

Programming with Data Bootcamp: Lecture 5

Slides courtesy of Sam Madden /
Tim Kraska (6.S079)

Key ideas:

Traditional ML

- Clustering
- Dim. Reduction
- Classification
- Regression

<http://dsg.csail.mit.edu/6.S079/>



MACHINE LEARNING PROBLEMS

(Boosted-) Decision Trees

K-Means
Agglomerative clustering
DBScan

Supervised Learning

Unsupervised Learning

Discrete

classification or categorization

clustering

Continuous

regression

dimensionality reduction

(Boosted-) Decision Trees

PCA

MACHINE LEARNING PROBLEMS

(Boosted-) Decision Trees

K-Means
Agglomerative clustering
DBScan

Supervised Learning

Unsupervised Learning

Discrete

classification or categorization

clustering

Continuous

regression

dimensionality reduction

(Boosted-) Decision Trees

PCA

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

K-MEANS ALGORITHM

Select K random data points $\{s_1, s_2, \dots, s_K\}$ as centroids c_j .

Until clustering converges or other stopping criterion {

For each data point x_i :

Assign x_i to the closes centroid such that

$dist(x_i, c_j)$ is minimal.

For each cluster c_j , update the centroids

$$c_j = \mu(c_j)$$

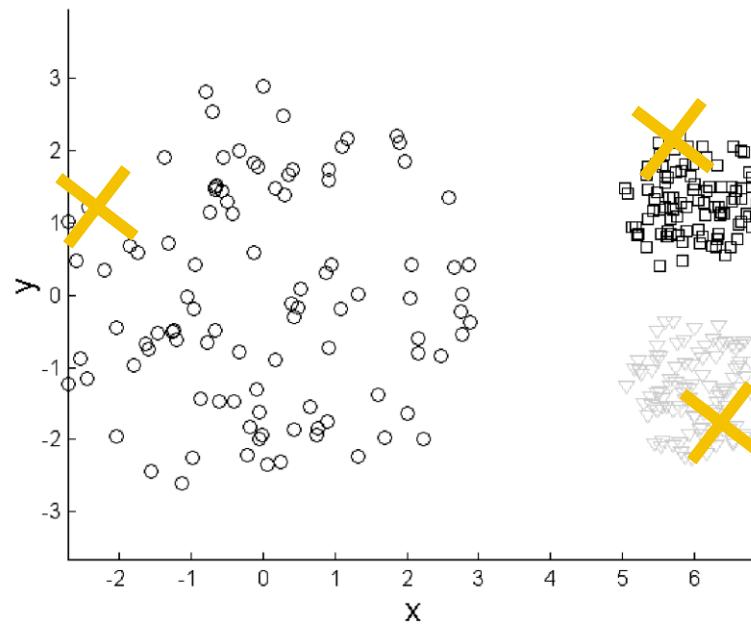
}

TERMINATION CONDITIONS

Several possibilities, e.g.,

- A fixed number of iterations.
- Partition unchanged.
- Centroid positions don't change.

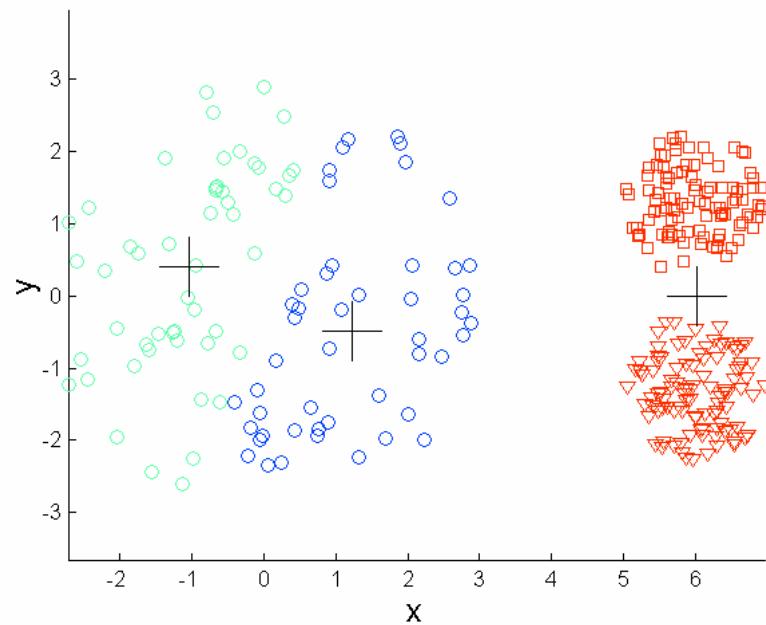
CLASS TASK



Original Points

What cluster will you get with the yellow centroids?

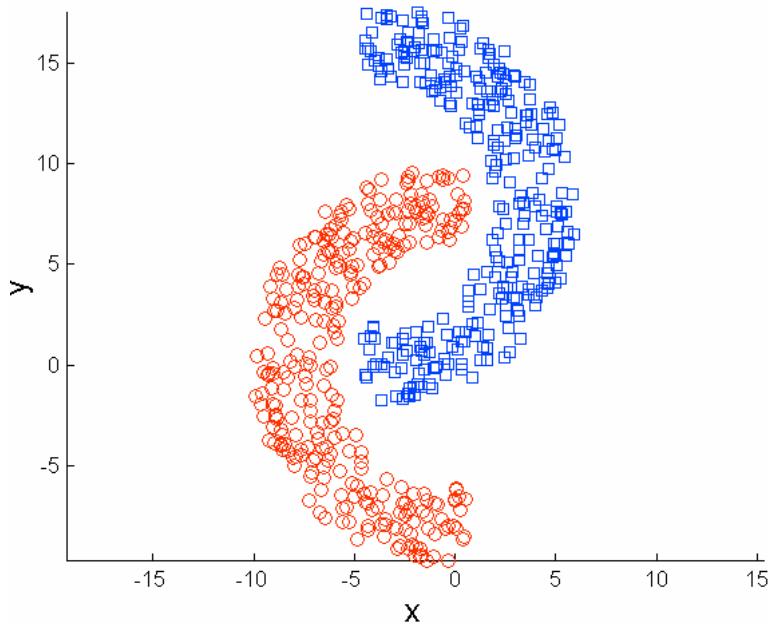
CLASS TASK



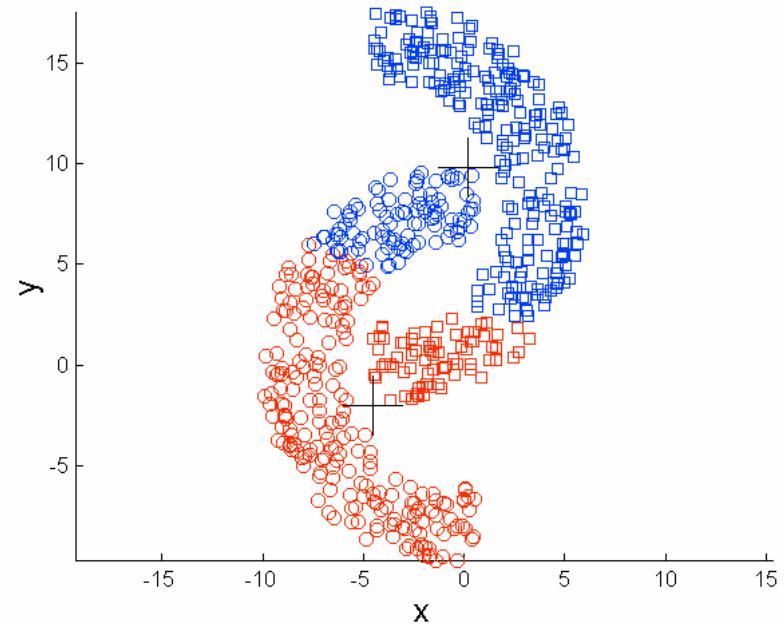
K-means (3 Clusters)

LIMITATIONS OF K-MEANS

Non-globular Shapes

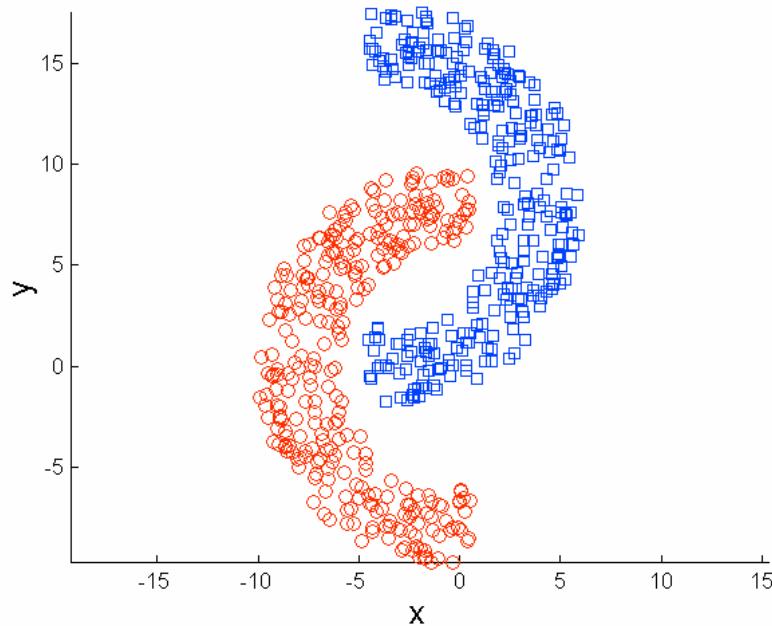


Original Points

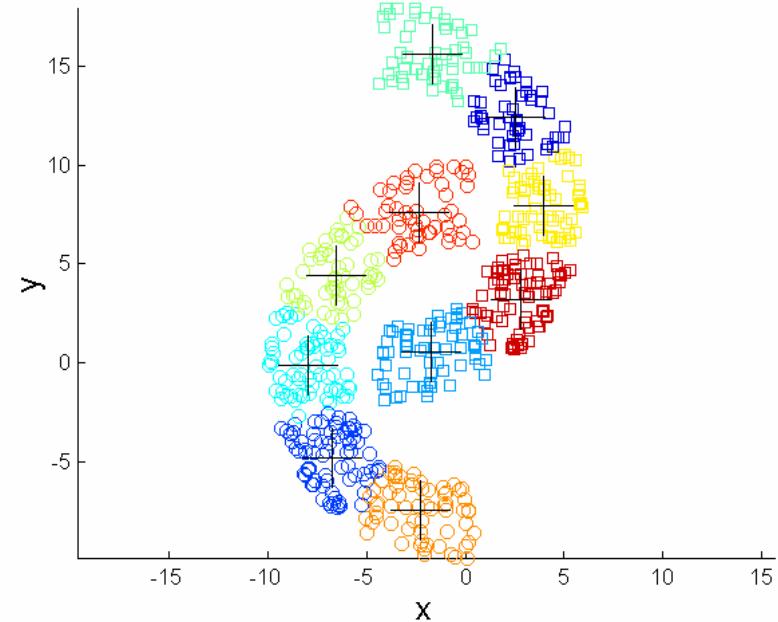


K-means (2 Clusters)

OVERCOMING K-MEANS LIMITATIONS



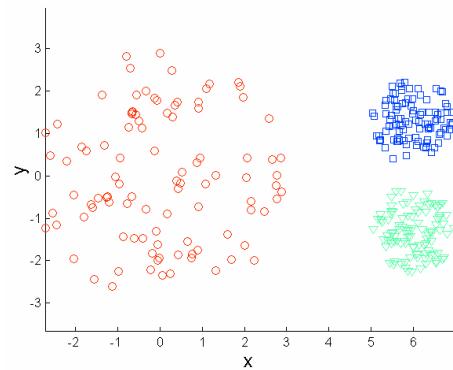
Original Points



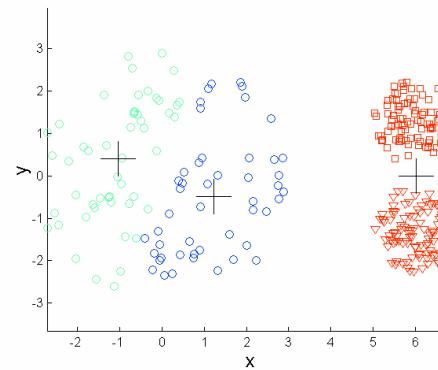
K-means Clusters

Can you think of other ways to overcome the limitations?

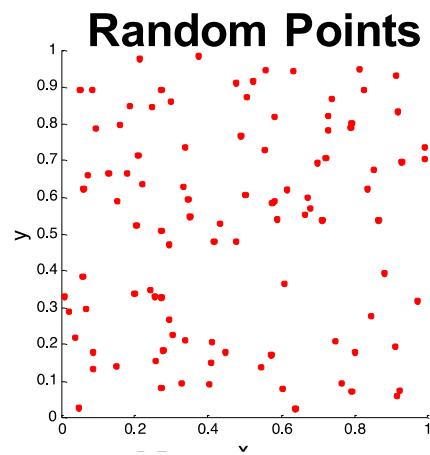
How do I know how good the clustering is?



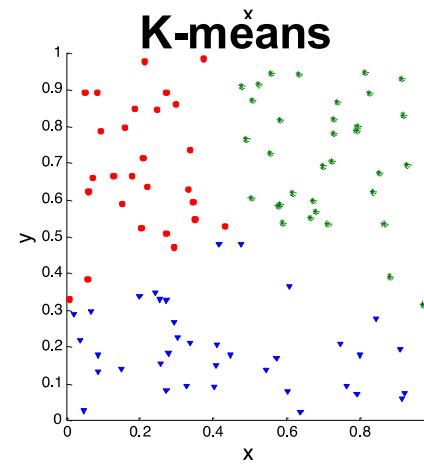
Original Points



K-means (3 Clusters)



Random Points

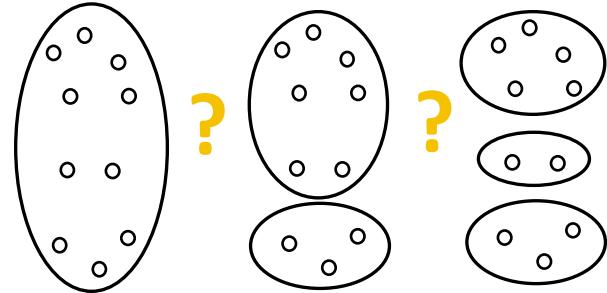


K-means

Measuring clustering validity

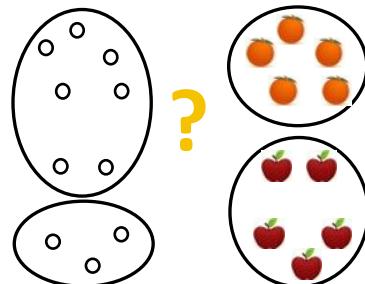
Internal Index:

- Validate *without* external info
- With different number of clusters |



External Index

Validate against ground truth



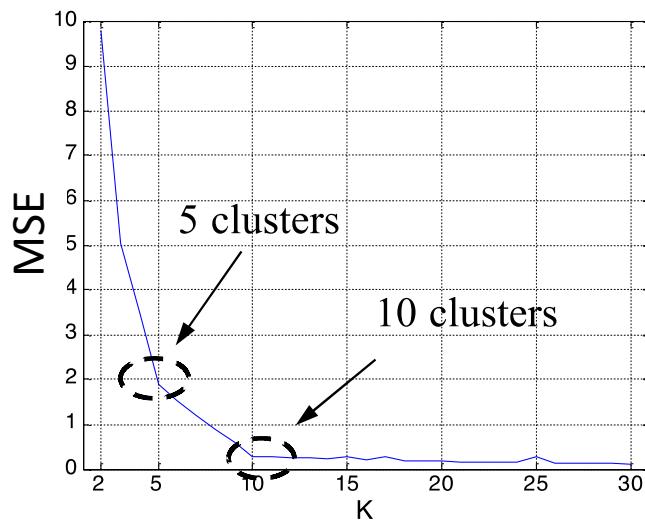
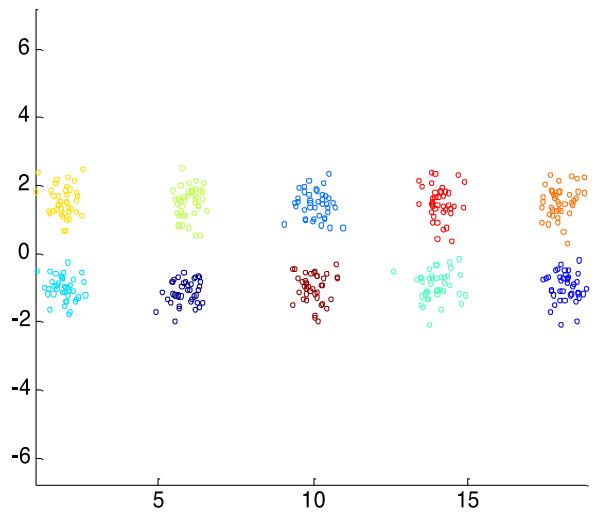
INTERNAL INDEXES

Ground truth is rarely available but unsupervised validation must be done.

Minimizes (or maximizes) internal index:

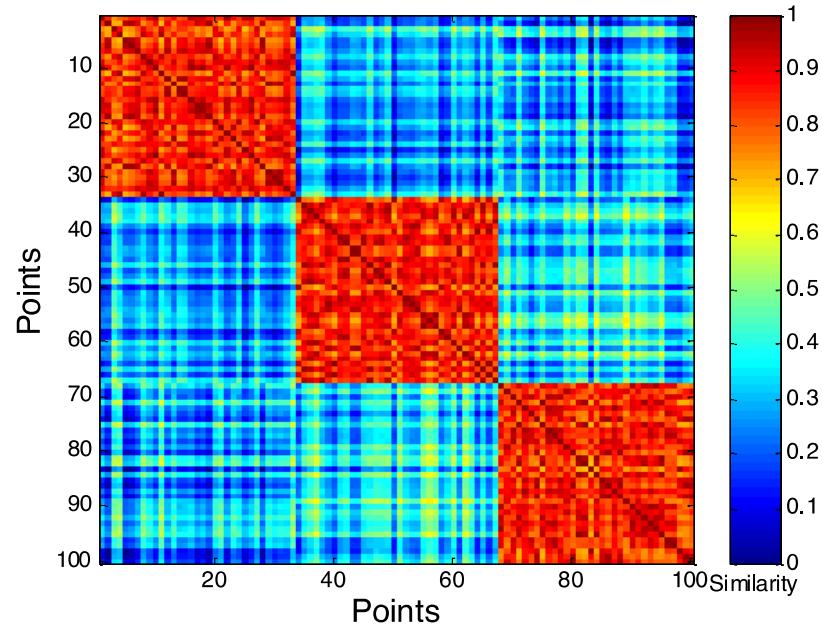
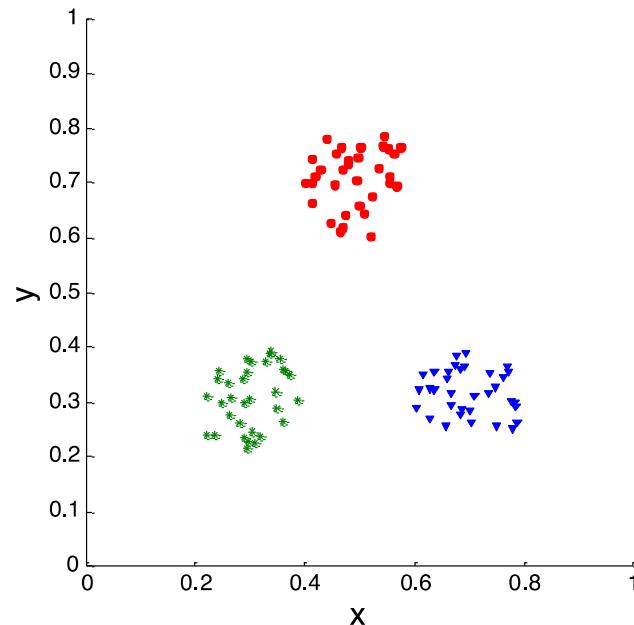
- Variances of within cluster and between clusters
- Rate-distortion method
- F-ratio
- Davies-Bouldin index (DBI)
- Bayesian Information Criterion (BIC)
- Silhouette Coefficient
- Minimum description principle (MDL)
- Stochastic complexity (SC)

MEAN SQUARE ERROR (MSE)



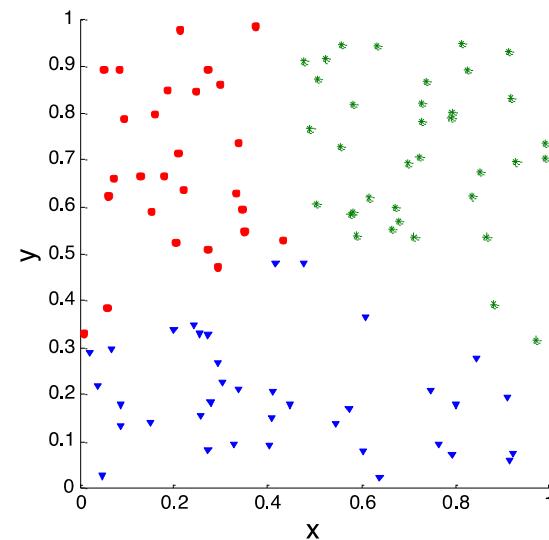
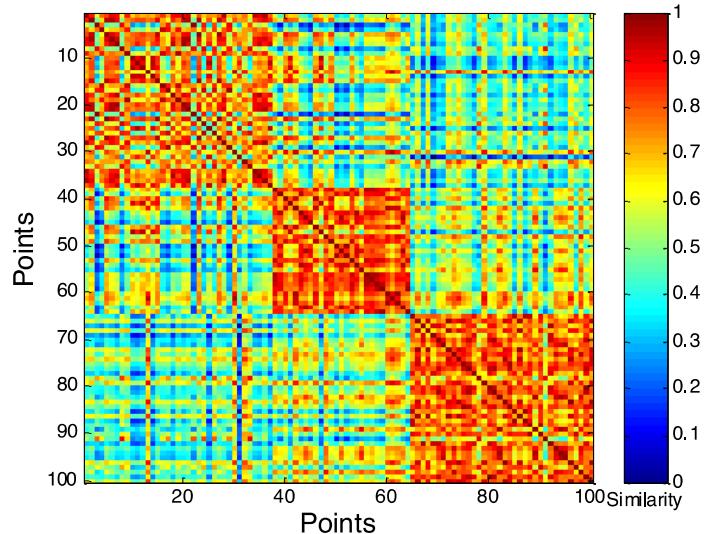
USING SIMILARITY MATRIX FOR CLUSTER VALIDATION

Order the similarity matrix with respect to cluster labels and inspect visually.



USING SIMILARITY MATRIX FOR CLUSTER VALIDATION

Clusters in random data are not so crisp



K-means

CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

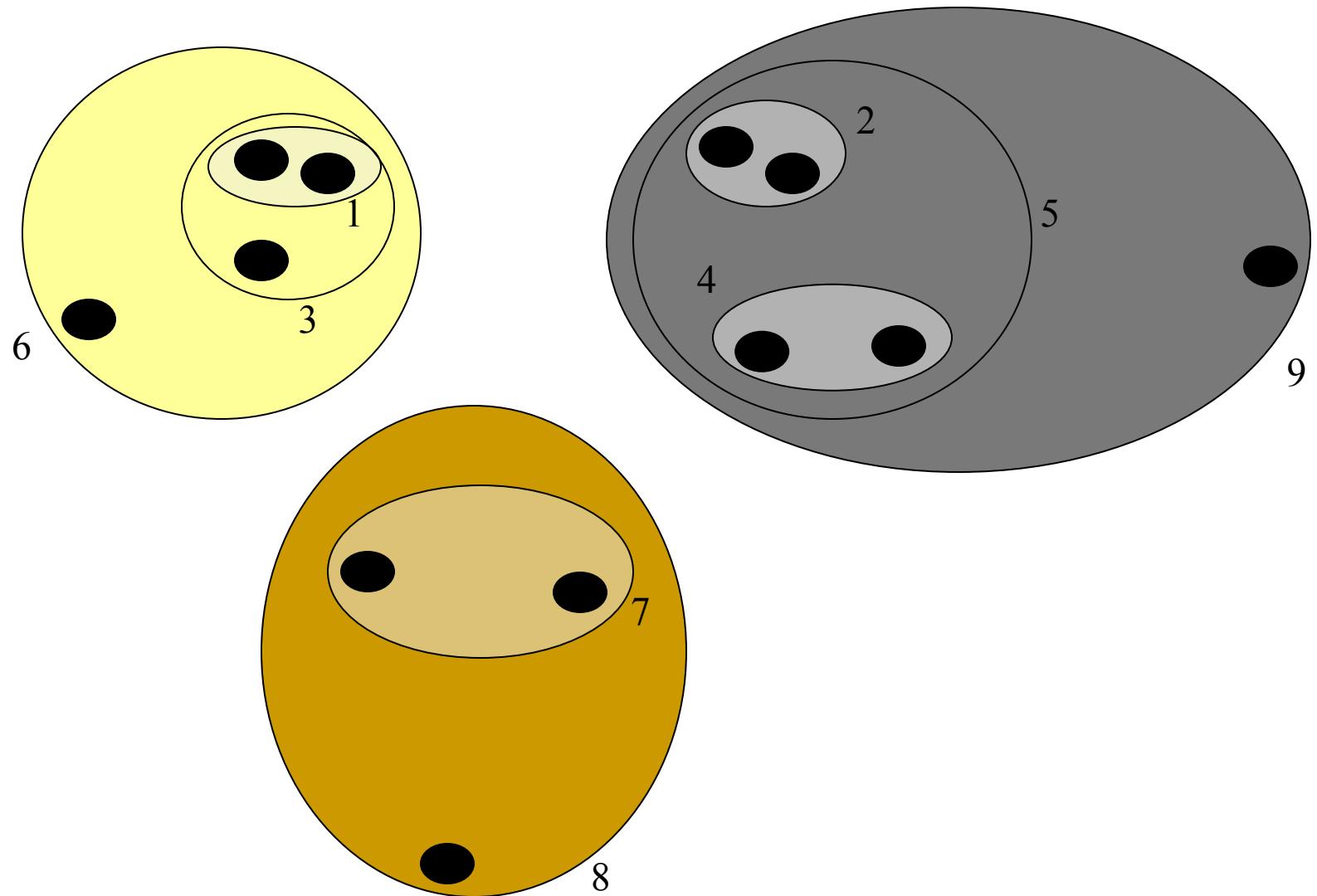
As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

HIERARCHICAL AGGLOMERATIVE CLUSTERING METHODS

Generic Agglomerative Procedure (Salton '89):

- result in nested clusters via iterations
1. Compute all pairwise document-document similarity coefficients
 2. Place each of n documents into a class of its own
 3. Merge the two most similar clusters into one;
 - replace the two clusters by the new cluster
 - recompute intercluster similarity scores w.r.t. the new cluster
 4. Repeat the above step until there are only k clusters left (note k could = 1).

