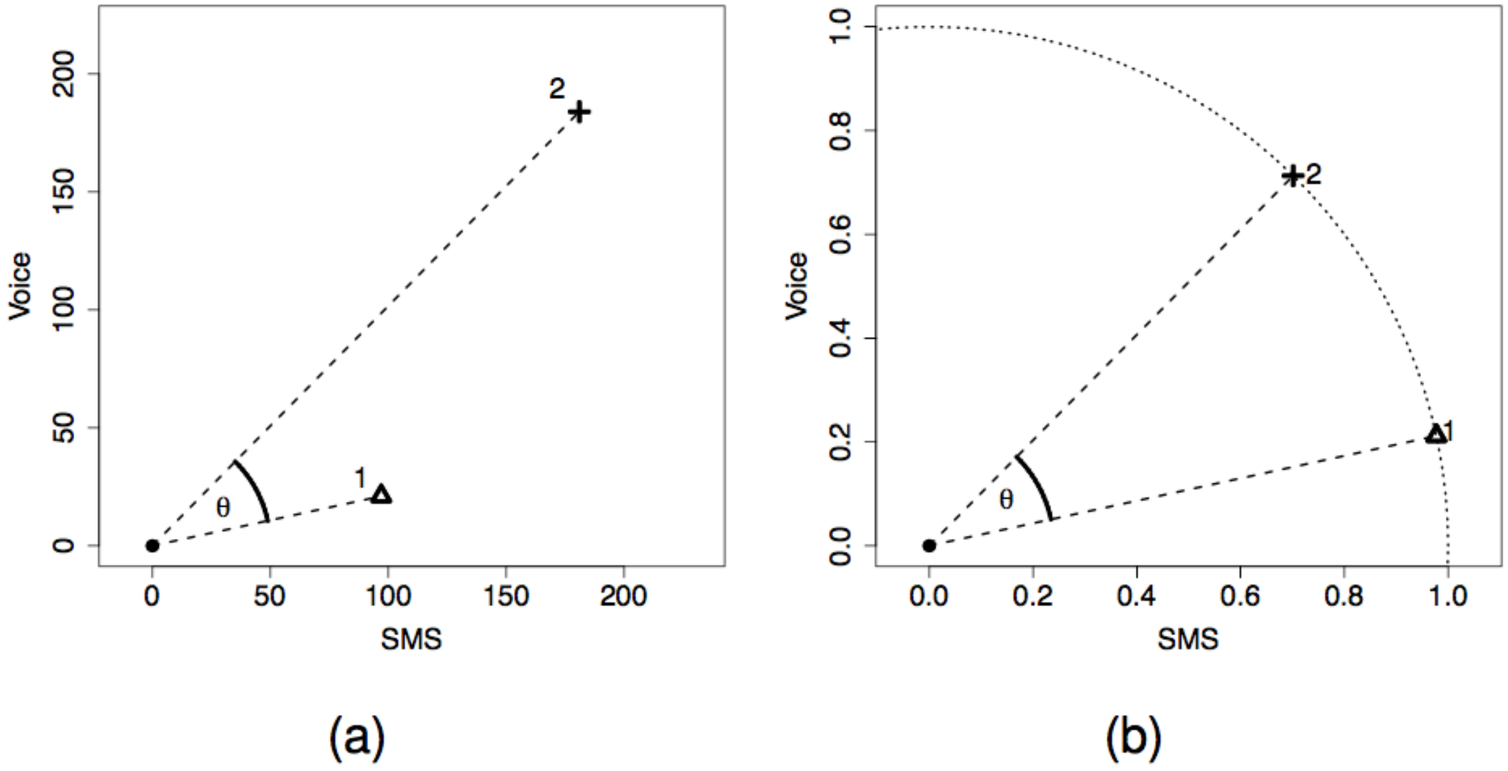
$$\mathbf{d}_2 = \langle \text{SMS} = 181, \text{VOICE} = 184 \rangle$$

$$sim_{COSINE}(\mathbf{a}, \mathbf{b}) = \frac{\sum_{i=1}^{m} (\mathbf{a}[i] \times \mathbf{b}[i])}{\sqrt{\sum_{i=1}^{m} \mathbf{a}[i]^2} \times \sqrt{\sum_{i=1}^{m} \mathbf{b}[i]^2}}$$

$$sim_{COSINE}(\mathbf{d}_1, \mathbf{d}_2) = \frac{(97 \times 181) + (21 \times 184)}{\sqrt{97^2 + 21^2} \times \sqrt{181^2 + 184^2}} = 0.8362$$

Other distance/similarity metrics

- cosine similarity (continuous features)



Curse of dimensionality

- kNN & k-means work well in low dimensional space with plenty of data
- for high dimensions (15-20 and above), even the nearest neighbor is very far and not informative

Python activity

- Tutorial on sqlite database
- Homework 2 on smart meter data

Questions?