Group Agglomerative Clustering



CLUSTERING STRATEGIES

K-means

- Iteratively re-assign points to the nearest cluster center

Agglomerative clustering

- Start with each point as its own cluster and iteratively merge the closest clusters

Mean-shift clustering

- Estimate modes of PDF (i.e., the value x at which its probability mass function takes its maximum value)

Spectral clustering

- Split the nodes in a graph based on assigned links with similarity weights

DBSCAN (Density-based spatial clustering of applications with noise)

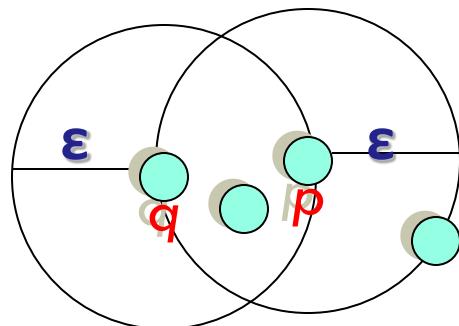
As we go down this chart, the clustering strategies have more tendency to transitively group points even if they are not nearby in feature space

ε -NEIGHBORHOOD

ε -Neighborhood – Objects within a radius of ε from an object.

$$N_{\varepsilon}(p) : \{q \mid d(p, q) \leq \varepsilon\}$$

“High density” – ε -Neighborhood of an object contains at least MinPts of objects.



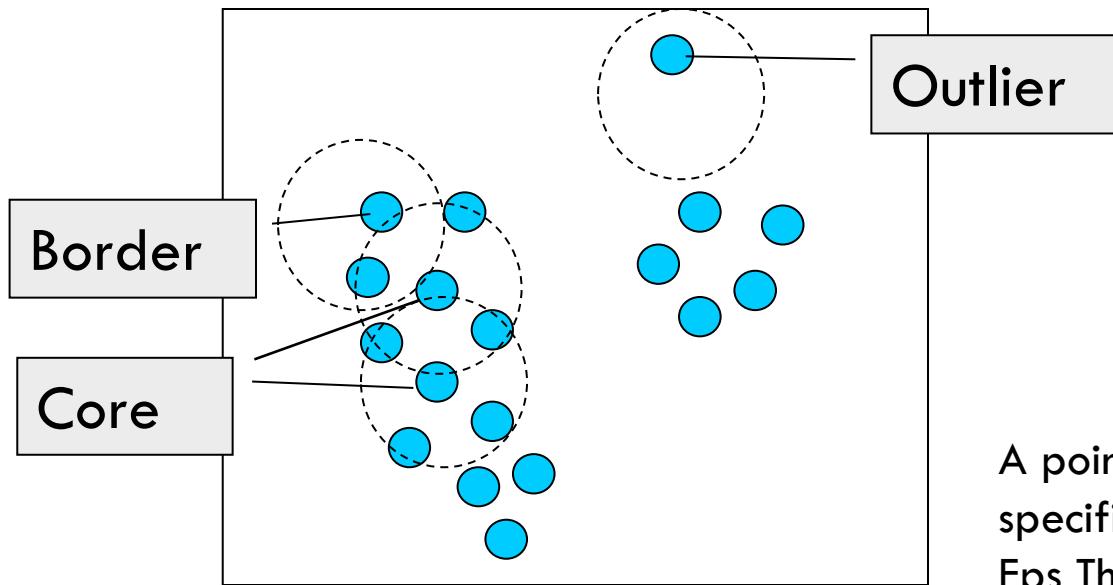
ε -Neighborhood of p

ε -Neighborhood of q

Density of p is “high” ($\text{MinPts} = 4$)

Density of q is “low” ($\text{MinPts} = 4$)

CORE, BORDER & OUTLIER



$\epsilon = 1 \text{ unit}$, $\text{MinPts} = 5$

Given ϵ and MinPts , categorize the objects into three exclusive groups.

A point is a **core point** if it has more than a specified number of points (MinPts) within ϵ . These are points that are at the interior of a cluster.

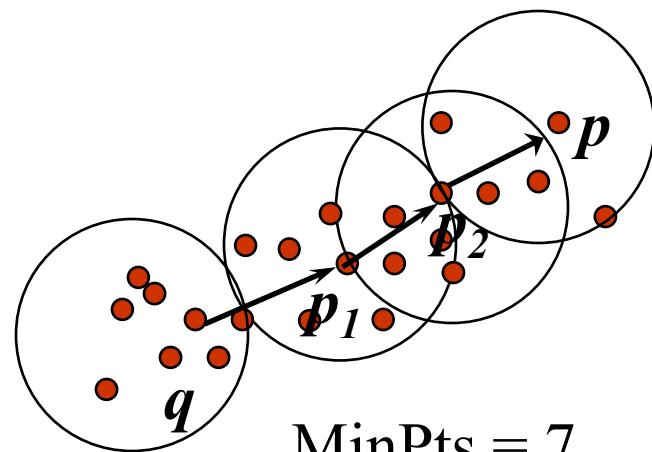
A **border point** has fewer than MinPts within ϵ , but is in the neighborhood of a core point..

A **noise point (outlier)** is any point that is not a core point nor a border point.

DENSITY-REACHABILITY

Density-Reachable (directly and indirectly):

- A point p is directly density-reachable from p_2 ;
- p_2 is directly density-reachable from p_1 ;
- p_1 is directly density-reachable from q ;
- $p \leftarrow p_2 \leftarrow p_1 \leftarrow q$ form a chain.



p is (indirectly) density-reachable from q

q is not density- reachable from p ?

DBSCAN ALGORITHM

Input: The data set D

Parameter: ε , MinPts

For each object p in D

 if p is a core object and not processed then

 C = retrieve all objects density-reachable from p

 mark all objects in C as processed

 report C as a cluster

 else mark p as outlier

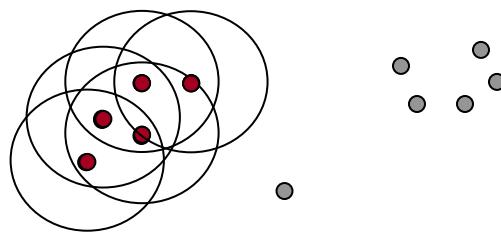
 end if

End For

DBSCAN ALGORITHM: EXAMPLE

Parameter

- $\varepsilon = 2 \text{ cm}$
- $\text{MinPts} = 3$

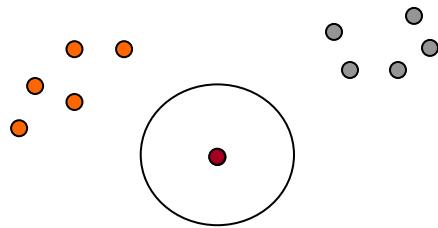


```
for each  $o \in D$  do
    if  $o$  is not yet classified then
        if  $o$  is a core-object then
            collect all objects density-reachable from  $o$ 
            and assign them to a new cluster.
        else
            assign  $o$  to NOISE
```

DBSCAN ALGORITHM: EXAMPLE

Parameter

- $\varepsilon = 2 \text{ cm}$
- $\text{MinPts} = 3$

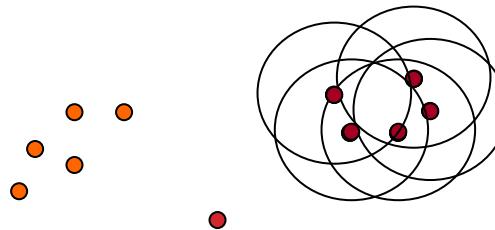


```
for each  $o \in D$  do
    if  $o$  is not yet classified then
        if  $o$  is a core-object then
            collect all objects density-reachable from  $o$ 
            and assign them to a new cluster.
        else
            assign  $o$  to NOISE
```

DBSCAN ALGORITHM: EXAMPLE

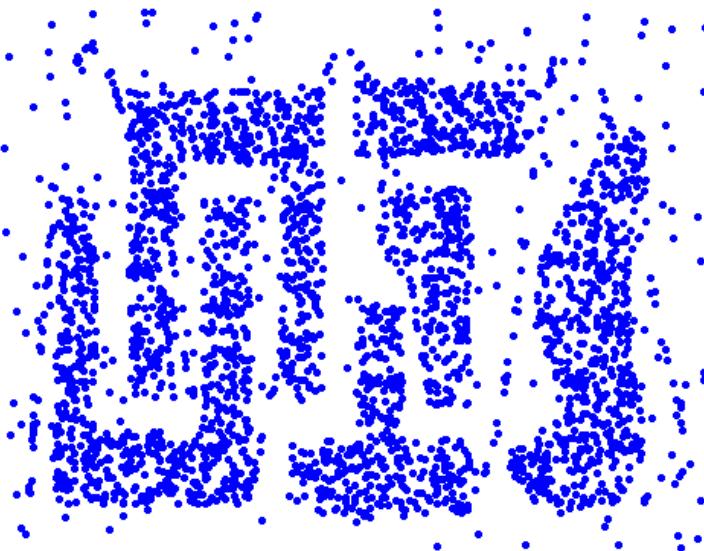
Parameter

- $\varepsilon = 2 \text{ cm}$
- $\text{MinPts} = 3$

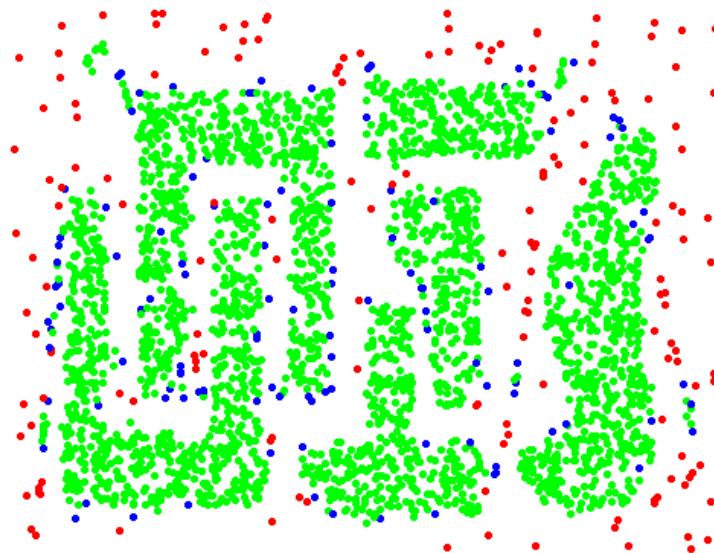


```
for each  $o \in D$  do
    if  $o$  is not yet classified then
        if  $o$  is a core-object then
            collect all objects density-reachable from  $o$ 
            and assign them to a new cluster.
        else
            assign  $o$  to NOISE
```

EXAMPLE



Original Points



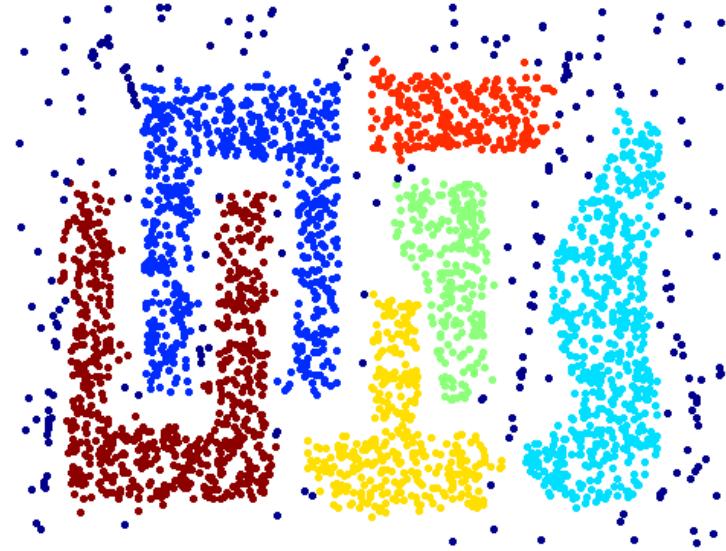
Point types: **core**,
border and **outliers**

$\varepsilon = 10$, MinPts = 4

WHEN DBSCAN WORKS WELL



Original Points

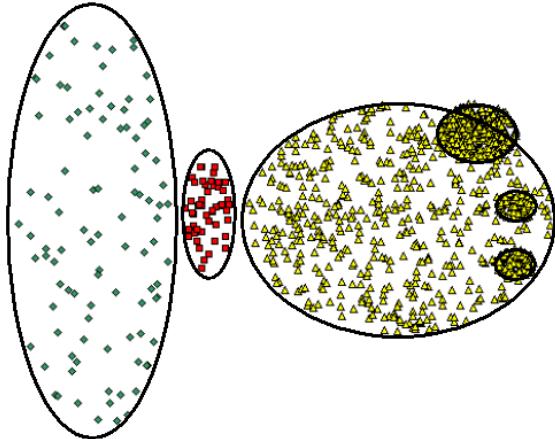


Clusters

- Resistant to Noise
- Can handle clusters of different shapes and sizes

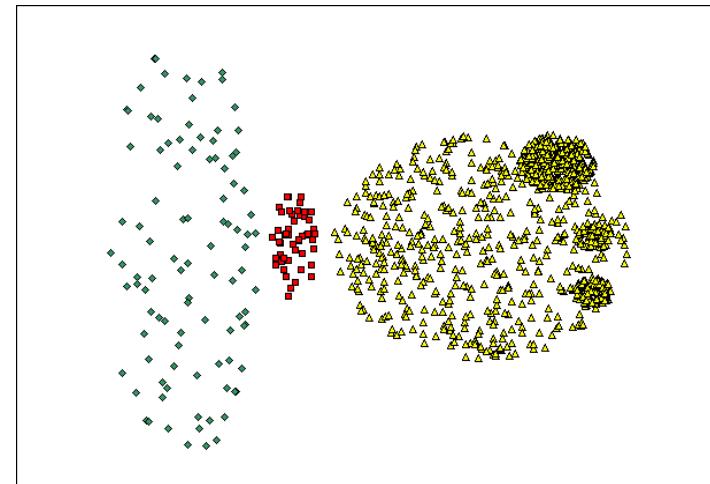
CAN YOU CREATE AN EXAMPLE FOR WHICH
DBSCAN WILL NOT WORK WELL

WHEN DBSCAN DOES NOT WORK WELL

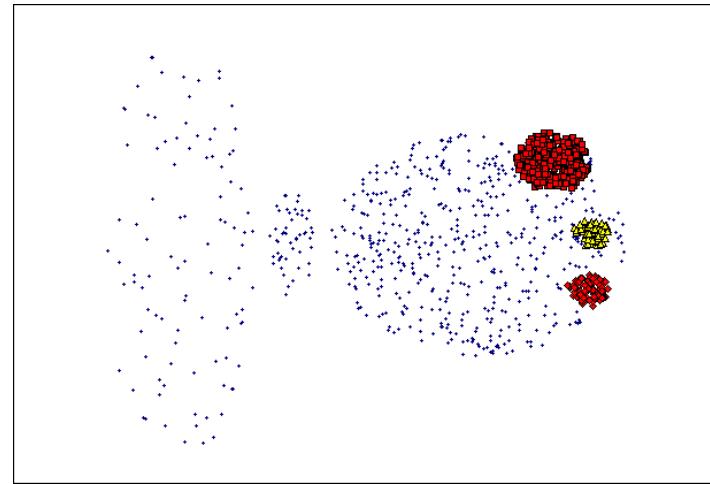


Original Points

- Cannot handle Varying densities
- Sensitive to parameters

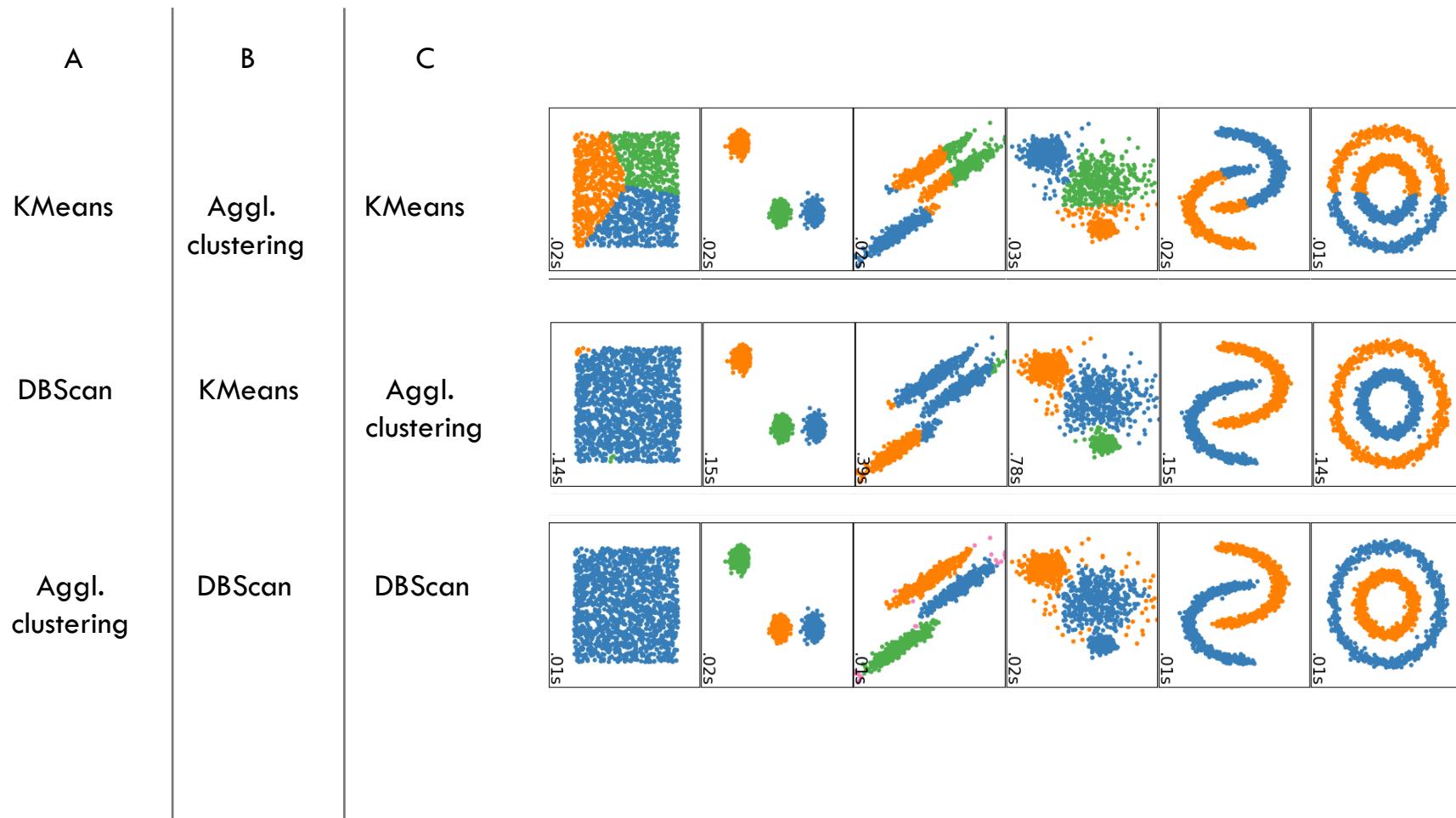


(MinPts=4, Eps=9.92).

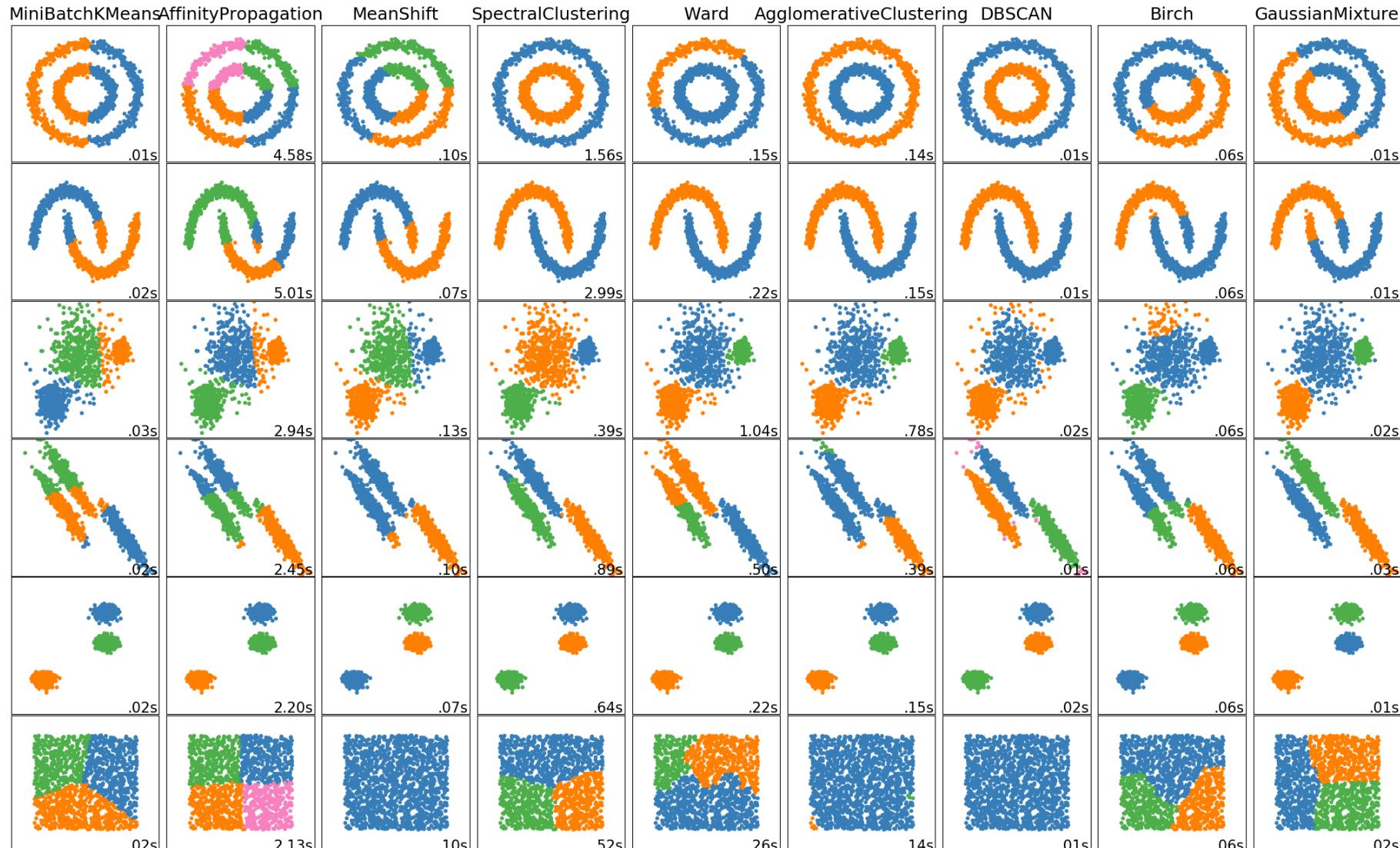


(MinPts=4, Eps=9.75)

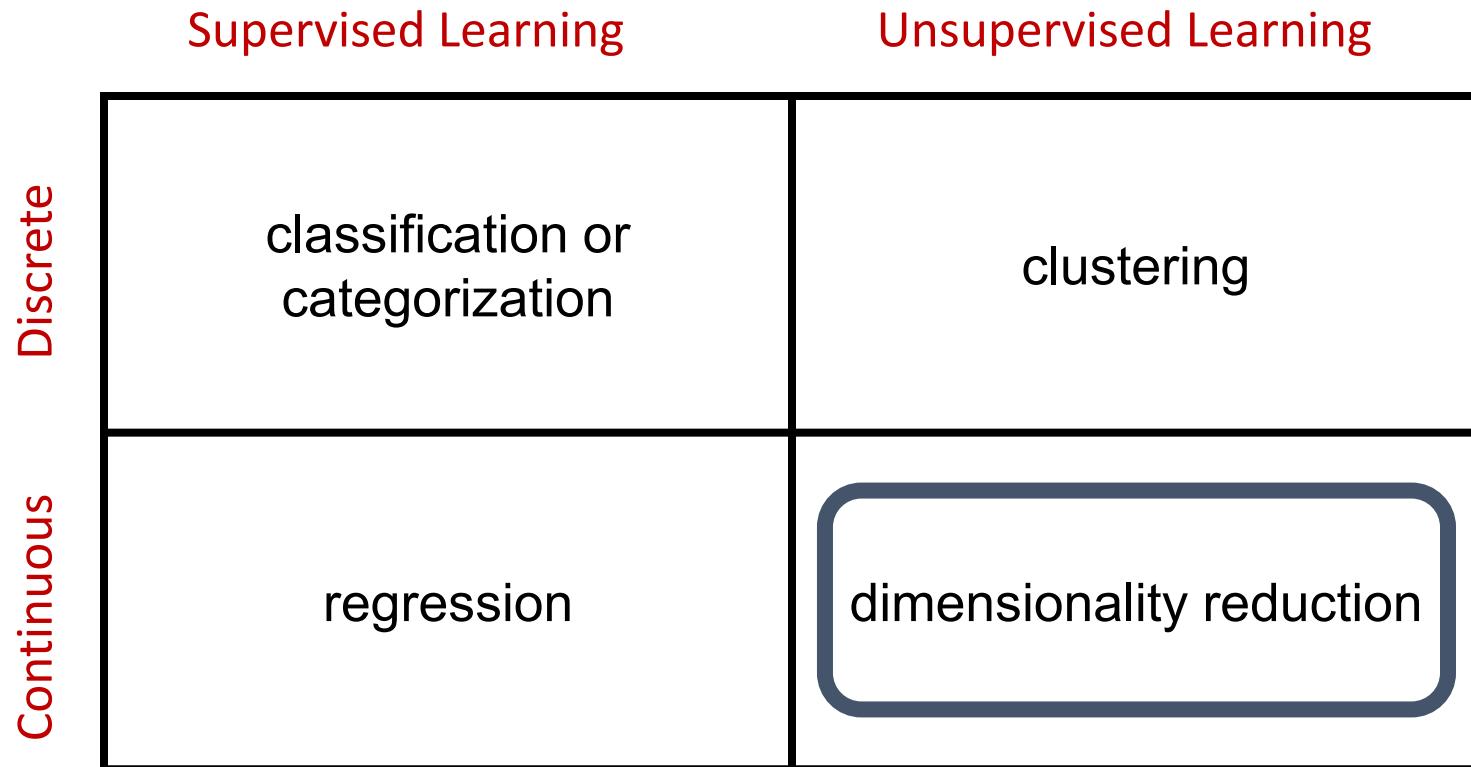
CLICKER - [HTTPS://CLICKER.CSAIL.MIT.EDU/6.S079/](https://clicker.csail.mit.edu/6.S079/)



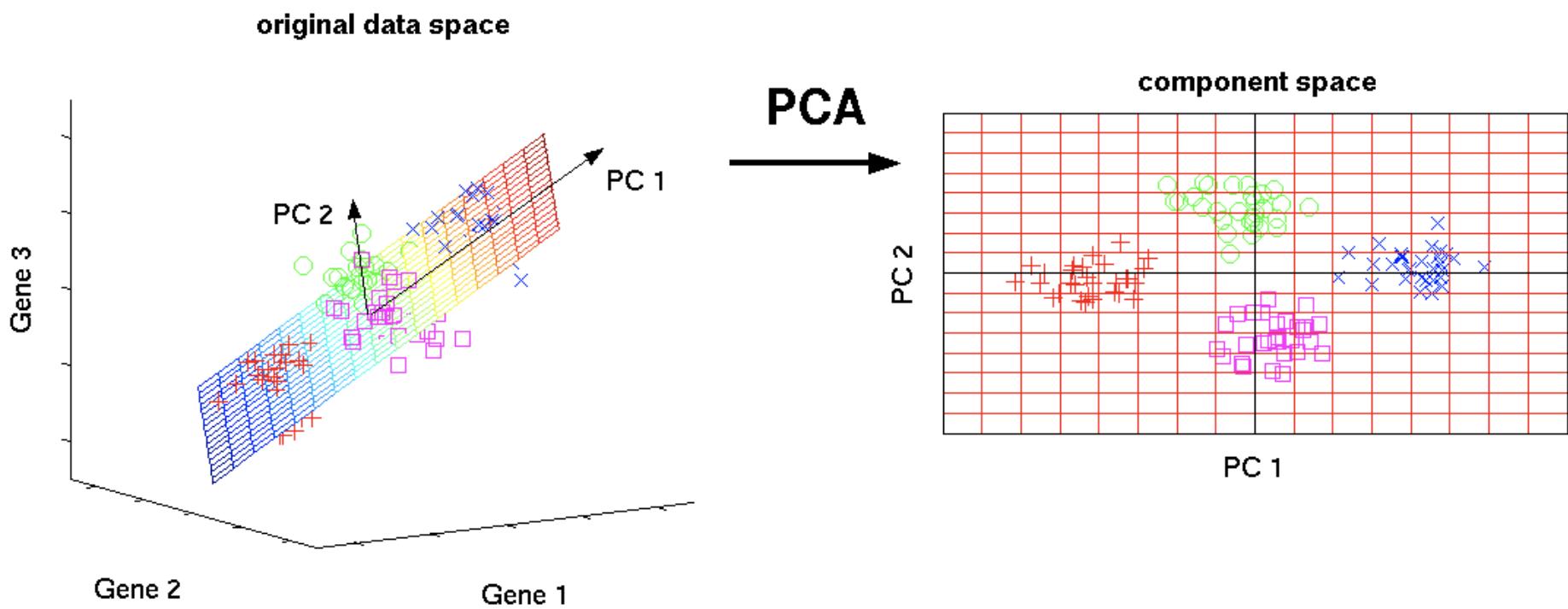
CLUSTERING



Machine Learning Problems

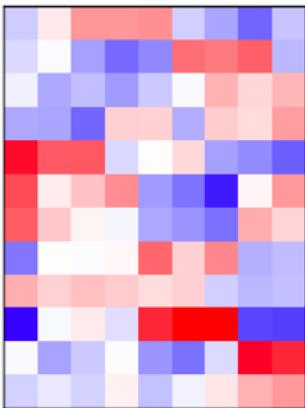


PCA



PCA Intuition

Original Data

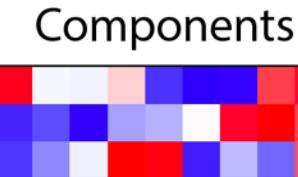


\approx

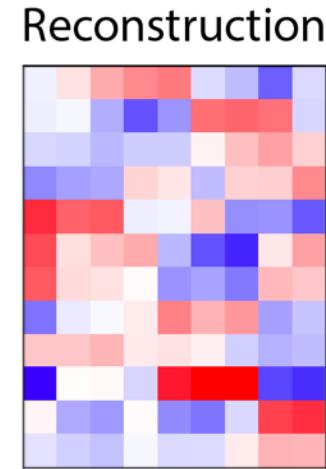
Loadings



\times



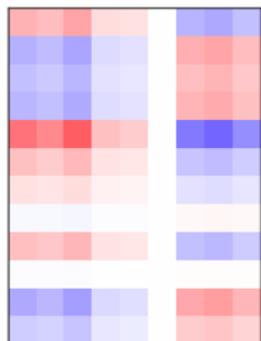
$=$



Sum
of
Rank-1
Matrices



$+$



$+$

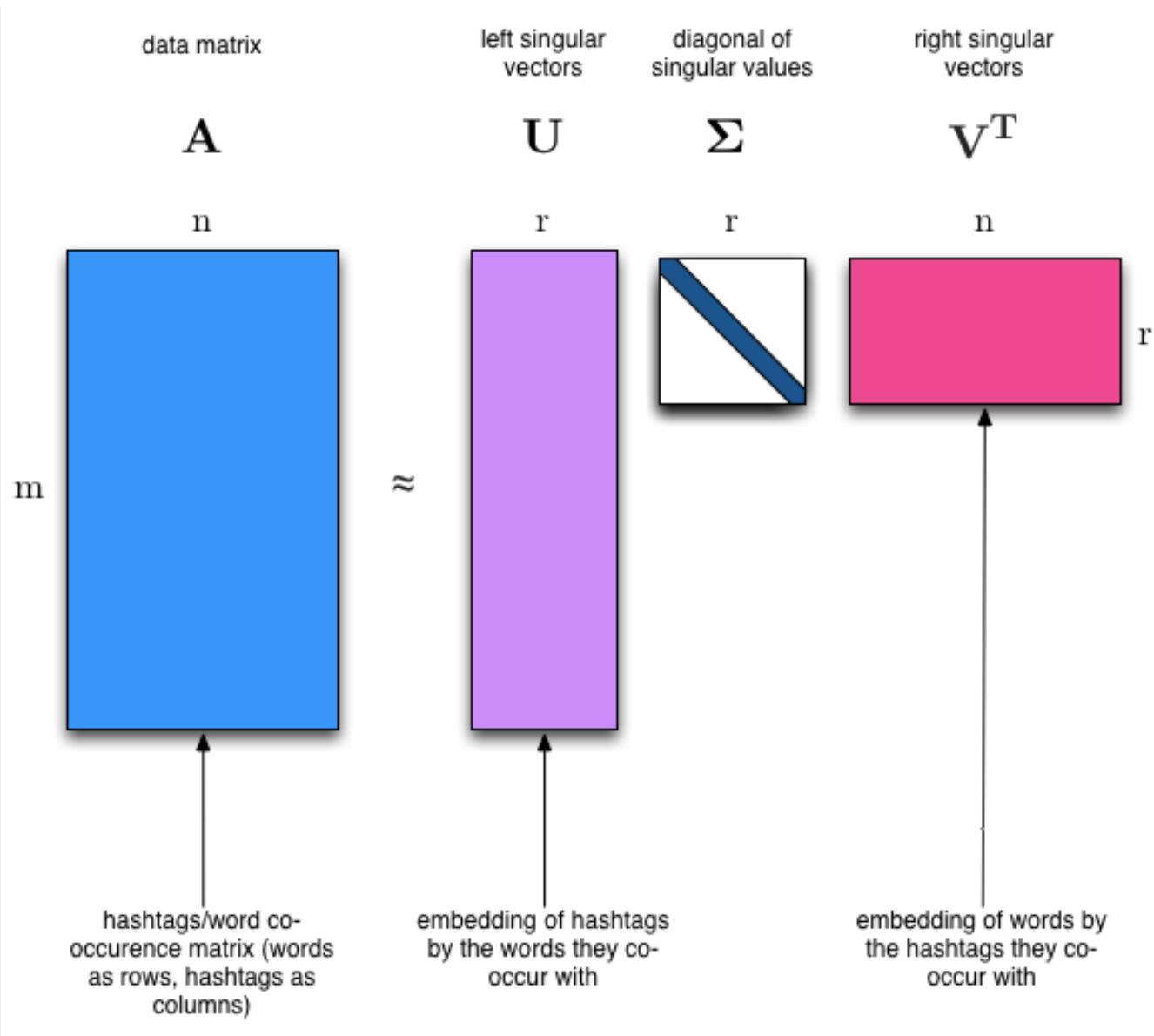


Positive
Numbers

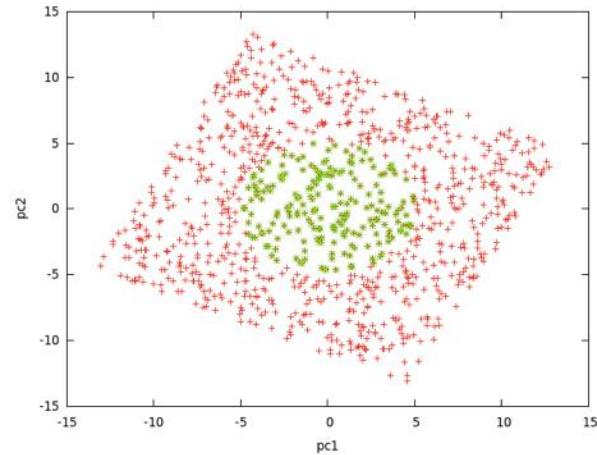
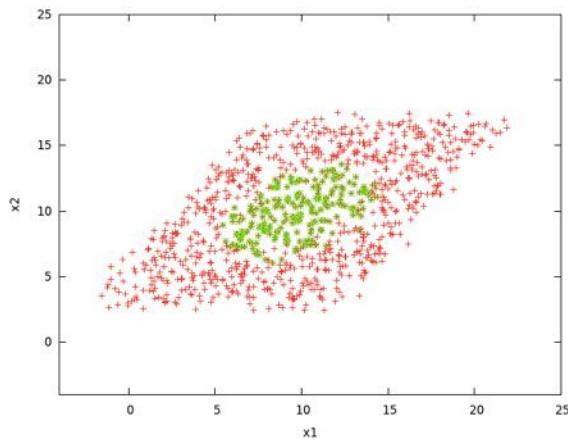
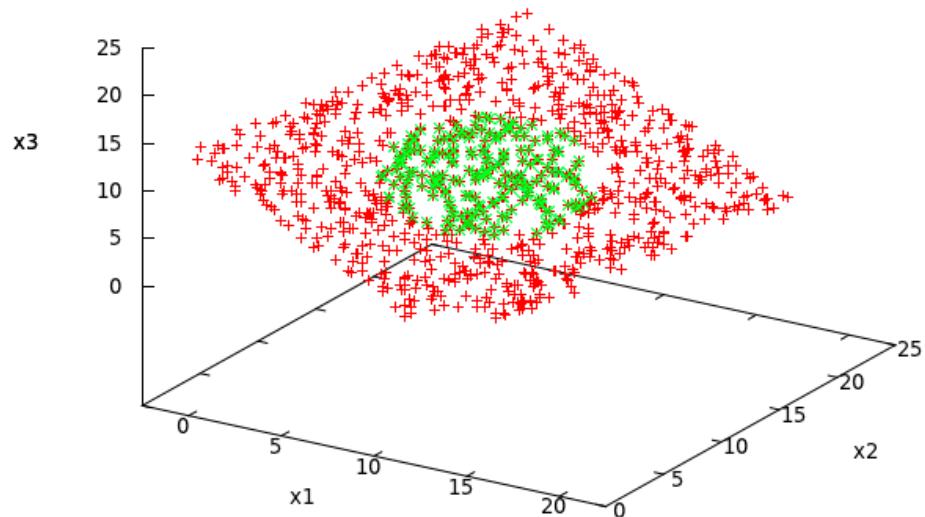
zero →

Negative
Numbers





PRINCIPAL COMPONENT ANALYSIS (PCA)



MACHINE LEARNING PROBLEMS

(Boosted-) Decision Trees

K-Means
Agglomerative clustering
DBScan

Supervised Learning

Unsupervised Learning

Discrete

classification or categorization

clustering

Continuous

regression

dimensionality reduction

(Boosted-) Decision Trees

PCA

WHAT IS A CLASSIFIER

Apply a prediction function to a feature representation of an image/data-set to get the desired output:

$$f(\text{apple}) = \text{"apple"}$$

$$f(\text{tomato}) = \text{"tomato"}$$

$$f(\text{cow}) = \text{"cow"}$$

THE MACHINE LEARNING FRAMEWORK

$$y = f(x)$$

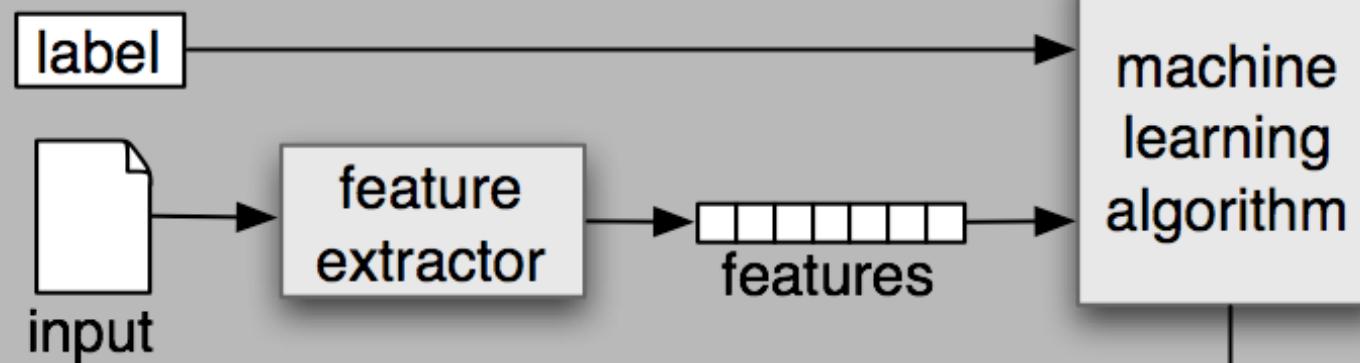
↑
output prediction
function features

Training: given a *training set* of labeled examples $\{(x_1, y_1), \dots, (x_N, y_N)\}$, estimate the prediction function f by minimizing the prediction error on the training set

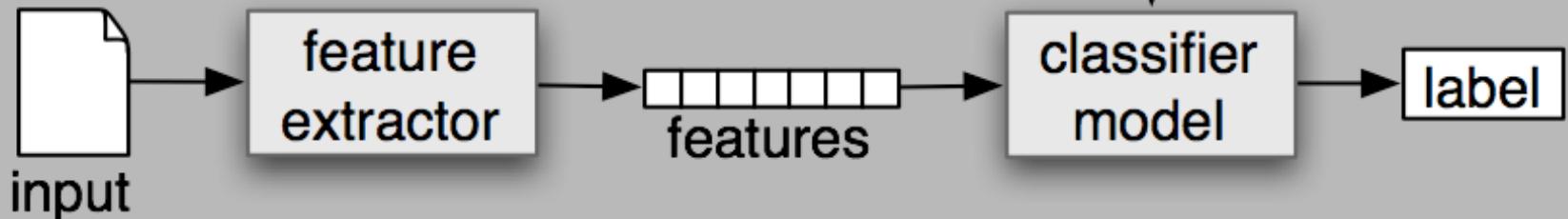
Testing: apply f to a never before seen *test example* x and output the predicted value $y = f(x)$

ML PIPELINE (SUPERVISED)

(a) Training

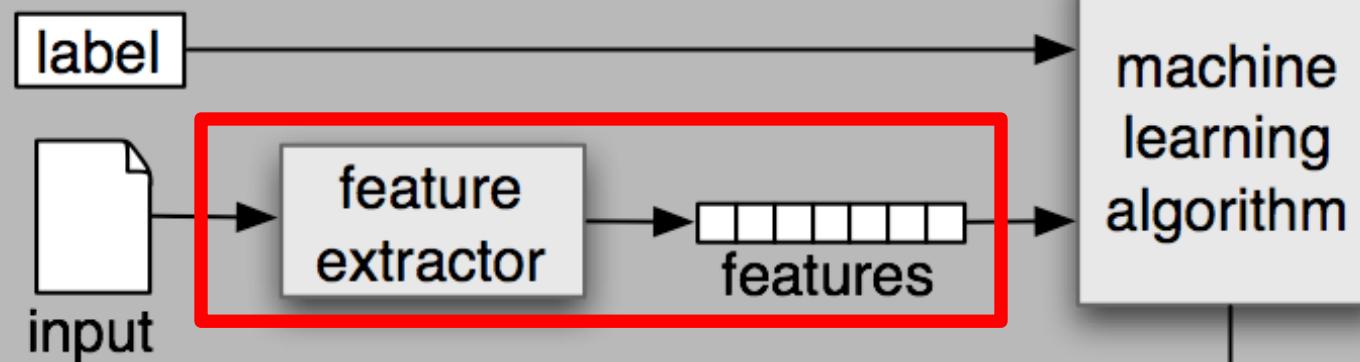


(b) Prediction

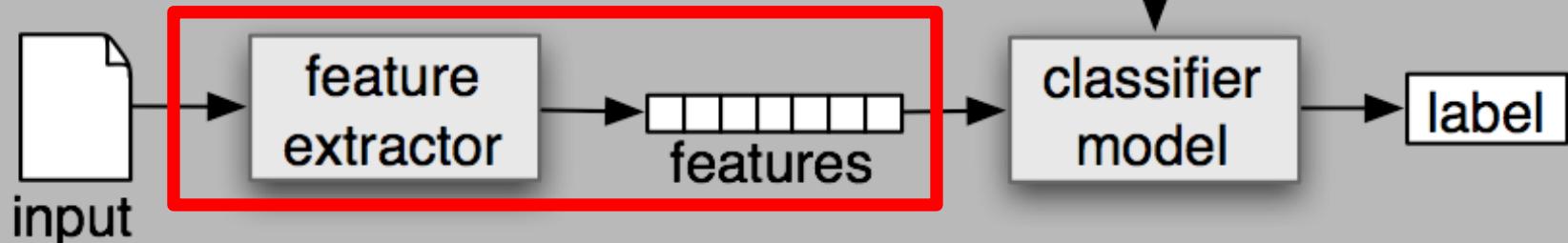


ML PIPELINE (SUPERVISED)

(a) Training



(b) Prediction



FEATURES

Fact Table	
-	<u>Shop_ID</u>
-	<u>Customer_ID</u>
-	<u>Date_ID</u>
-	<u>Product_ID</u>
-	Amount
-	Volume
-	Profit
-	...

Fact Table	
-	<u>Shop_ID</u>
-	<u>Customer_ID</u>
-	<u>Date_ID</u>
-	<u>Product_ID</u>
-	Amount
-	Volume
-	Profit
-	Delivery Time
-	...

Product	
-	<u>Product_ID</u>
-	Type_ID
-	Brand_ID
-	Length
-	Height
-	Depth
-	Weight
-	...

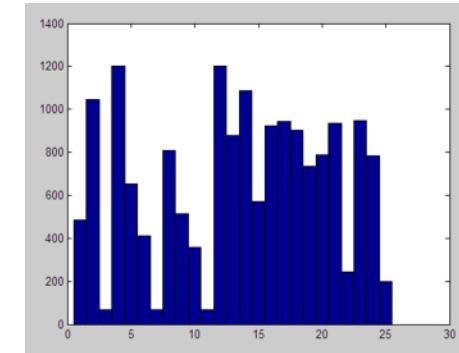
Product_Type	
-	<u>Type_ID</u>
-	Name
-	Description
-	...

Brand	
-	<u>Brand_ID</u>
-	Name
-	...

Customer State	Product Type	Product Weight	Volume (L*H*D)	Month	Delivery Time

IMAGE FEATURES

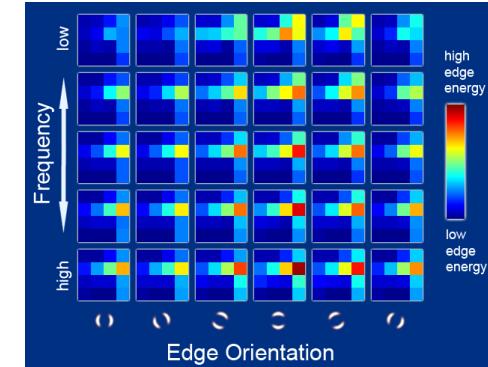
Raw pixels



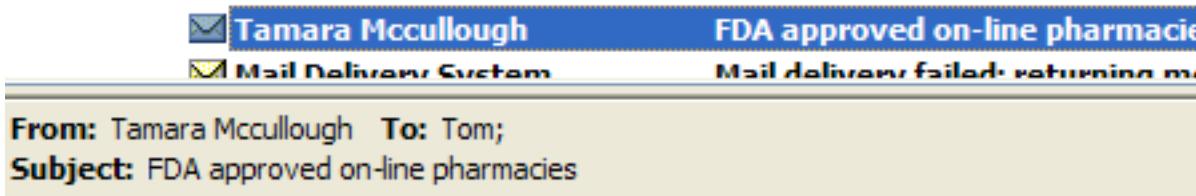
Histograms

GIST descriptors

...



TEXT FEATURES

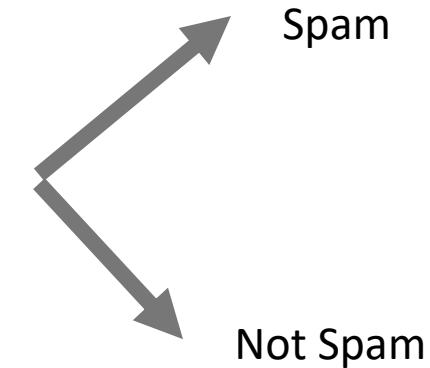


FDA approved on-line pharmacies.
Chose your product and site below:

[**Canadian pharmacy**](#) - Cialis Soft Tabs - \$5.78, [**Viagra Professional**](#) - \$1.38, Human Growth Hormone - \$43.37, Meridia - \$3.32, Tramadol - \$1.38

[**HerbalKing**](#) - Herbal pills for **Hair enlargement**. Techniques, products, dangerous pumps, exercises and surgeries.

[**Anatrim**](#) - Are you ready for Summer? Use **Anatrim**, the most powerful



Bag of Words

$\left(\begin{array}{l} Viagra: 1 \\ Soft: 1 \\ Herbel: 2 \\ Pills: 2 \\ Are: 1 \\ \dots \end{array} \right)$

N-Grams

$\left(\begin{array}{l} herbel pills: 1 \\ pills for: 1 \\ for Hair: 2 \\ Hair enlargement: 1 \\ surgeries: 2 \\ \dots \end{array} \right)$

ONE-HOT ENCODING

Bag of Words

$$\begin{pmatrix} Viagra \\ Soft \\ Herbel \\ Pills \\ Are \\ \dots \end{pmatrix}$$

ID	Viagra	Soft	Herbel	Pills	Are
Mail1	0	1	1	0	1	...
Mail2	1	0	0	1	1	...
...

PREDICTOR FOR GRAD-SCHOOL APPLICATIONS

Name	ZipCode	Age	Sex	Area	Avg Grade	Statement	Early admit	Accepted
Mike	02474	23	M	DB	B-	Since I was born, I knew I wanted to code. My first program I wrote in binary code literally in the sandbox, though I am not sure it was correct...	No	NO
Sam	02456	21	M	Sensor	A	Celine Dion's song "A New Day Has Come" taught me that CS is the best subject in the world. I never felt...	Yes	Yes
Amadou	15106	22	M	DB	A+	I want to get out of Pittsburgh.	No	Yes
Anna	02319	22	F	ML	A-	I already wrote 10 papers and I think I am ready to graduate now.	Yes	Yes
...

HOW WOULD YOU ENCODE THE TABLE?

PREDICTOR FOR GRAD SCHOOL APPLICATION

Name	ZipCode	Age	Sex	Area	Avg Grade	Statement	Early admit	Accepted
Mike	02474	23	M	DB	B-	Since I was born, I knew I wanted to code. My first program I wrote in binary code literally in the sandbox, though I am not sure it was correct...	No	NO
Sam	02456	21	null	Sensor	A	Celine Dion's song "A New Day Has Come" taught me that CS is the best subject in the world. I never felt...	Yes	Yes
Amadou	15106	22	M	DB	A+	I want to get out of Pittsburgh.	No	Yes
Anna	null	22	F	ML	A-	I already wrote 10 papers and I think I am ready to graduate now.	Yes	Yes
...

Encode as numbers
(0-1)

Encode as numbers
(0-1)

Bag of words
1-Hot Encoding

Remove
(information leakage)

Remove
identifiers

Encode as (1) Lat/Lon
and scale to 0-1, or
remove

Scale to 0-1

1-Hot
Encode or
remove

PREDICTOR FOR GRAD-SCHOOL APPLICATIONS

Name	ZipCode	Age	Sex	Area	Avg Grade	Statement	Early admit	Accepted
Mike	02474	23	M	DB	B-	Since I was born, I knew I wanted to code. My first program I wrote in binary code literally in the sandbox, though I am not sure it was correct...	No	NO
Sam	02456	21	null	Sens or	A	Celine Dion's song "A New Day Has Come" taught me that CS is the best subject in the world. I never felt...	Yes	Yes
Amadou	15106	22	M	DB	A+	I want to get out of Pittsburgh.	No	Yes
Anna	null	22	F	ML	A-	I already wrote 10 papers and I think I am ready to graduate now.	Yes	Yes
...

MANY CLASSIFIERS TO CHOOSE FROM

K-nearest neighbor

Support Vector Machines

Decision Trees

Random Forrest

(Gradient) Boosted Decision Trees

Logistic Regression

Naïve Bayes

Bayesian network

RBM s

....

Which is the best one?

MANY CLASSIFIERS TO CHOOSE FROM

K-nearest neighbor

Support Vector Machines

Which is the best one?

Decision Trees

Random Forrest

(Gradient) Boosted Decision Trees

Logistic Regression

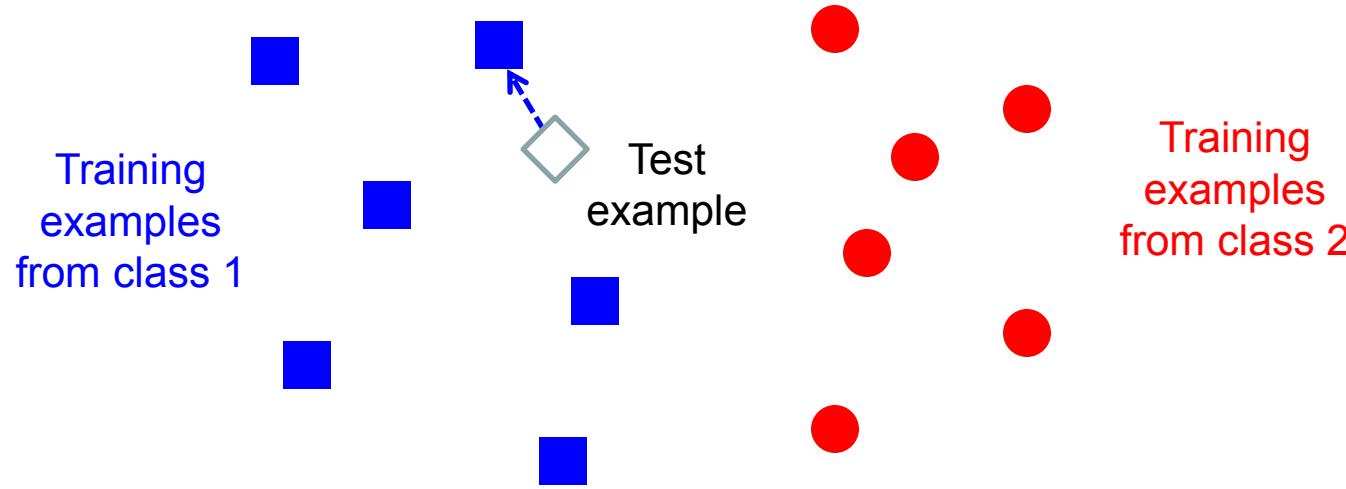
Naïve Bayes

Bayesian network

RBM s

....

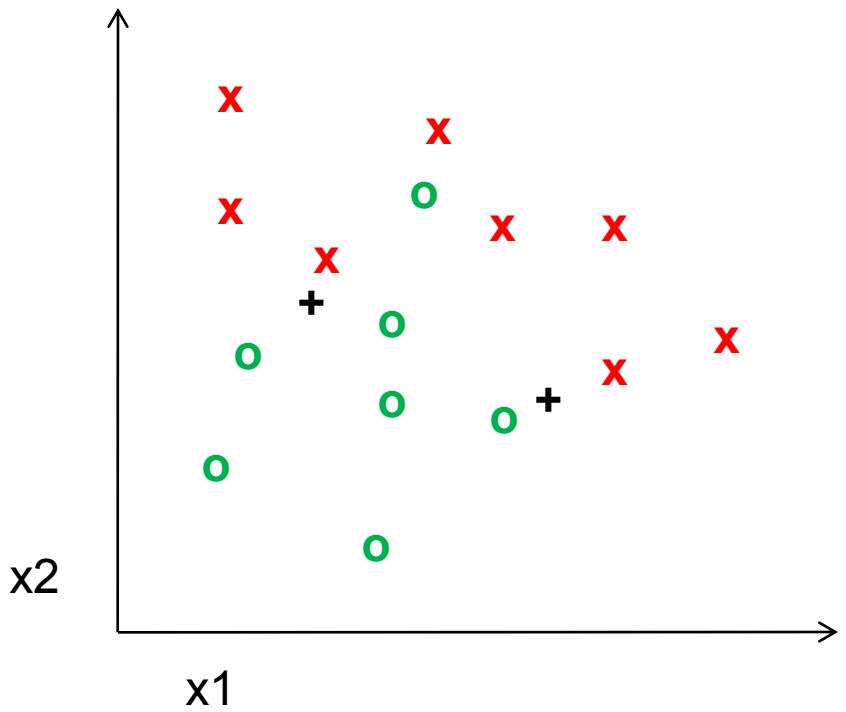
CLASSIFIERS: NEAREST NEIGHBOR



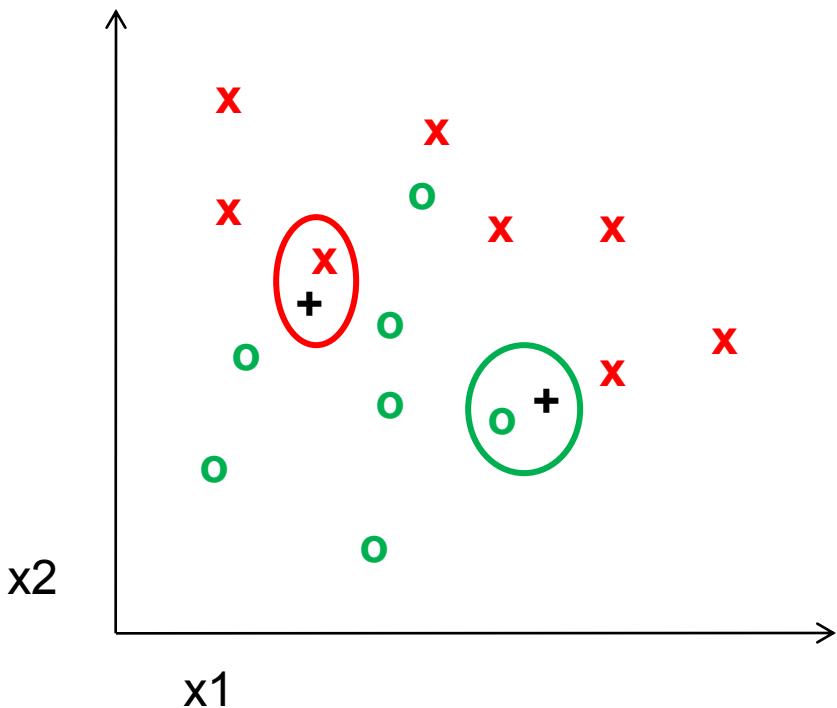
$f(x)$ = label of the training example nearest to x

- All we need is a distance function for our inputs
- No training required!

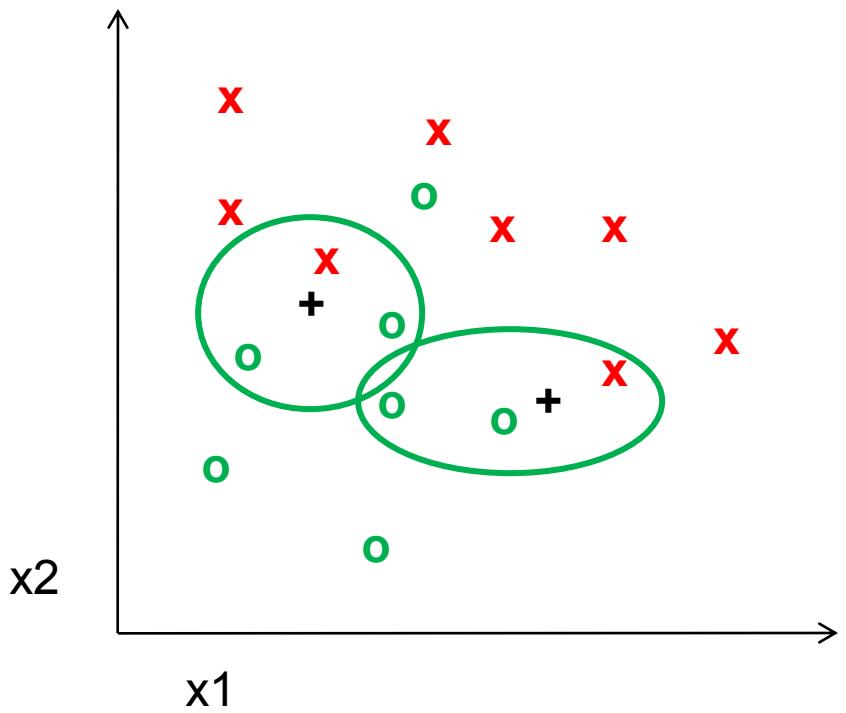
K-NEAREST NEIGHBOR



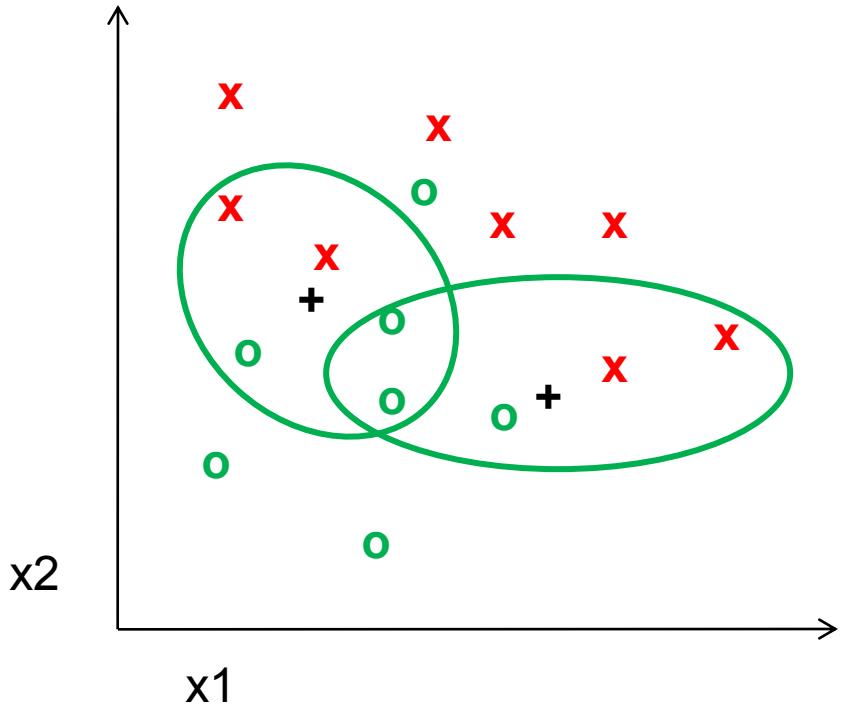
1-NEAREST NEIGHBOR



3-NEAREST NEIGHBOR

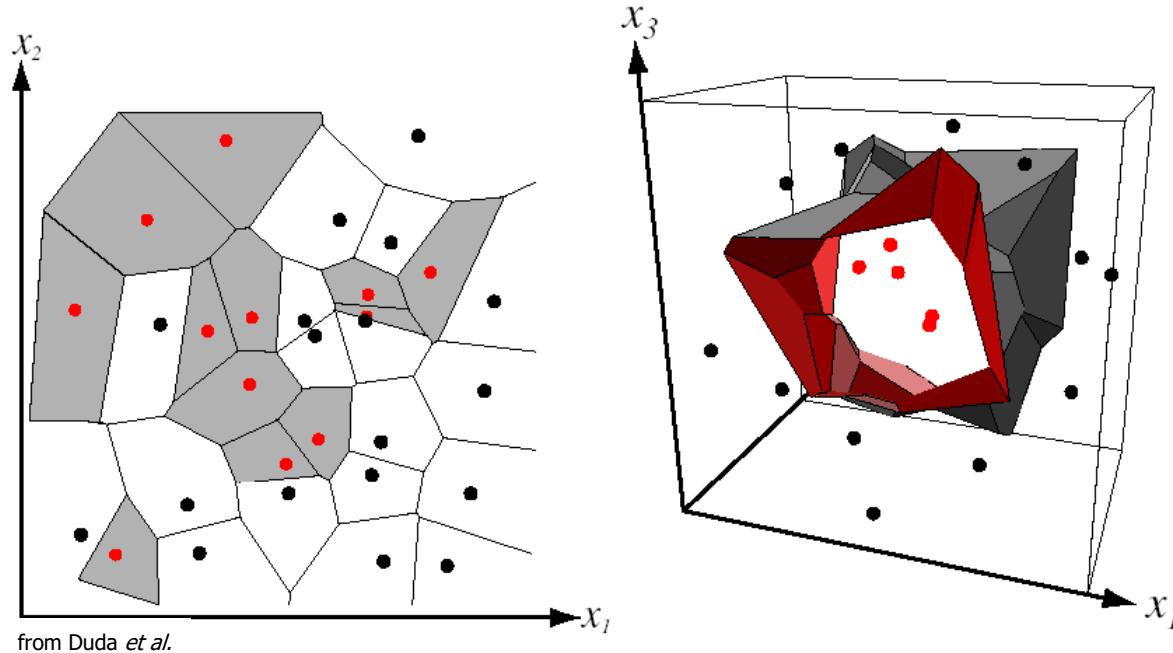


5-NEAREST NEIGHBOR



DECISION BOUNDARIES KNN

Assign label of nearest training data point to each test data point



Voronoi partitioning of feature space
for two-category 2D and 3D data

MANY CLASSIFIERS TO CHOOSE FROM

K-nearest neighbor

Support Vector Machines

Which is the best one?

Decision Trees

Random Forrest

(Gradient) Boosted Decision Trees

Logistic Regression

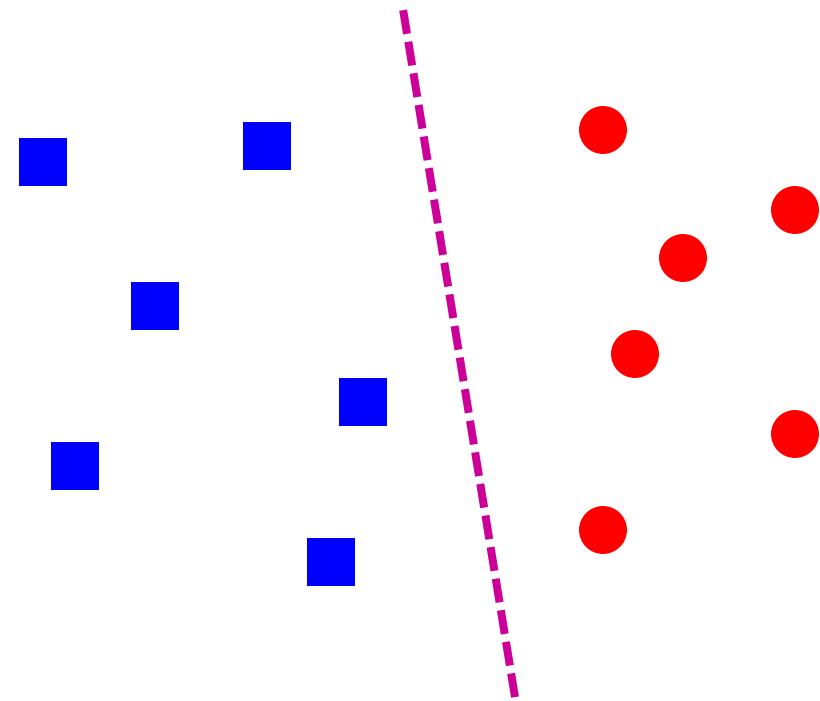
Naïve Bayes

Bayesian network

RBM s

....

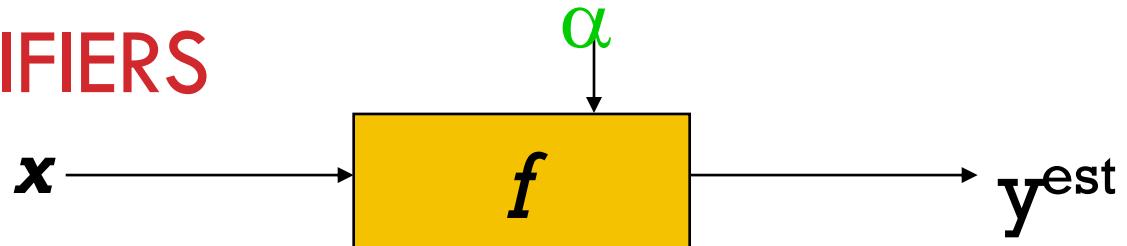
CLASSIFIERS: LINEAR



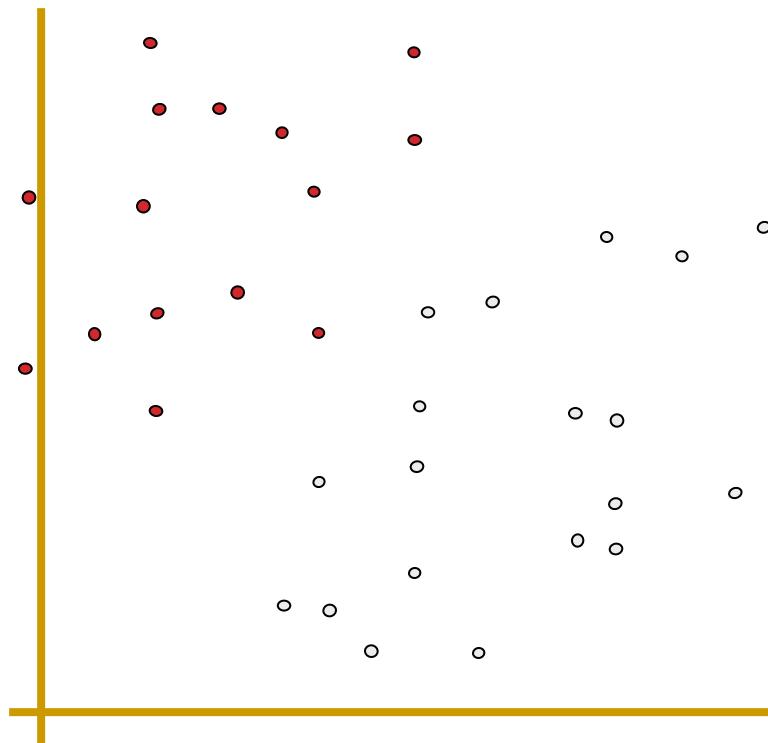
Find a *linear function* to separate the classes:

$$f(x) = \text{sgn}(w \cdot x + b)$$

LINEAR CLASSIFIERS



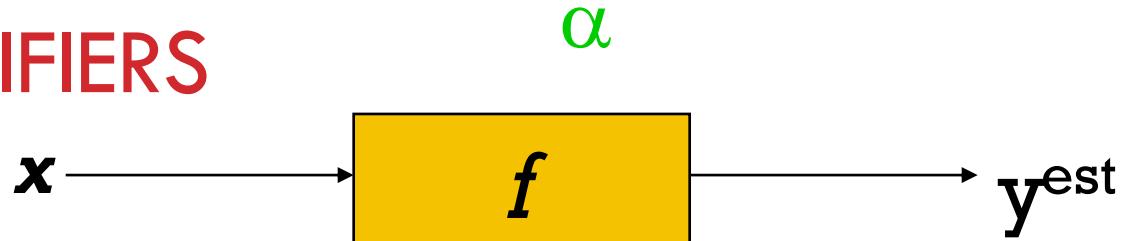
- denotes +1
- denotes -1



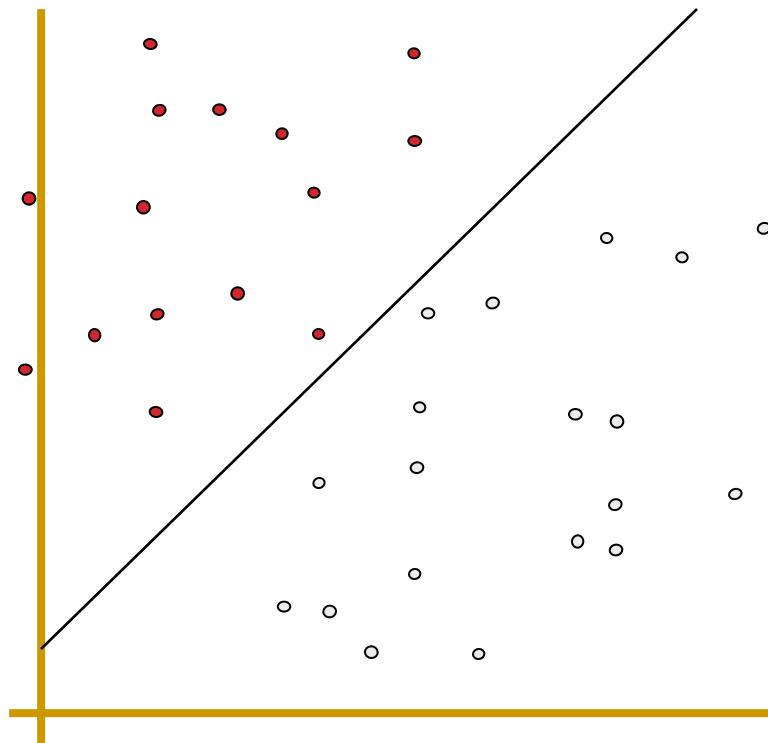
$$f(\mathbf{x}, \mathbf{w}, b) = \text{sign}(\mathbf{w} \cdot \mathbf{x} - b)$$

How would you
classify this data?

LINEAR CLASSIFIERS



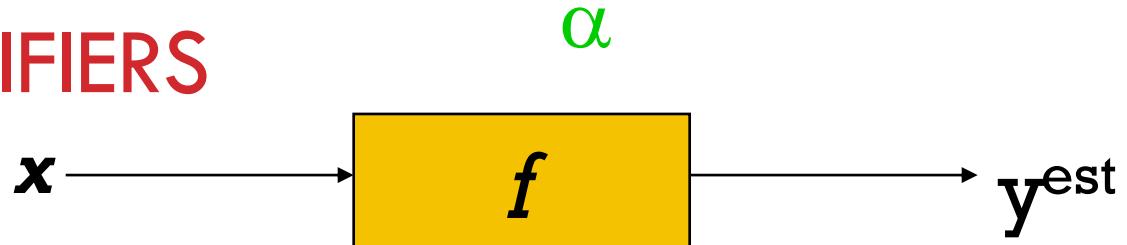
- denotes +1
- denotes -1



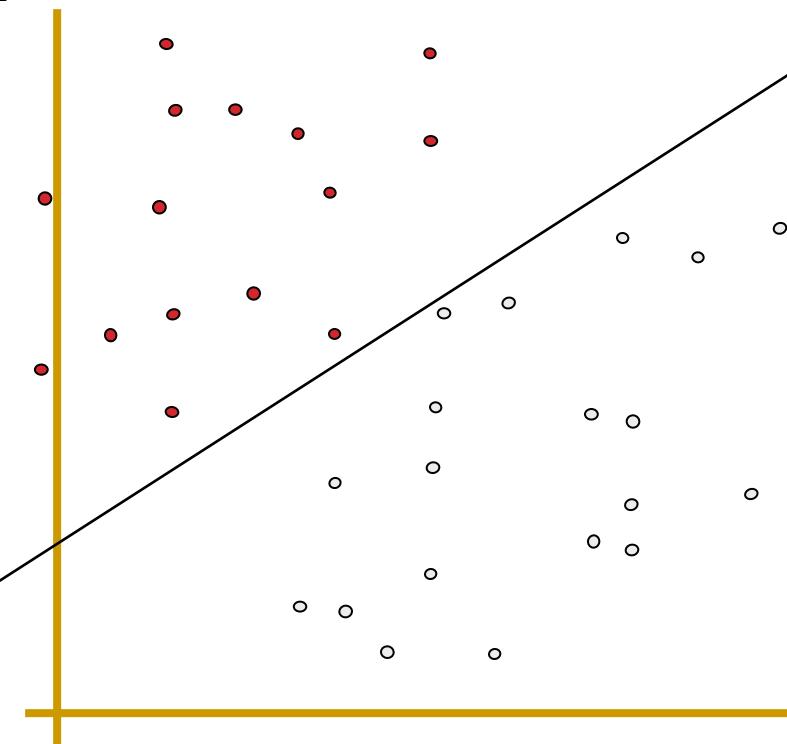
$$f(\mathbf{x}, \mathbf{w}, b) = \text{sign}(\mathbf{w} \cdot \mathbf{x} - b)$$

How would you
classify this data?

LINEAR CLASSIFIERS



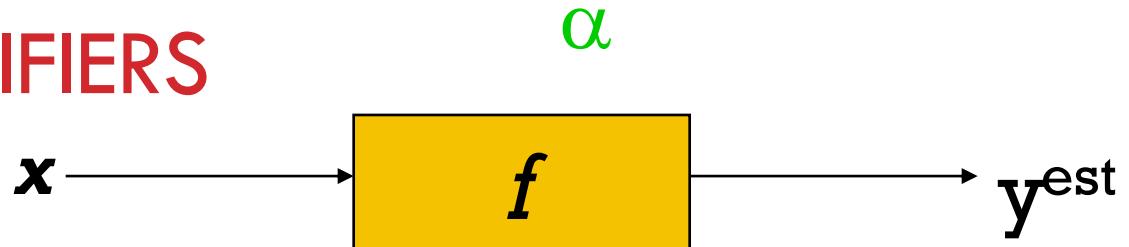
- denotes +1
- denotes -1



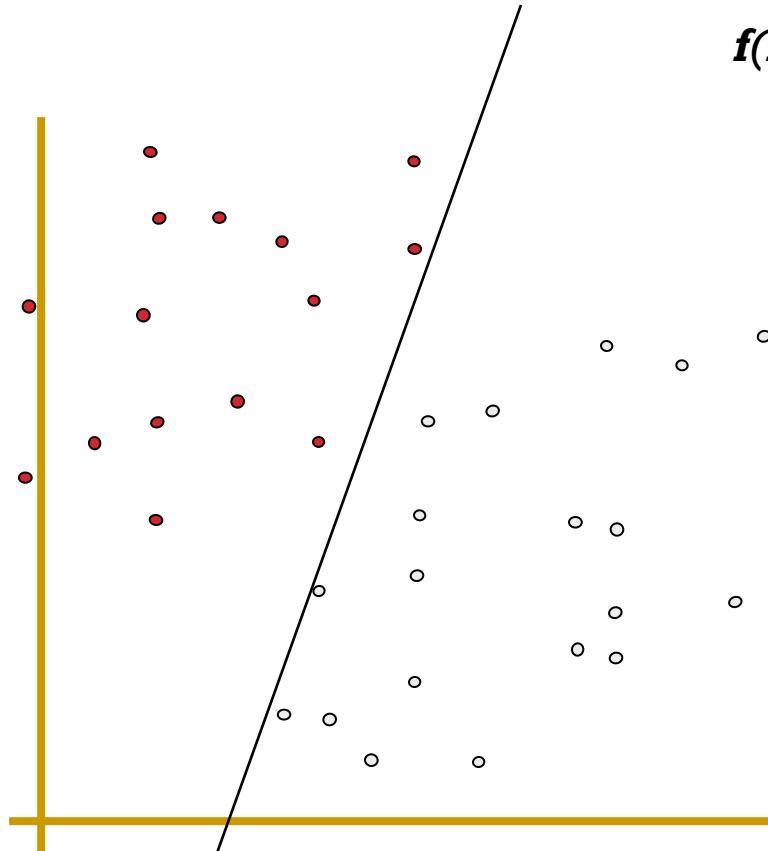
$$f(x, w, b) = \text{sign}(w \cdot x - b)$$

How would you
classify this data?

LINEAR CLASSIFIERS



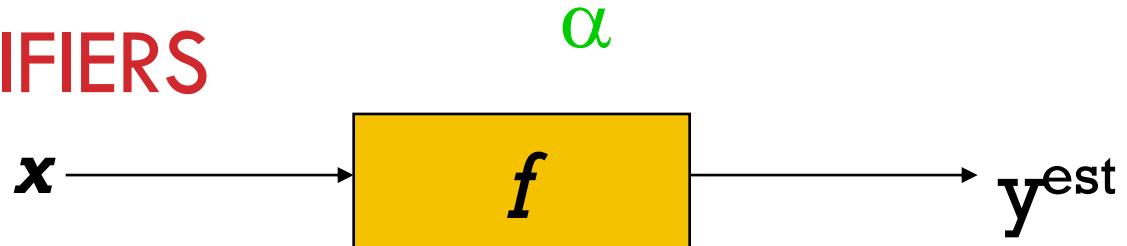
- denotes +1
- denotes -1



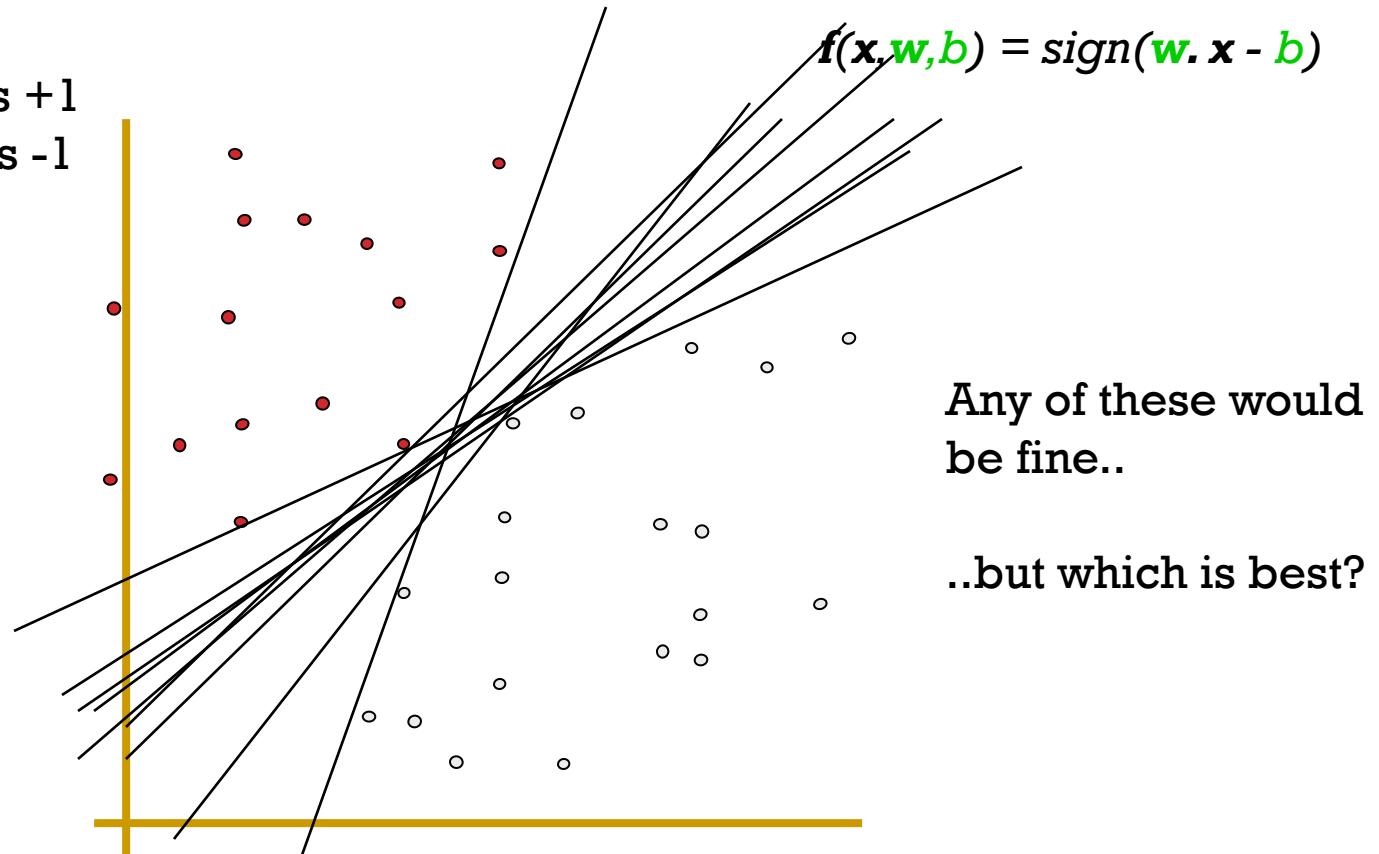
$$f(x, w, b) = \text{sign}(w \cdot x - b)$$

How would you
classify this data?

LINEAR CLASSIFIERS



- denotes +1
- denotes -1



CLASSIFIER MARGIN

x

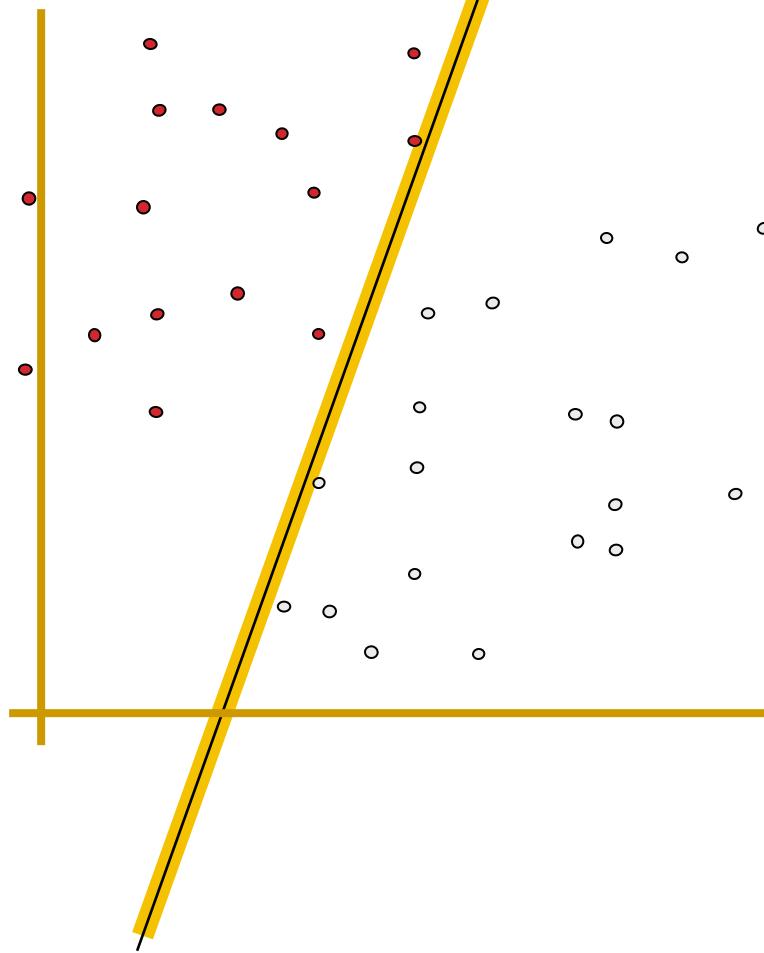
α

f

y^{est}

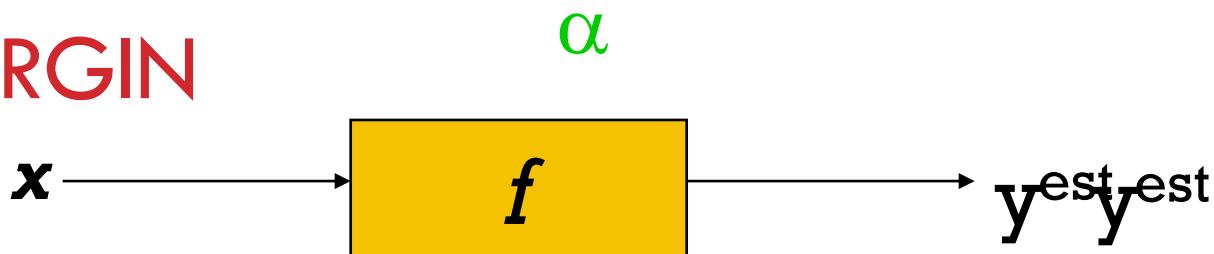
$$f(x, w, b) = \text{sign}(w \cdot x - b)$$

- denotes +1
- denotes -1

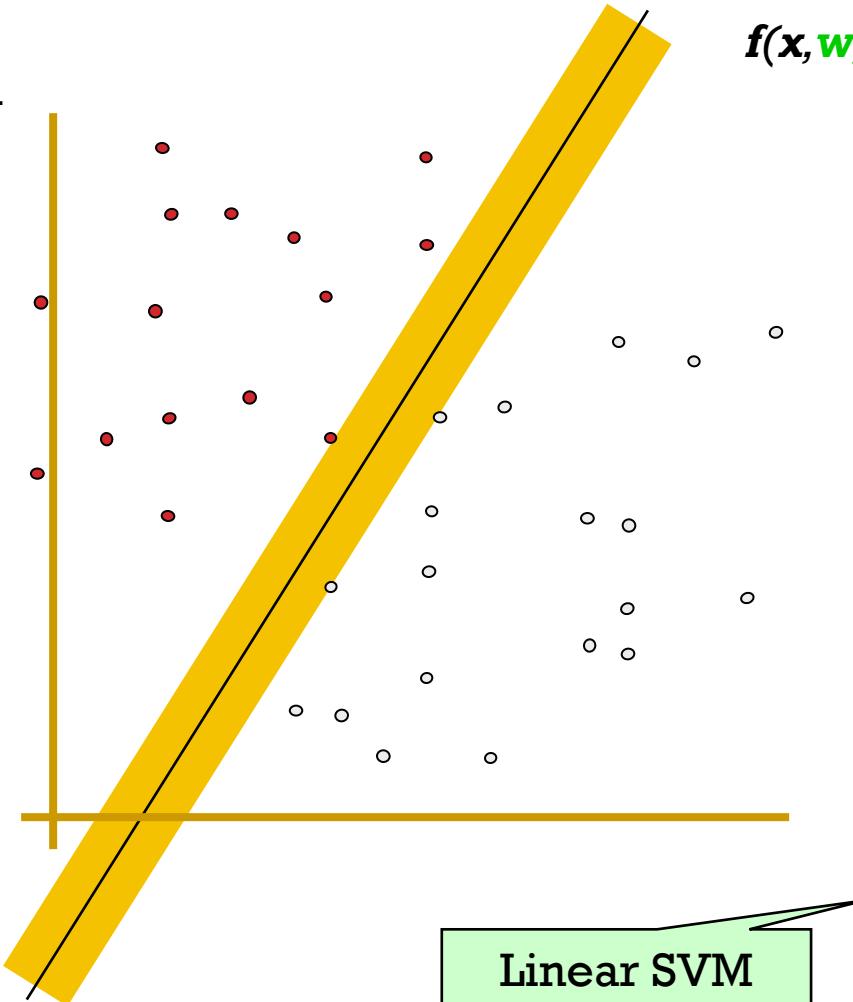


Define the **margin** of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

MAXIMUM MARGIN



- denotes +1
- denotes -1



The maximum margin linear classifier is the linear classifier with the, um, maximum margin. This is the simplest kind of SVM (Called an LSVM)

MAXIMUM MARGIN

x

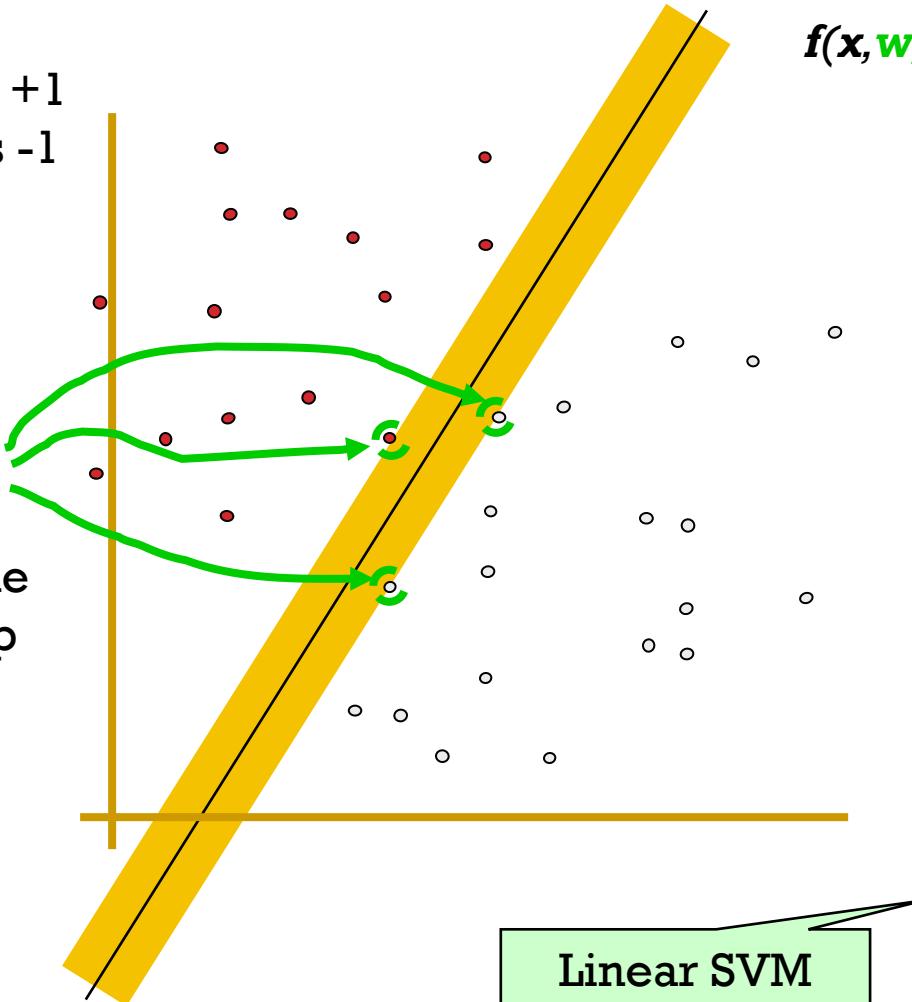
α

f

y^{est}

- denotes +1
- denotes -1

Support Vectors
are those
datapoints that the
margin pushes up
against

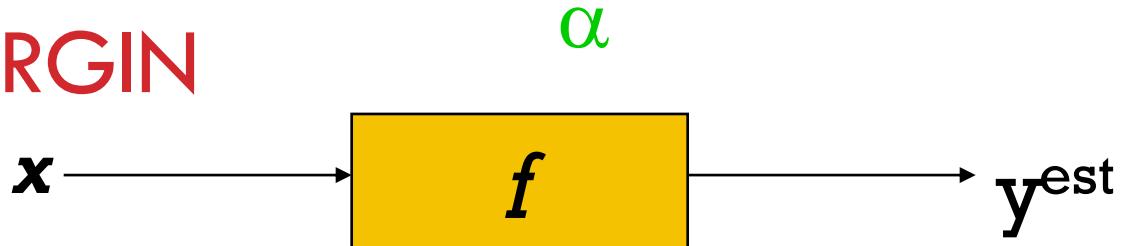


$$f(x, w, b) = \text{sign}(w \cdot x - b)$$

The maximum margin linear classifier is the linear classifier with the, um, maximum margin. This is the simplest kind of SVM (Called an LSVM)

Linear SVM

MAXIMUM MARGIN



- denotes +1
- denotes -1

$$f(\mathbf{x}, \mathbf{w}, b) = \text{sign}(\mathbf{w} \cdot \mathbf{x} - b)$$

The maximum

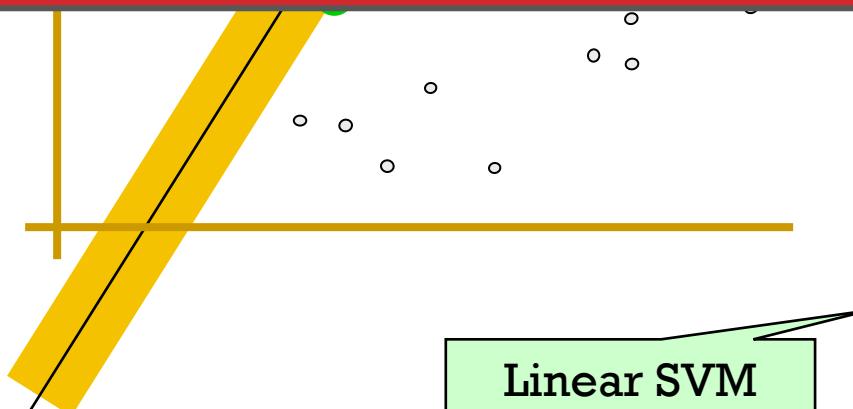
sklearn.linear_model.SGDClassifier

Default loss: “hinge” → linear SVM.

Support
are those
datapoint

margin pushes up
against

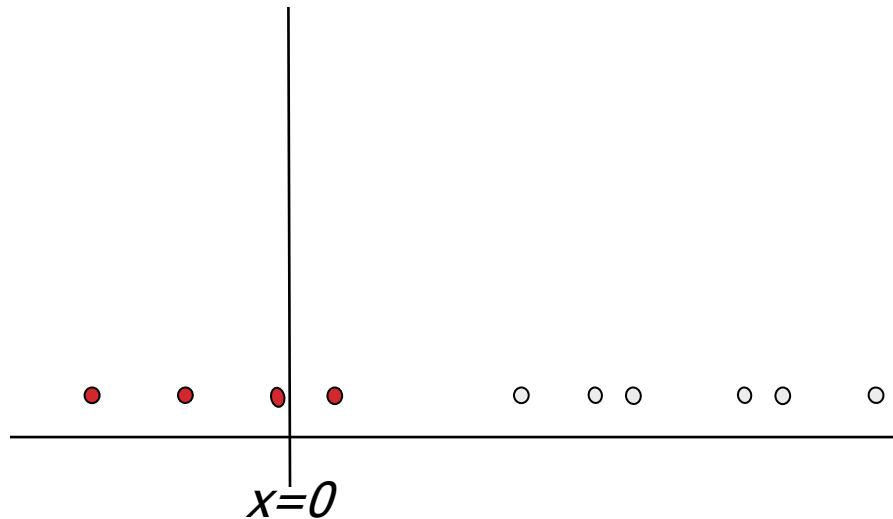
This is the
simplest kind of
SVM (Called an
LSVM)



Linear SVM

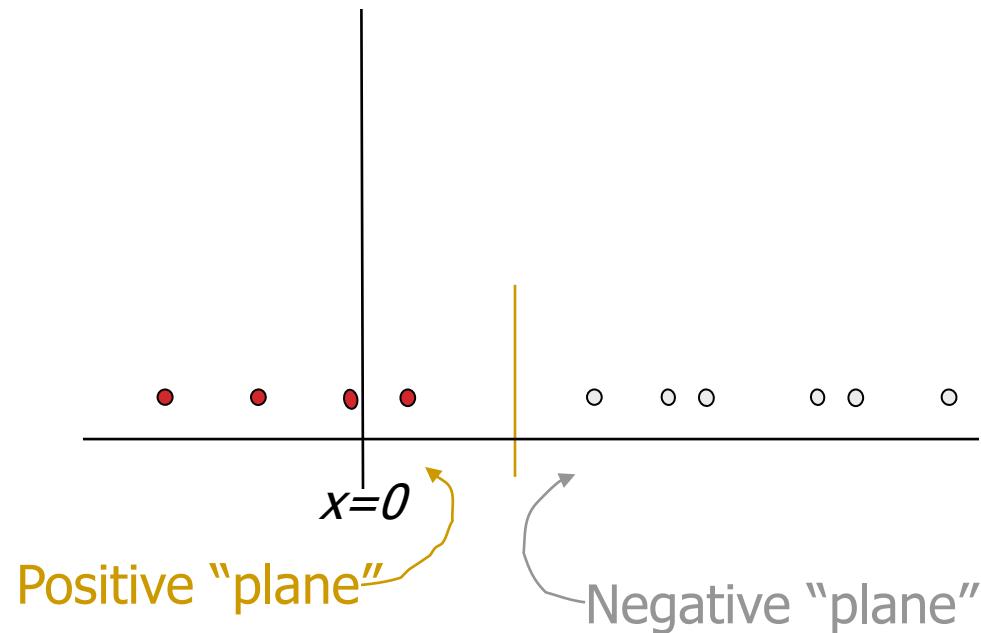
SUPPOSE WE'RE IN 1-DIMENSION

What would
SVMs do with
this data?



SUPPOSE WE'RE IN 1-DIMENSION

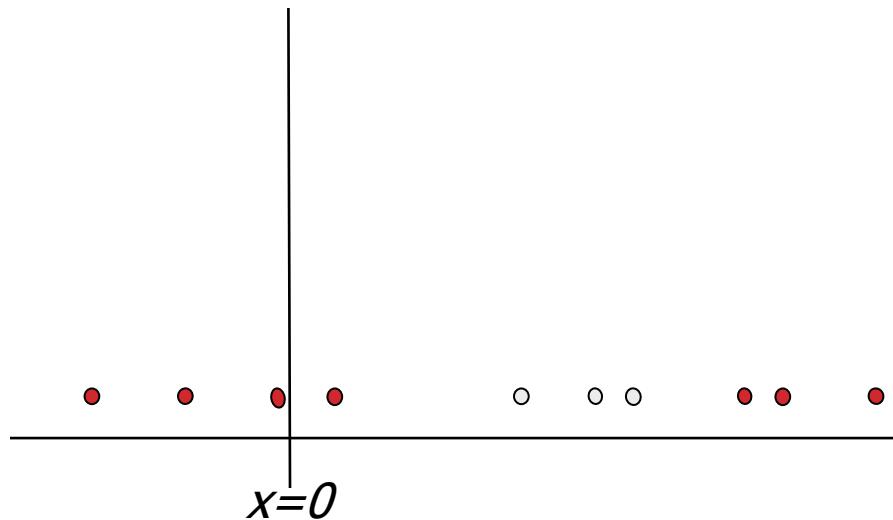
Not a big surprise



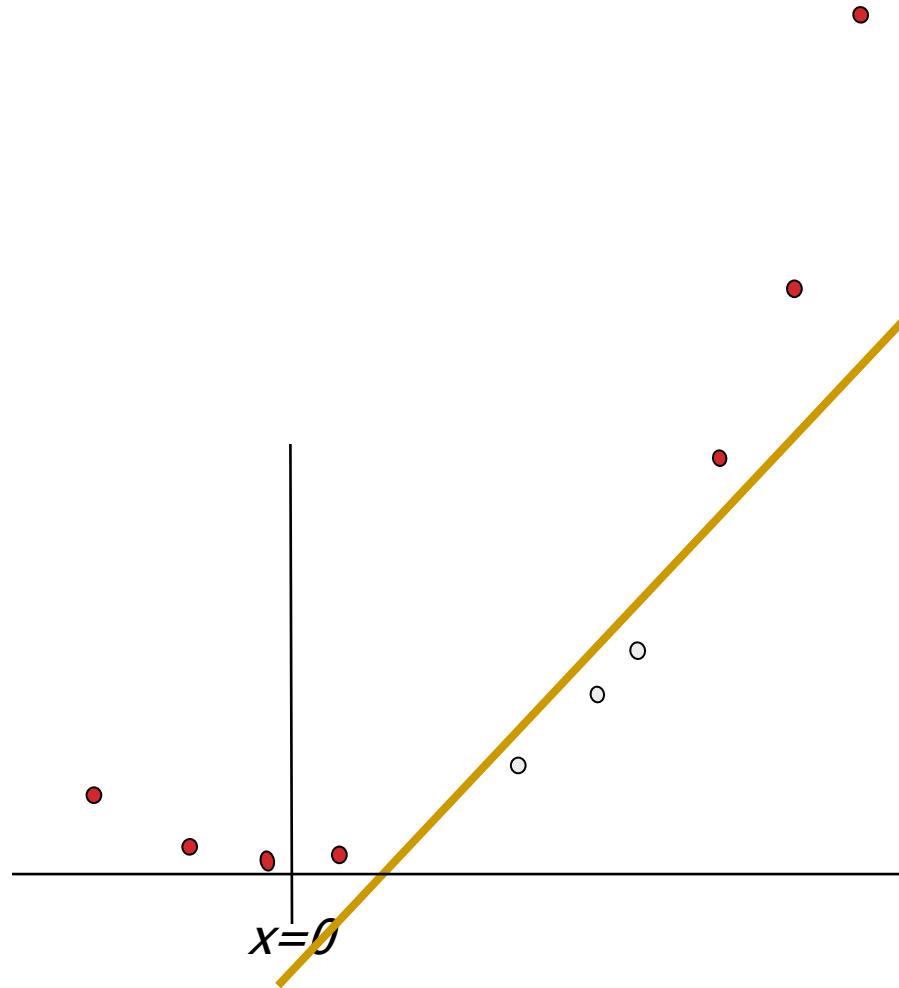
HARDER 1-DIMENSIONAL DATASET

That's wiped the smirk off SVM's face.

What can be done about this?



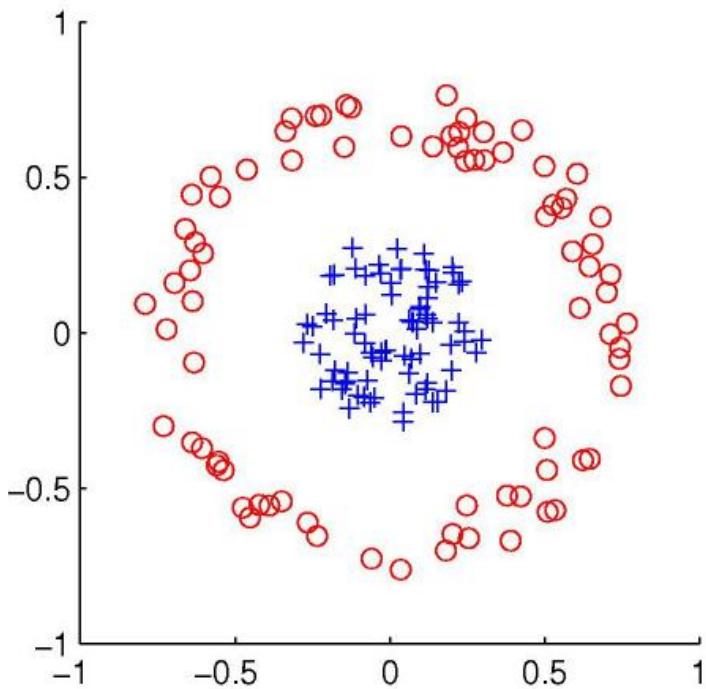
HARDER 1-DIMENSIONAL DATASET



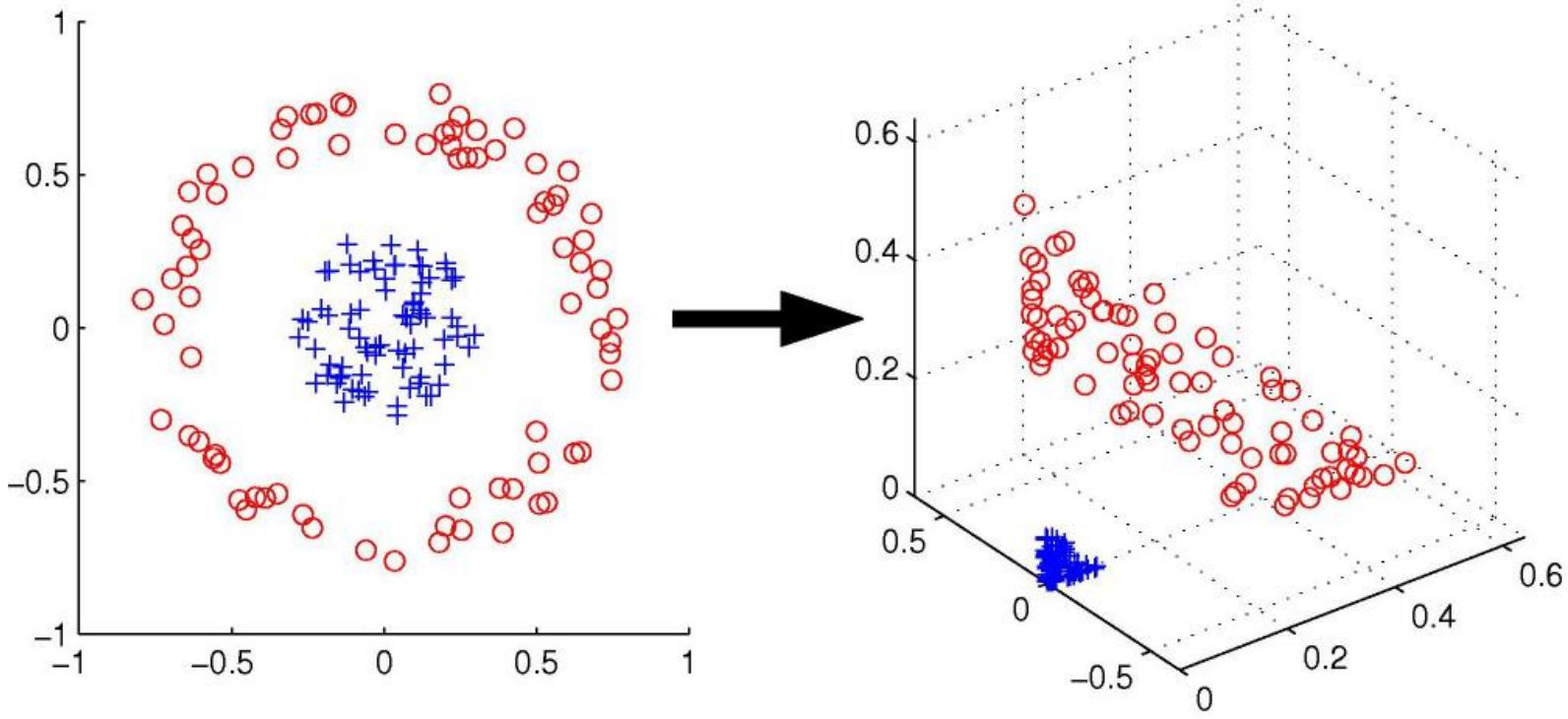
Permitting non-linear basis functions

$$\mathbf{z}_k = (x_k, x_k^2)$$

THE KERNEL TRICK



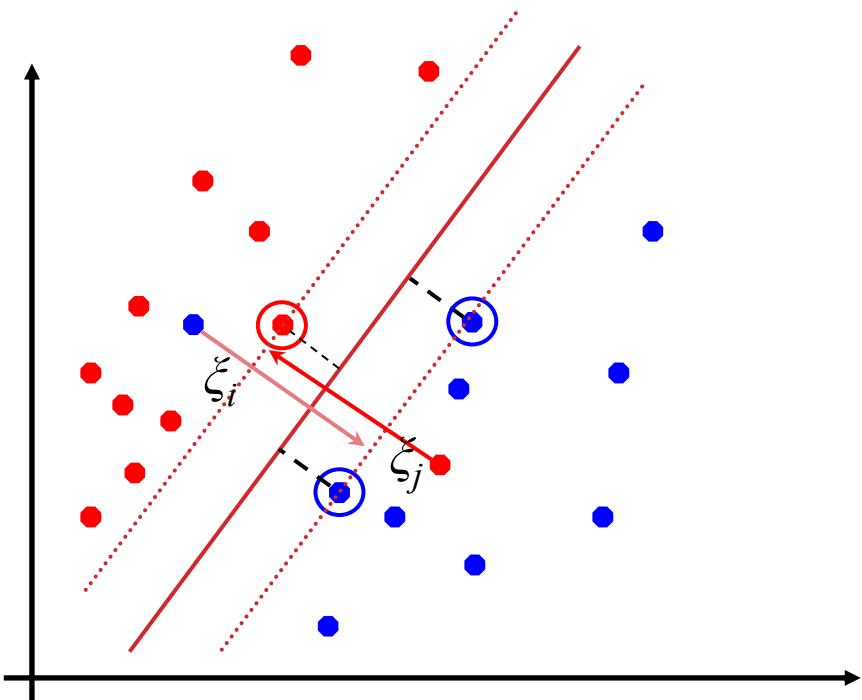
THE KERNEL TRICK



$$\begin{aligned}\phi : \quad \mathbb{R}^2 &\longrightarrow \mathbb{R}^3 \\ (x_1, x_2) &\longmapsto (z_1, z_2, z_3) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)\end{aligned}$$

[<http://www.cs.berkeley.edu/~jordan/courses/281B-spring04/lectures/lec3.pdf>]

SOFT MARGIN CLASSIFICATION

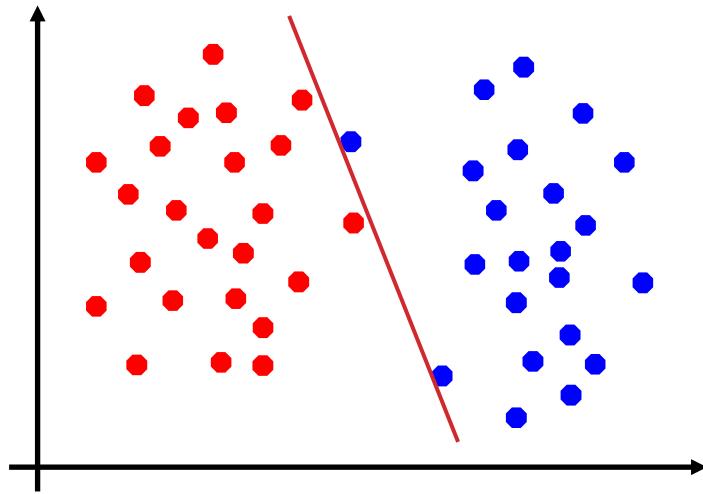


If the training data is not linearly separable, *slack variables* ξ_i (a **regularization parameter**) can be added to allow misclassification of difficult or noisy examples.

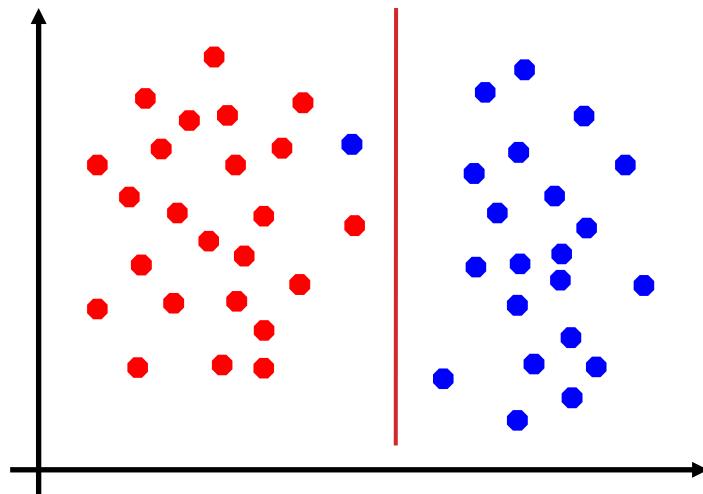
Still, try to minimize training set errors, and to place hyperplane “far” from each class (large margin)

THE IMPACT OF REGULARIZATION

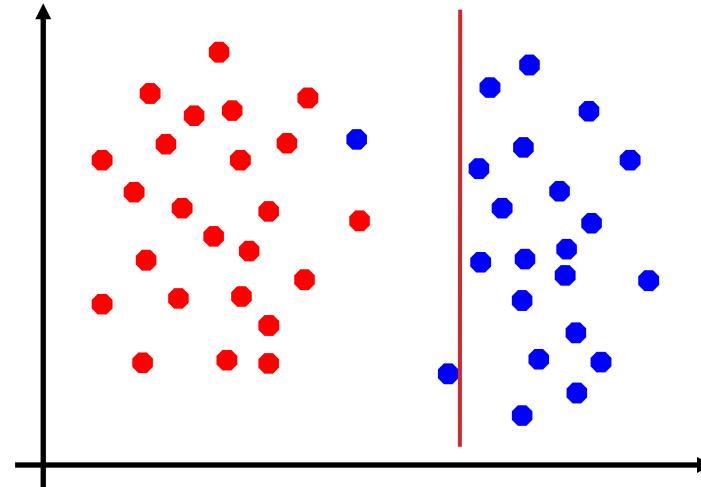
No regularization



Right amount



Too much



MANY CLASSIFIERS TO CHOOSE FROM

K-nearest neighbor

Support Vector Machines

Which is the best one?

Decision Trees

Random Forrest

(Gradient) Boosted Decision Trees

Logistic Regression

Naïve Bayes

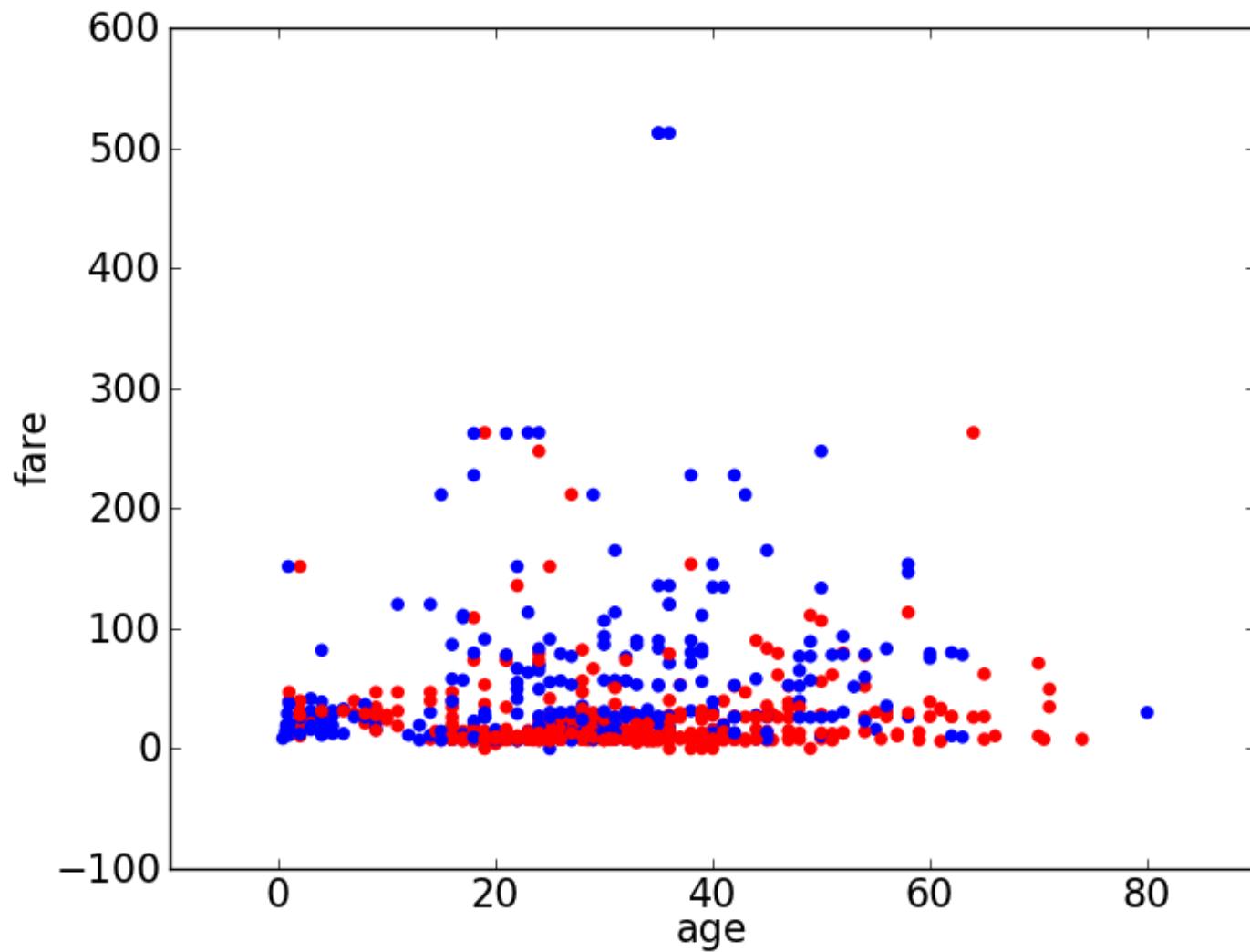
Bayesian network

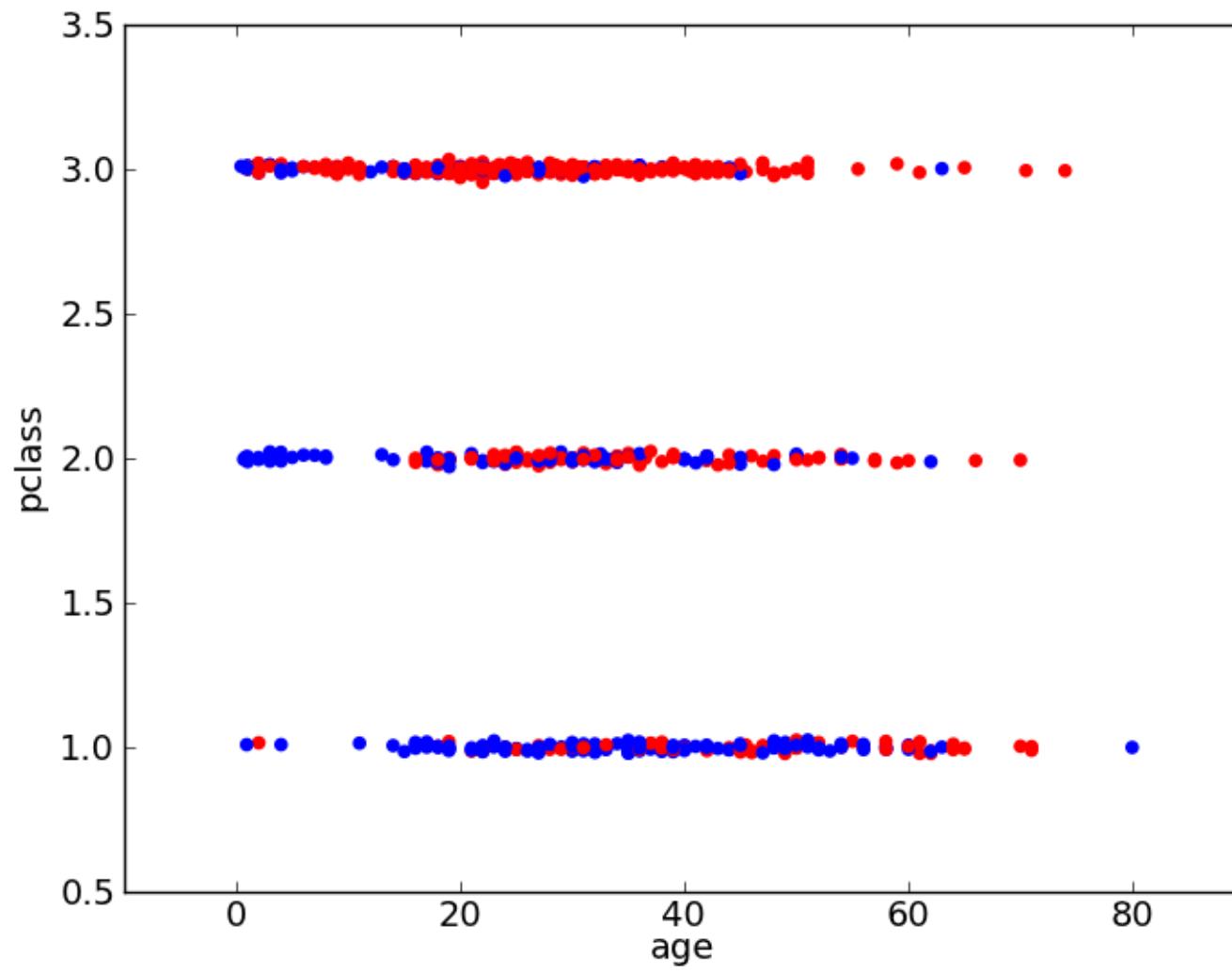
RBM_s

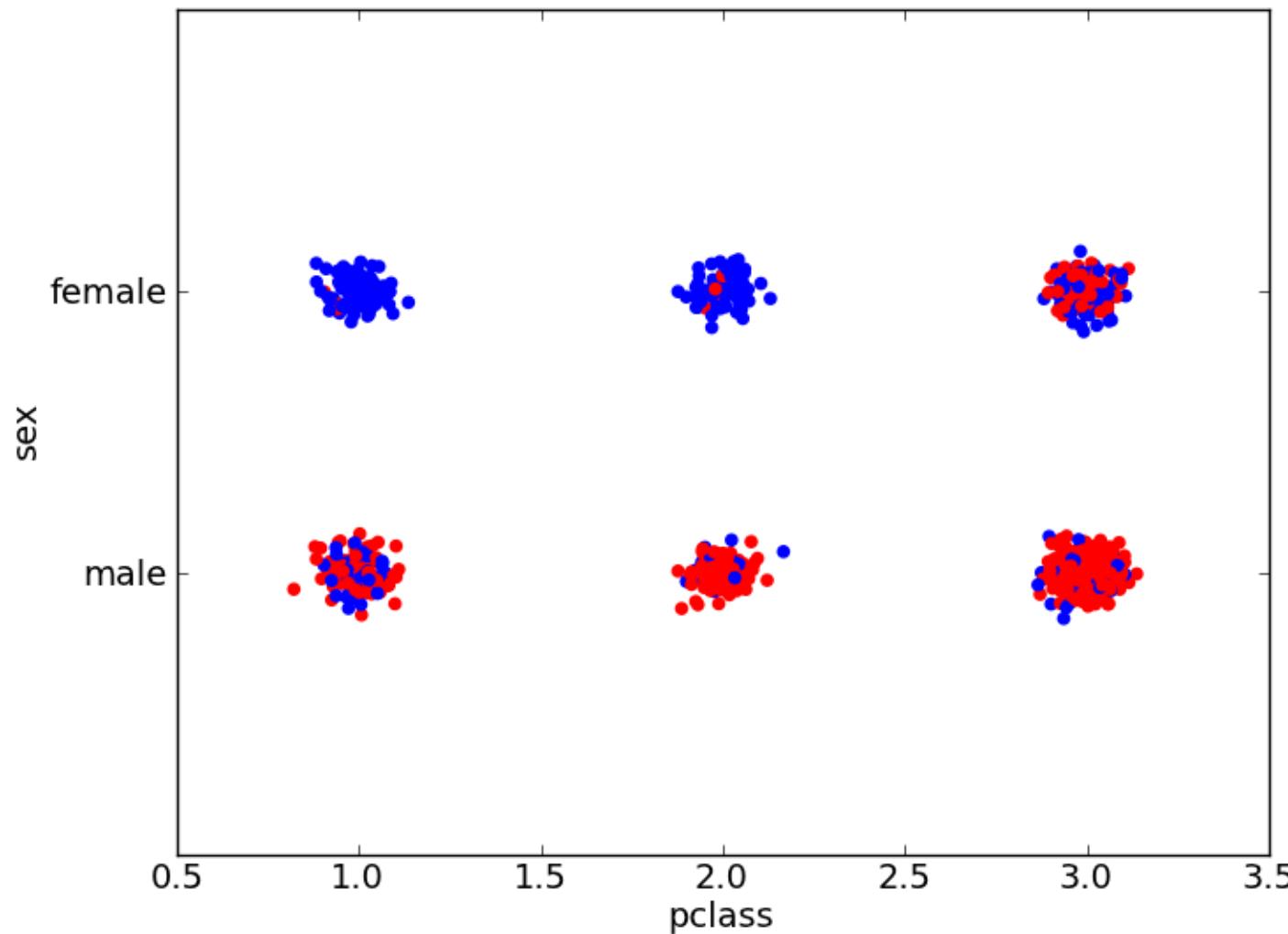
....

TITANIC DATASET

survived	pclass	sex	age	sibsp	parch	fare	cabin	embarked
0	3	male	22	1	0	7.25		S
1	1	female	38	1	0	71.2833	C85	C
1	3	female	26	0	0	7.925		S
1	1	female	35	1	0	53.1	C123	S
0	3	male	35	0	0	8.05		S
0	3	male		0	0	8.4583		Q
0	1	male	54	0	0	51.8625	E46	S
0	3	male	2	3	1	21.075		S
1	3	female	27	0	2	11.1333		S
1	2	female	14	1	0	30.0708		C
1	3	female	4	1	1	16.7	G6	S
1	1	female	58	0	0	26.55	C103	S
0	3	male	20	0	0	8.05		S







IF sex='female' THEN survive=yes

ELSE IF sex='male' THEN survive = no

confusion matrix

no	yes	<-- classified as
468	109	no
81	233	yes

$$(468 + 233) / (468+109+81+233) = 79\% \text{ correct (and 21\% incorrect)}$$

Not bad!

IF pclass='1' THEN survive=yes

ELSE IF pclass='2' THEN survive=yes

ELSE IF pclass='3' THEN survive=no

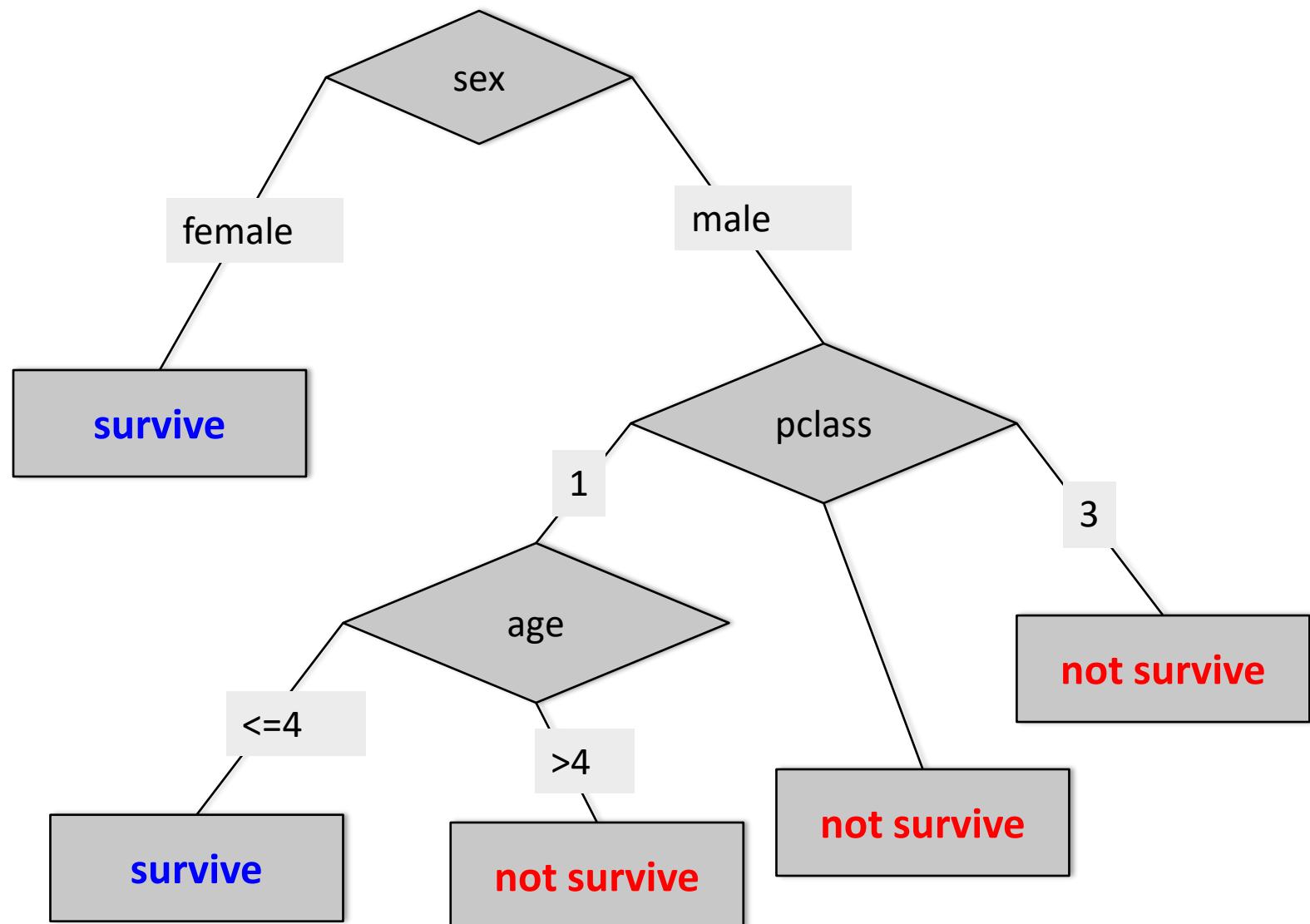
confusion matrix

no	yes	<-- classified as
372	119	no
177	223	yes

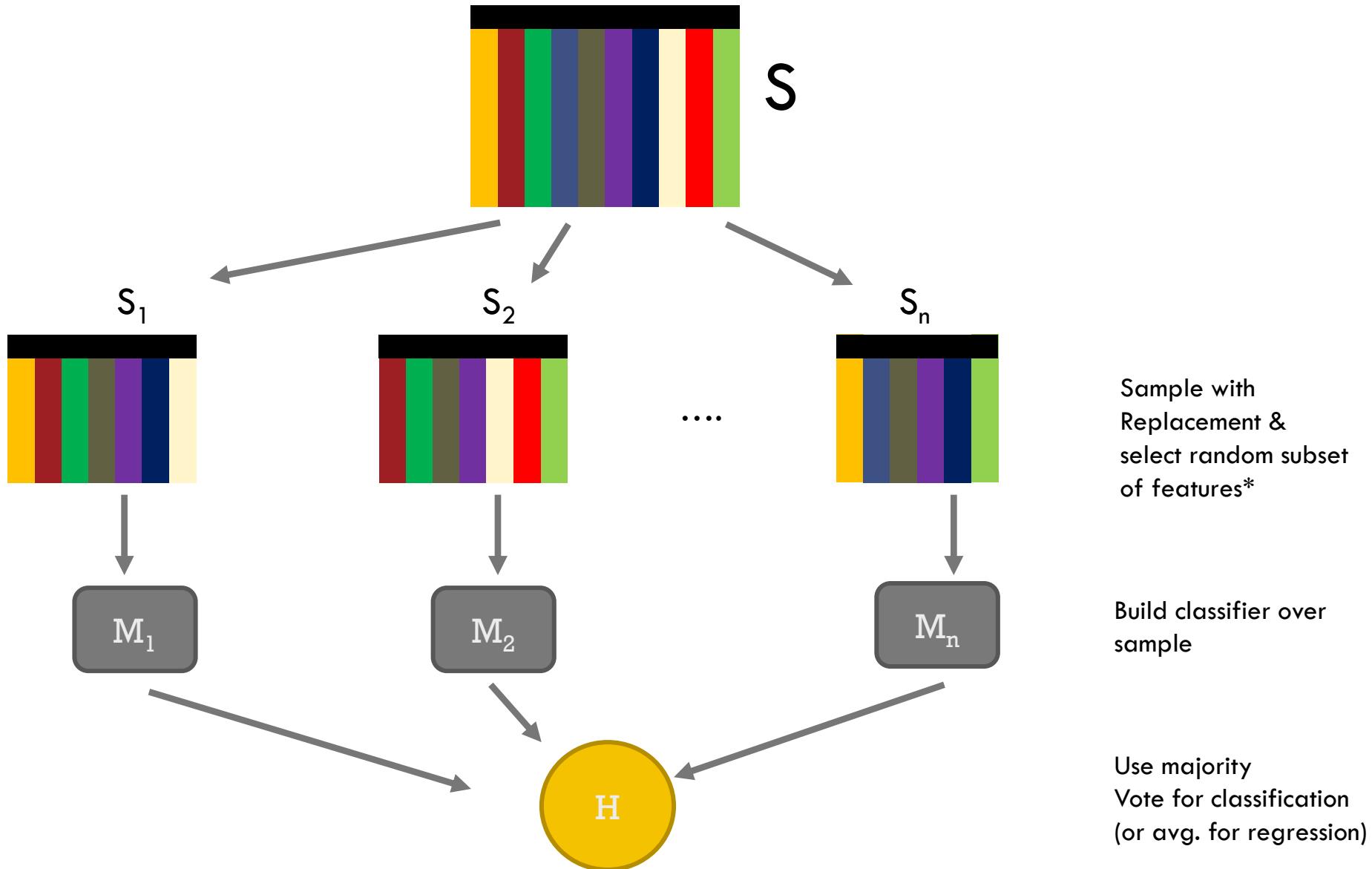
$$(372 + 223) / (372+119+223+177) = 67\% \text{ correct (and 33\% incorrect)}$$

a little worse

DECISION TREES

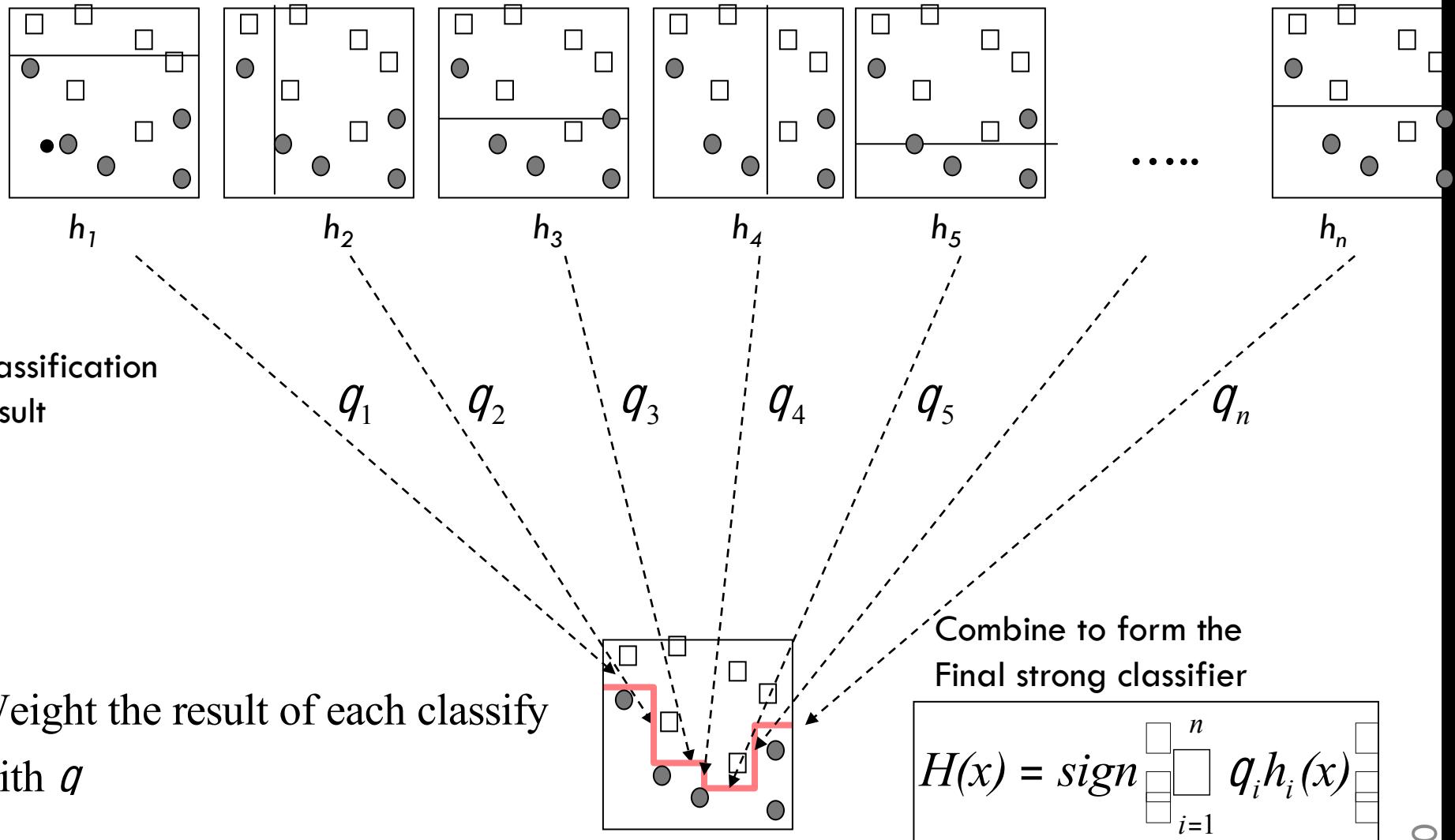


RANDOM FORREST



ADABOOST - CORE IDEA

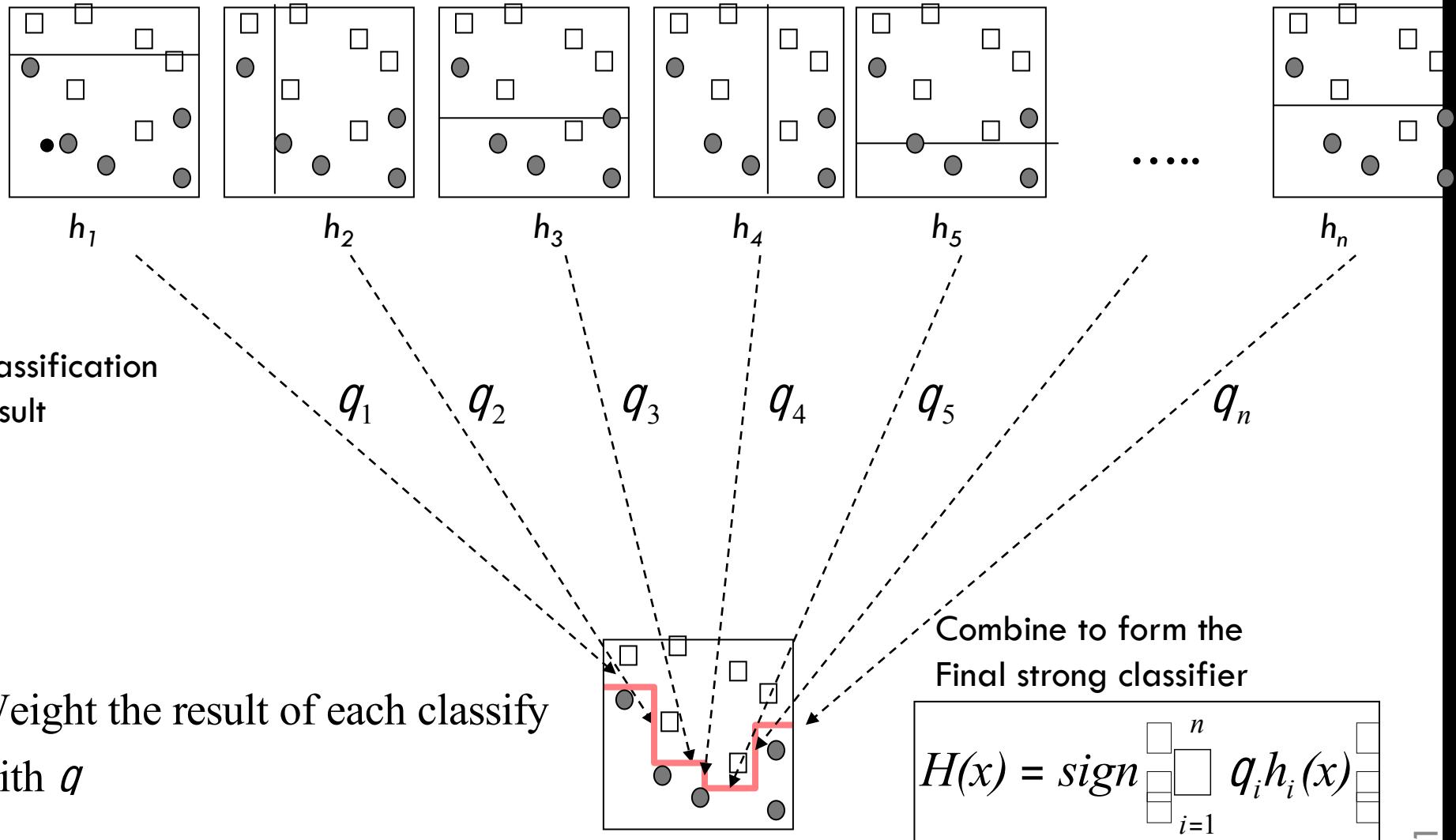
Take a set of **weak classifiers** (normally they should do better than guessing)



Weight the result of each classify with q

ADABOOST - CORE IDEA

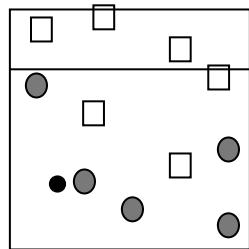
Take a set of **weak classifiers** (normally they should do better than guessing)



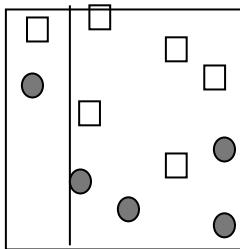
Weight the result of each classify with q

ADABOOST - CORE IDEA

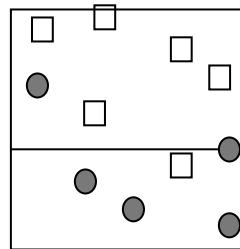
Take a set of **weak classifiers** (normally they should do better than guessing)



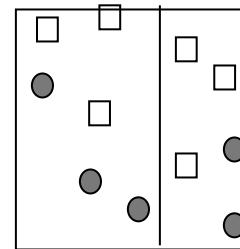
h_1



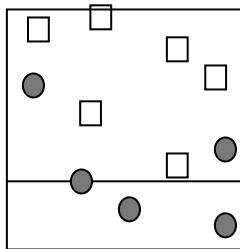
h_2



h_3

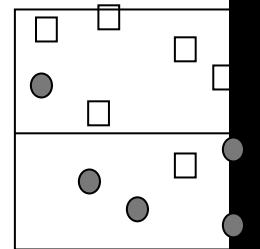


h_4



h_5

.....

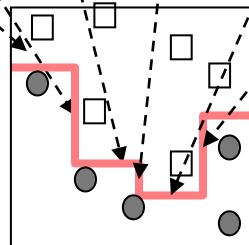


h_n

Classification
Result

XGBoost follows the same idea

Weight the result of each classify
with q



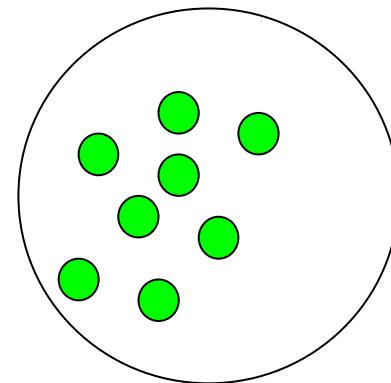
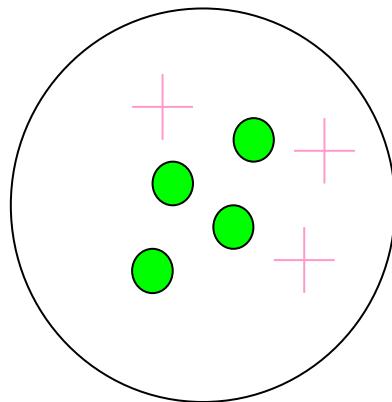
Combine to form the
Final strong classifier

$$H(x) = \text{sign} \sum_{i=1}^n q_i h_i(x)$$

ASIDE ON ENTROPY

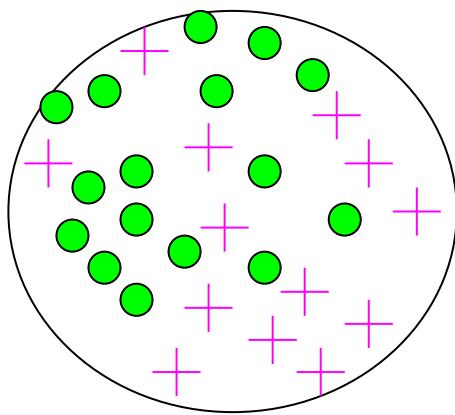
Impurity/Entropy (informal)

- Measures the level of **impurity** in a group of examples

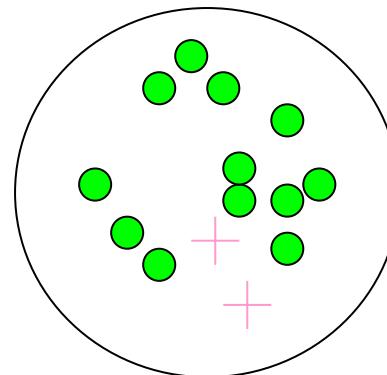


IMPURITY

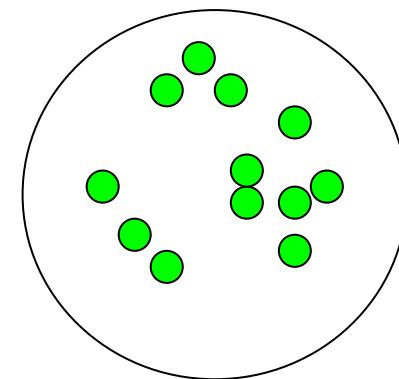
Very impure group



Less impure



Minimum impurity

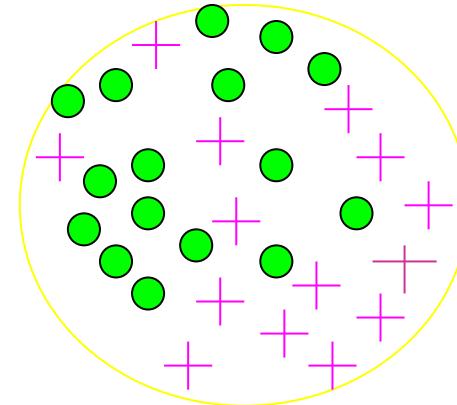


ENTROPY

- Entropy = $\sum_i p_i \log_2 p_i$

p_i is the probability of class i

Compute it as the proportion of class i in the set.



16/30 are green circles; 14/30 are pink crosses

$$\log_2(16/30) = -.9; \quad \log_2(14/30) = -1.1$$

$$\text{Entropy} = -(16/30)(-.9) - (14/30)(-1.1) = .99$$

- Entropy comes from information theory. The higher the entropy the more the information content.

What does that mean for learning from examples?

CLICKER

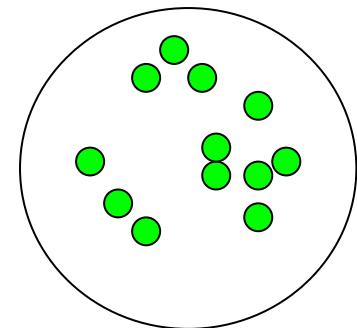
What is the entropy if all examples belong to the same class?

- a) 0
- b) 1
- c) Infinite

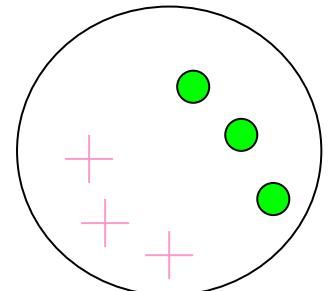
2 CLASS EXAMPLE

- What is the entropy of a group in which all examples belong to the same class?
- What is the entropy of a group with 50% in either class?

**Minimum
impurity**



**Maximum
impurity**



EXAMPLE: ROLLING A DIE

$$p_1 = \frac{1}{6}, p_2 = \frac{1}{6}, p_3 = \frac{1}{6}, \dots$$

$$\begin{aligned}\text{Entropy} &= - \sum_i p_i \log_2 p_i \\ &= -6 \times \left(\frac{1}{6} \log_2 \frac{1}{6} \right) \\ &\approx 2.58\end{aligned}$$

CLICKER

Has an unfair/weighted die a higher or lower entropy?

- A) Higher
- B) Lower

EXAMPLE: ROLLING A WEIGHTED DIE

$$p_1 = 0.1, p_2 = 0.1, p_3 = 0.1, \dots p_6 = 0.5$$

$$\begin{aligned}\text{Entropy} &= - \sum_i p_x \log_2 p_x \\ &= -5 \times (0.1 \log_2 0.1) - 0.5 \log_2 0.5 \\ &= 2.16\end{aligned}$$

The weighted die is **has less uncertainty** than a fair die

HOW UNCERTAIN IS YOUR DATA?

342/891 survivors in titanic training set

$$-\left(\frac{342}{891} \log_2 \frac{342}{891} + \frac{549}{891} \log_2 \frac{549}{891}\right) = 0.96$$

Say there were only 50 survivors

$$-\left(\frac{50}{891} \log_2 \frac{50}{891} + \frac{841}{891} \log_2 \frac{841}{891}\right) = 0.31$$

IN CLASS TASK

How can you use Entropy to build a decision tree.

Discuss with your neighbor(s)

Discuss the following ideas

Select the feature based on the highest entropy

Select the feature based on the lowest entropy

Stop splitting if the entropy is 0

Select the feature based on the entropy after the split

What if one group is under-/over represented

BACK TO DECISION TREES

Which attribute do we choose at each level?

The one with the highest information gain

- The one that reduces the uncertainty/impurity the most

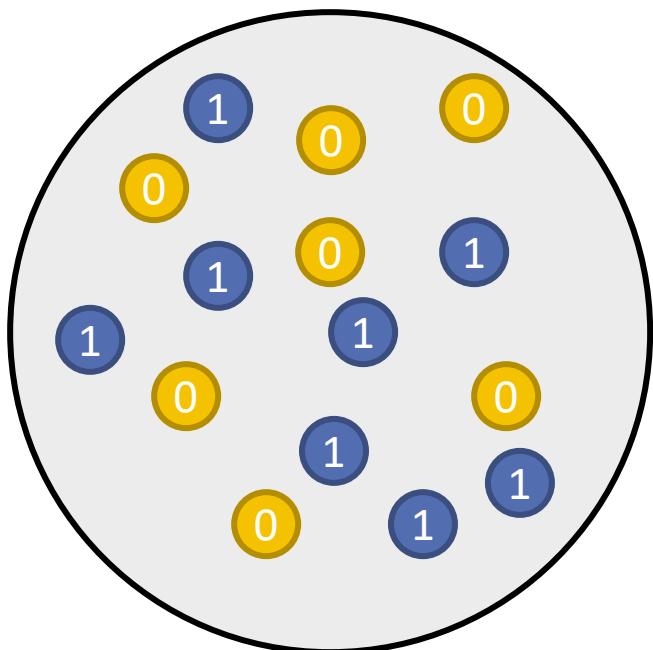
We want to determine which attribute in a given set of training feature vectors is most useful for discriminating between the classes to be learned.

Information gain tells us how important a given attribute of the feature vectors is.

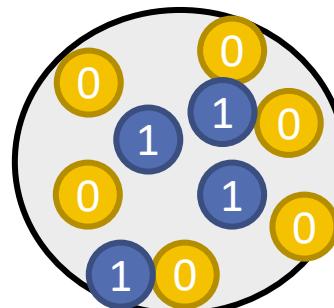
We will use it to decide the ordering of attributes in the nodes of a decision tree.

INFORMATION GAIN

Titanic Entropy = 0.96

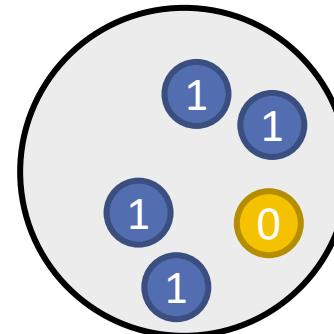


male



$$\begin{aligned} \text{Entropy} &= -\frac{682}{843} \log\left(\frac{682}{843}\right) \\ &\quad - \frac{161}{843} \log\left(\frac{161}{843}\right) \\ &= 0.21 \end{aligned}$$

female



$$\begin{aligned} \text{Entropy} &= -\frac{127}{466} \log\left(\frac{127}{466}\right) \\ &\quad - \frac{339}{466} \log\left(\frac{339}{466}\right) \\ &= 0.25 \end{aligned}$$

$$\text{Weighted Entropy: } \frac{466}{1309} * 0.25 + \frac{843}{1309} * 0.21 = 0.22$$

$$\text{Information Gain for split: } 0.96 - 0.22 = 0.74$$

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

If we choose **outlook**:

overcast : 4 records, 4 are “yes”

$$-\left(\frac{4}{4} \log_2 \frac{4}{4}\right) = 0$$

rainy : 5 records, 3 are “yes”

$$-\left(\frac{3}{5} \log_2 \frac{3}{5} + \frac{2}{5} \log_2 \frac{2}{5}\right) = 0.97$$

sunny : 5 records, 2 are “yes”

$$-\left(\frac{2}{5} \log_2 \frac{2}{5} + \frac{3}{5} \log_2 \frac{3}{5}\right) = 0.97$$

Expected new entropy:

$$\frac{4}{14} \times 0.0 + \frac{5}{14} \times 0.97 + \frac{5}{14} \times 0.97$$

$$= 0.69$$

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are "yes"

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

Clicker:

If we choose windy, what is the expected entropy?

a) 0.81

$$= - (6/8 \log(6/8) + 2/8 \log(2/8))$$

b) 0.89

$$= 6/14 * 1 + (-8/14 * (6/8 \log(6/8) + 2/8 \log(2/8)))$$

c) 1

$$= - (0.5 * \log(0.5) + 0.5 * \log(0.5))$$

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are "yes"

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

If we choose **windy**:

FALSE: 8 records, 6 are "yes"

$$0.81 = -(6/8 * \log(6/8) + 2/8 * \log(2/8))$$

TRUE: 6 records, 3 are "yes"

1

Expected new entropy:

$$0.81(8/14) + 1(6/14)$$

$$= \underline{0.89}$$

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

If we choose **temperature**:

cool : 4 records, 3 are “yes”

0.81

rainy : 4 records, 2 are “yes”

1.0

sunny : 6 records, 4 are “yes”

0.92

Expected new entropy:

$$0.81(4/14) + 1.0(4/14) + 0.92(6/14)$$

= 0.91

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are "yes"

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

If we choose **humidity**:

normal : 7 records, 6 are "yes"

0.59

high : 7 records, 2 are "yes"

0.86

Expected new entropy:

$$0.59(7/14) + 0.86(7/14)$$

$$= \underline{0.725}$$

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are "yes"

$$-\left(\frac{9}{14} \log_2 \frac{9}{14} + \frac{5}{14} \log_2 \frac{5}{14}\right) = 0.94$$

outlook

$$0.94 - 0.69 = 0.25 \quad \text{highest gain}$$

temperature

$$0.94 - 0.91 = 0.03$$

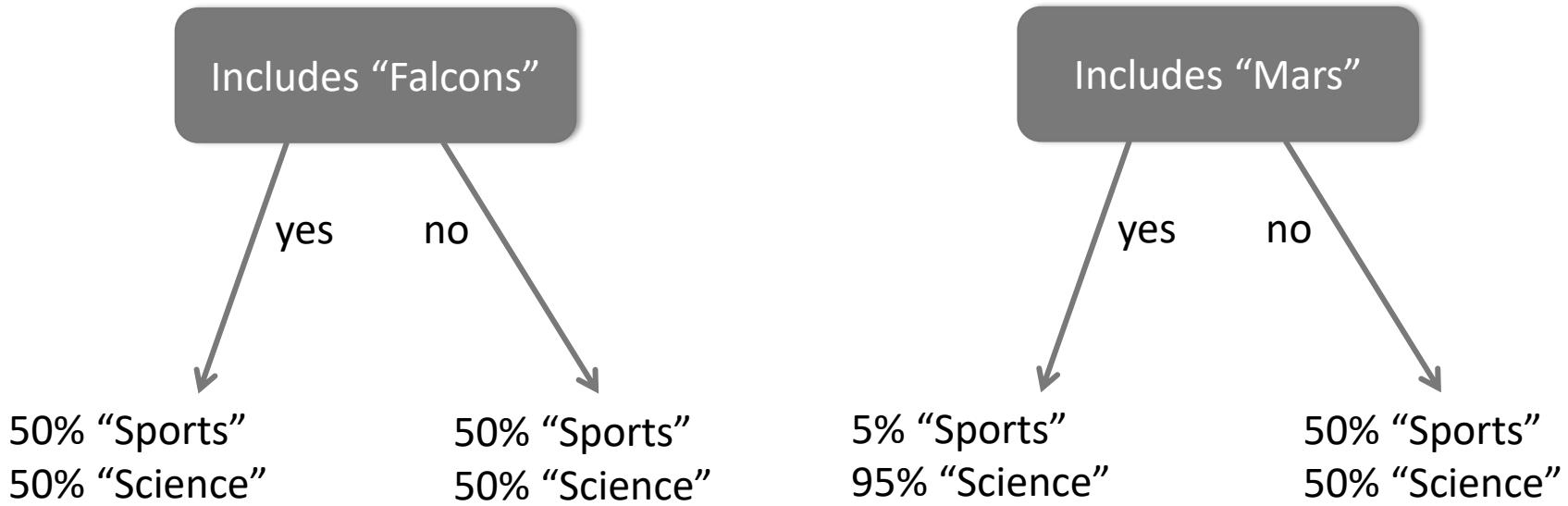
humidity

$$0.94 - 0.725 = 0.215$$

windy

$$0.94 - 0.87 = 0.07$$

DOCUMENT CLASSIFICATION



Clicker Question (assuming equal size):
a) Falcon's Information Gain is higher
b) Mars' Information Gain is higher

BUILDING A DECISION TREE (ID3 ALGORITHM)

Assume attributes are discrete

- Discretize continuous attributes

Choose the attribute with the highest Information Gain

Create branches for each value of attribute

Examples partitioned based on selected attributes

Repeat with remaining attributes

Stopping conditions

- All examples assigned the same label
- No examples left

PROBLEMS

Expensive to train

Prone to overfitting

- Drive to perfection on training data, bad on test data
- Pruning can help: remove or aggregate subtrees that provide little discriminatory power (C45)

C4.5 EXTENSIONS

Continuous Attributes

outlook	temperature	humidity	windy	play
overcast	cool	60	TRUE	yes
overcast	hot	80	FALSE	yes
overcast	hot	63	FALSE	yes
overcast	mild	81	TRUE	yes
rainy	cool	58	TRUE	no
rainy	mild	90	TRUE	no
rainy	cool	54	FALSE	yes
rainy	mild	92	FALSE	yes
rainy	mild	59	FALSE	yes
sunny	hot	90	FALSE	no
sunny	hot	89	TRUE	no
sunny	mild	90	FALSE	no
sunny	cool	60	FALSE	yes
sunny	mild	62	TRUE	yes

Consider every possible binary partition; choose the partition with the highest gain

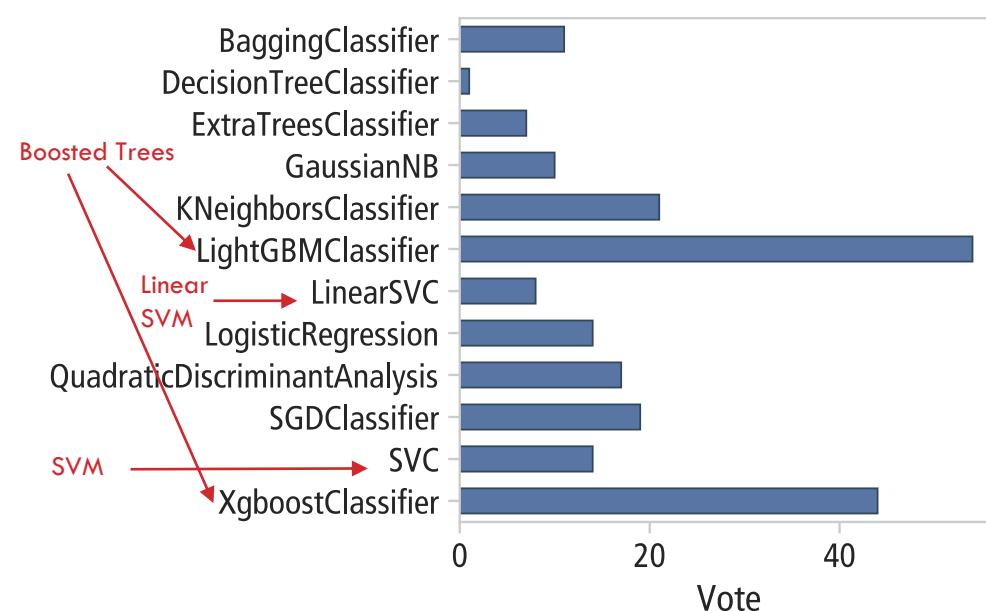
outlook	temperature	humidity	windy	play		
rainy	mild	54	FALSE	yes	E(6/6)	= 0.0
overcast	hot	58	FALSE	yes		
overcast	cool	59	TRUE	yes		
rainy	cool	60	FALSE	yes		
overcast	mild	60	TRUE	yes		
overcast	hot	62	FALSE	yes		
rainy	mild	63	TRUE	no		
sunny	cool	80	FALSE	yes		
rainy	mild	81	FALSE	yes		
sunny	mild	89	TRUE	yes		
sunny	hot	90	FALSE	no		
rainy	cool	90	TRUE	no		
sunny	hot	90	TRUE	no		
sunny	mild	92	FALSE	no		

$$\text{Expect} = \frac{8}{14} * 0.95 + \frac{6}{14} * 0 \\ = 0.54$$

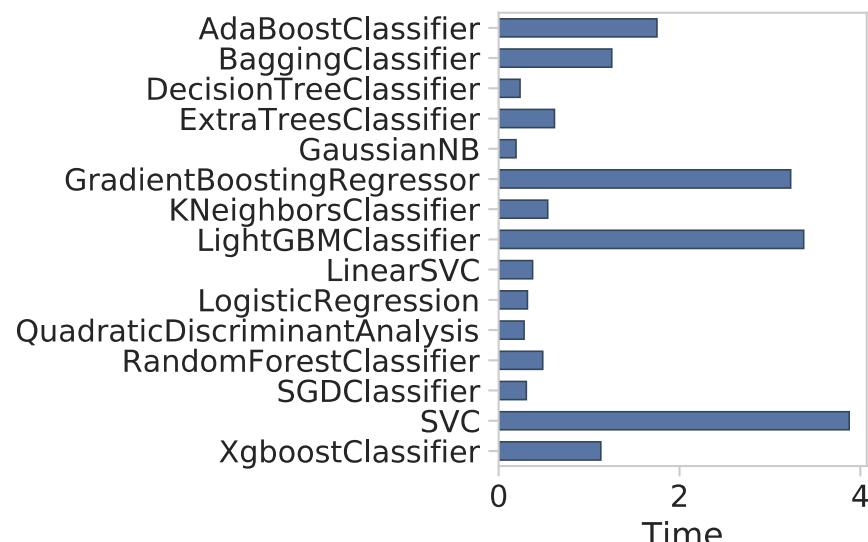
$$\text{Expect} = \frac{10}{14} * 0.47 + \frac{4}{14} * 0^{47} \\ = 0.33$$

PERFORMANCE OF DIFFERENT ML MODEL FAMILIES

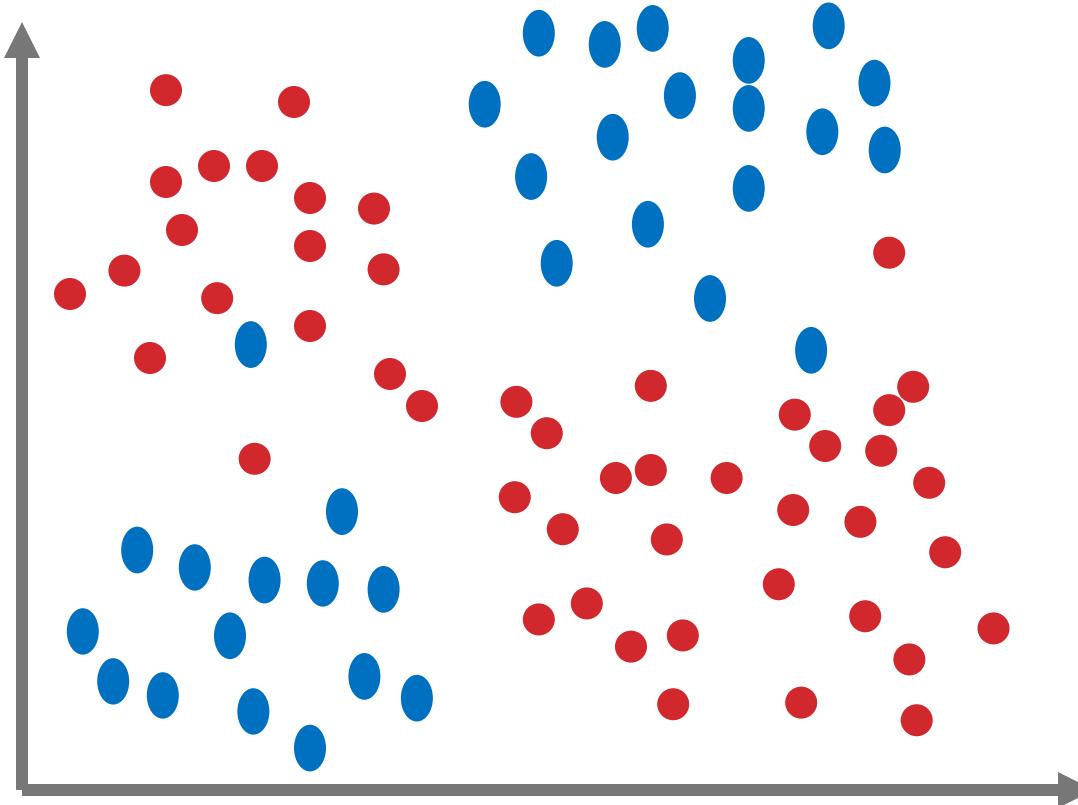
How often ranked 1st



Relative Training Time



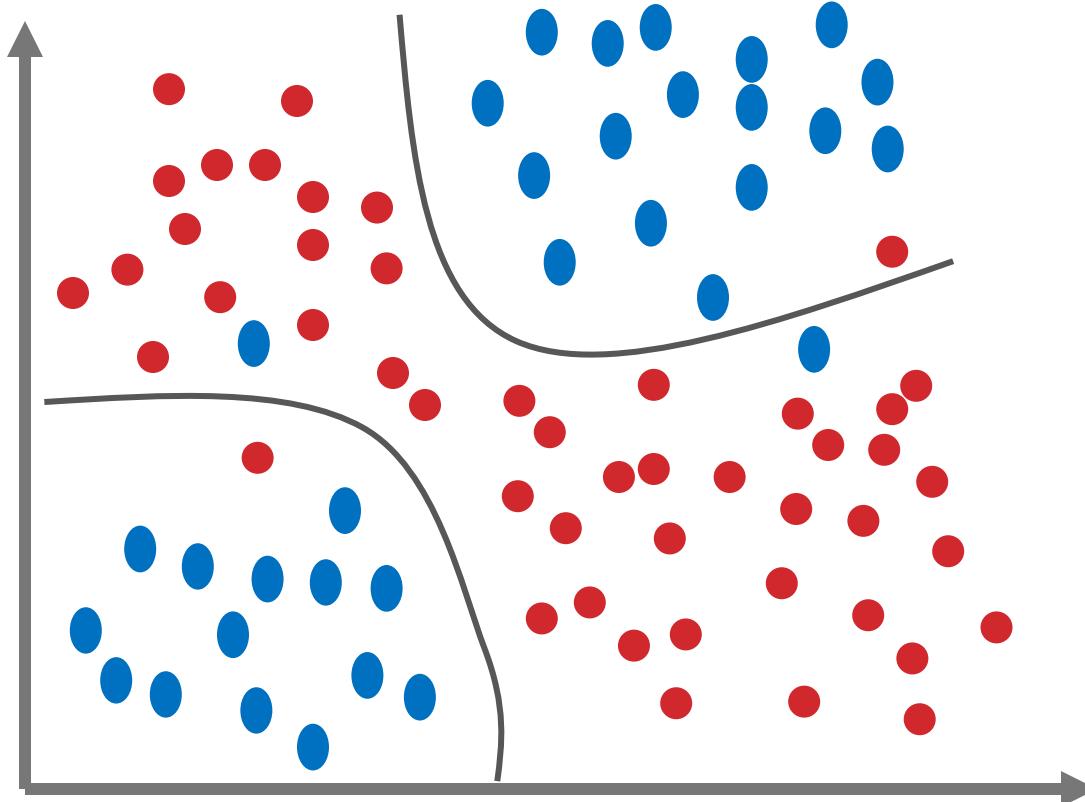
IN-CLASS TASK



How would you draw the expected decision boundary for

- Random Forrest
- SVM w/ kernel and regularization
- 1-KNN

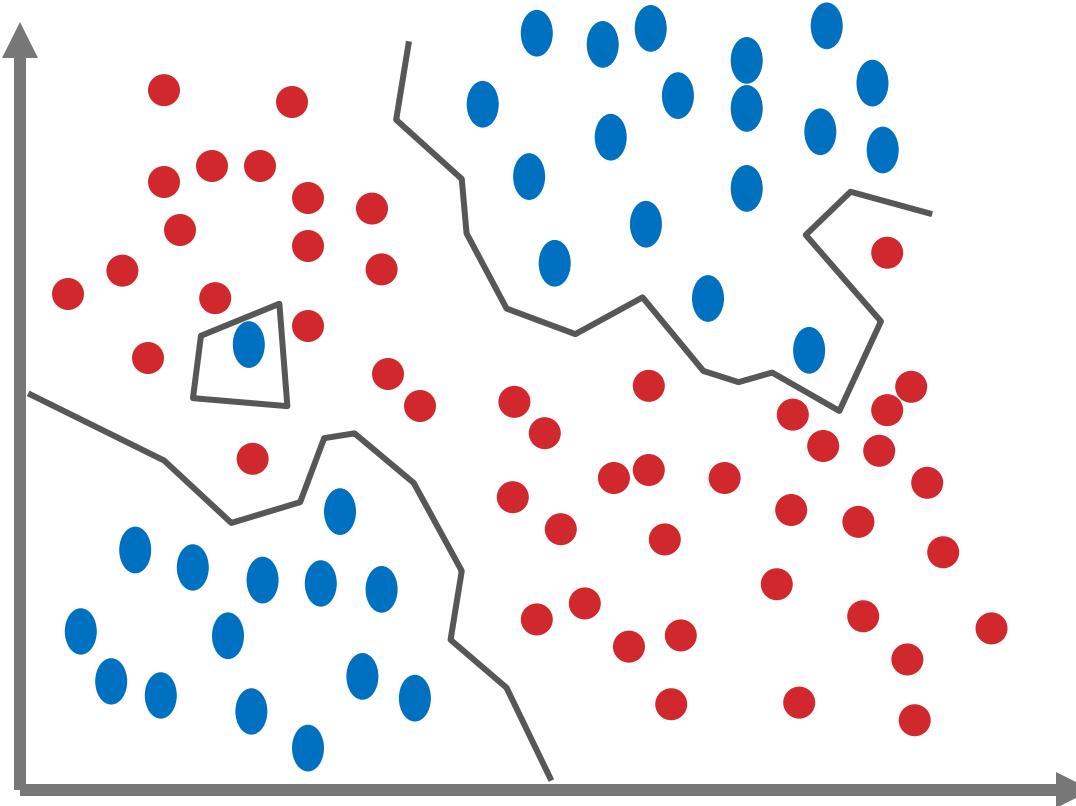
CLICKER



The decision boundary looks like the one of:

- a) Random Forrest
- b) SVM w/ kernel and regularization
- c) 1-KNN

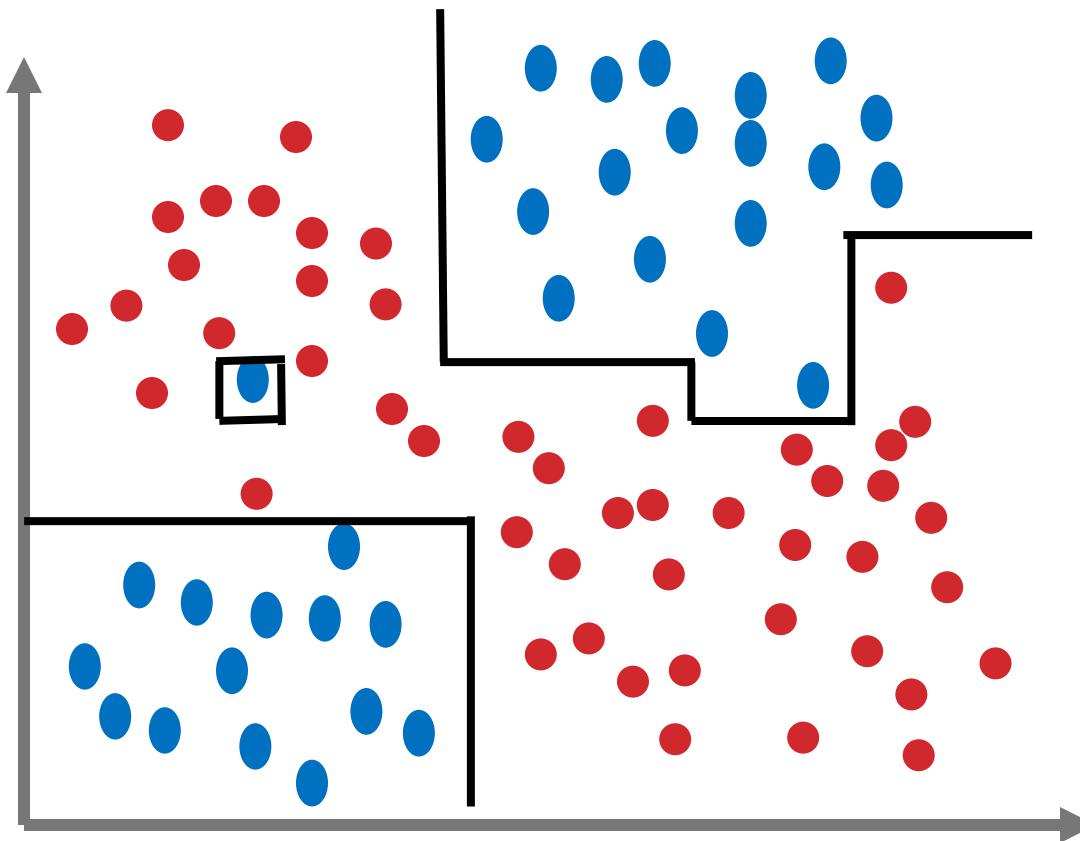
CLICKER



The decision boundary looks like the one of:

- a) Random Forrest
- b) SVM w/ kernel and regularization
- c) 1-KNN

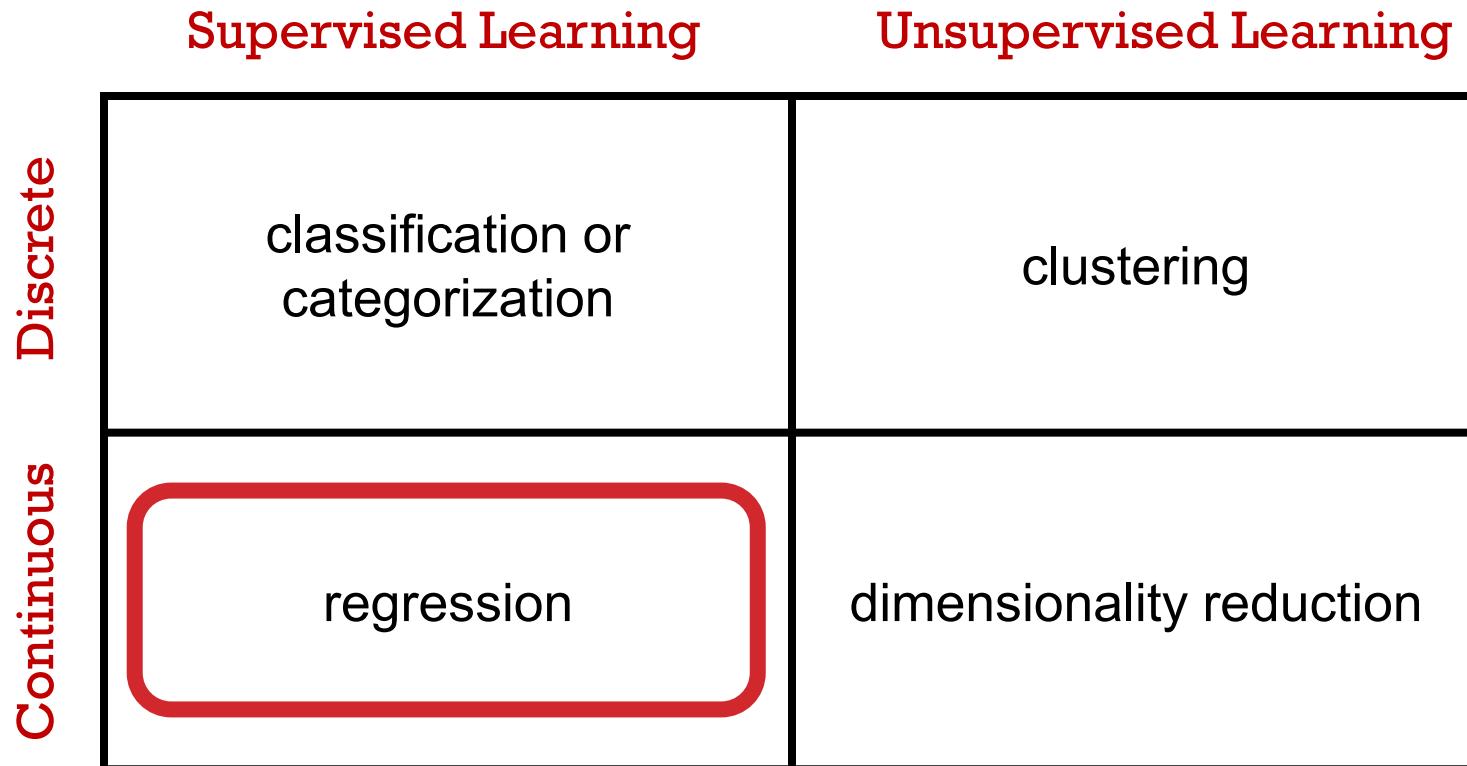
RANDOM FORREST



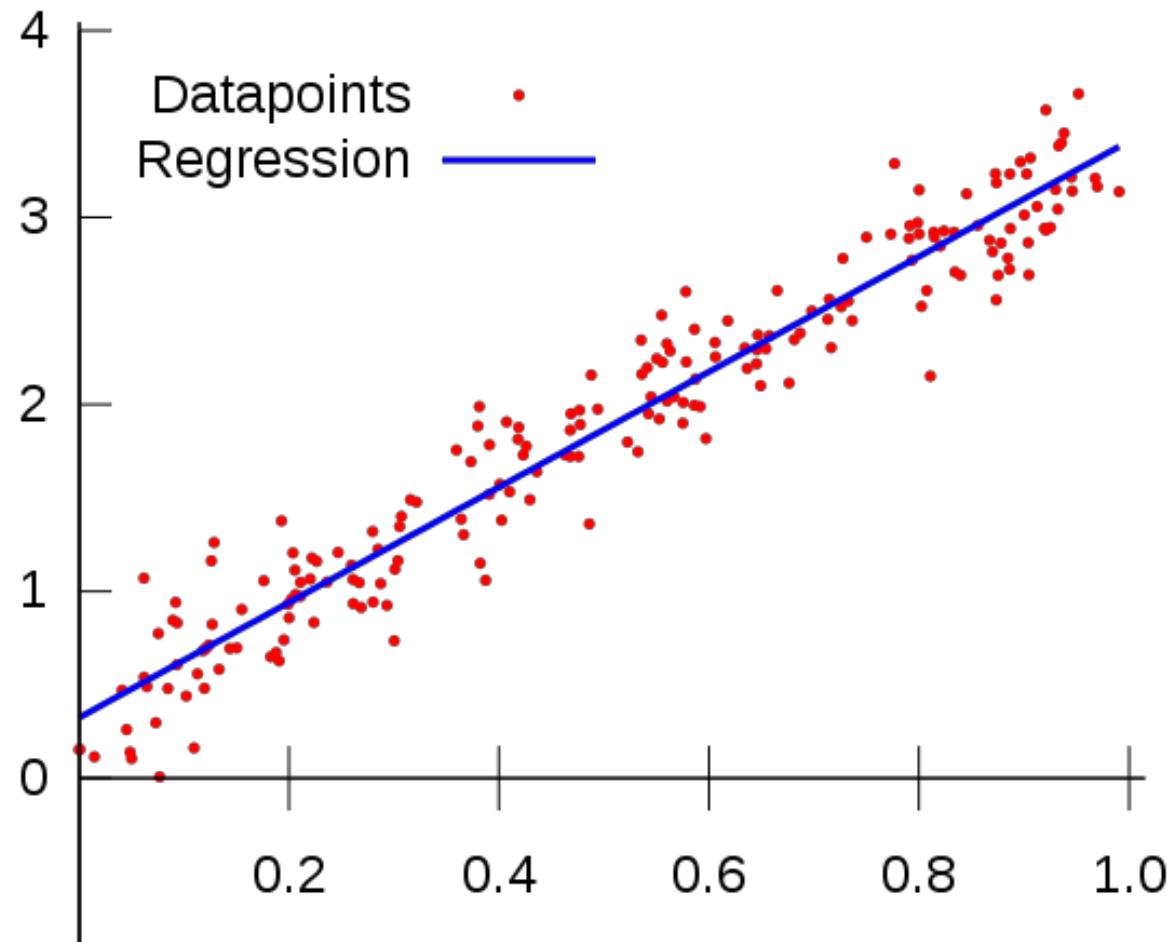
The decision boundary looks like the one of:

- a) Random Forrest
- b) SVM w/ kernel and regularization
- c) 1-KNN

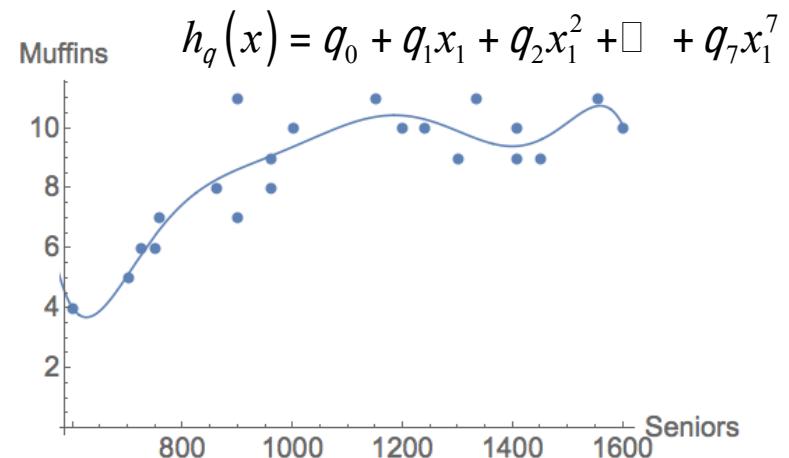
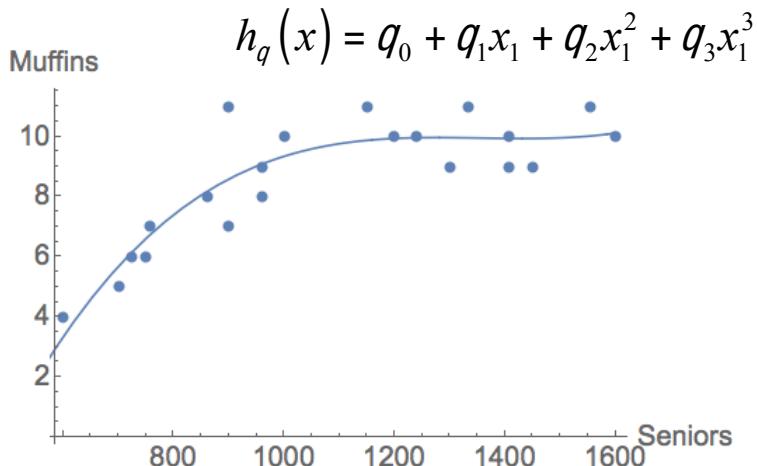
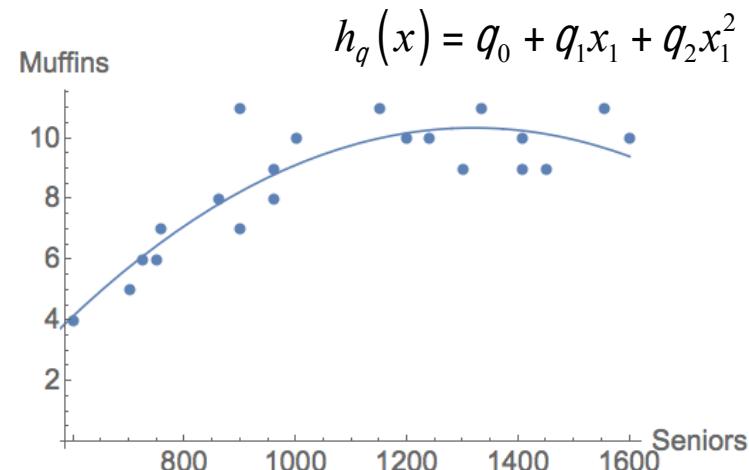
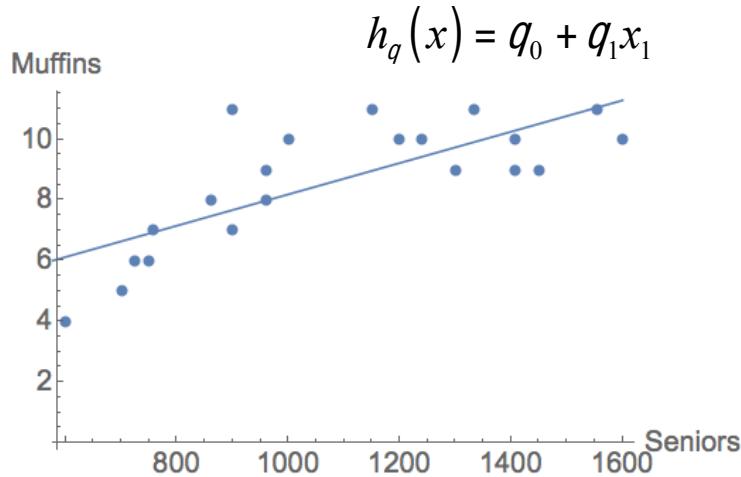
MACHINE LEARNING PROBLEMS



LINEAR REGRESSION

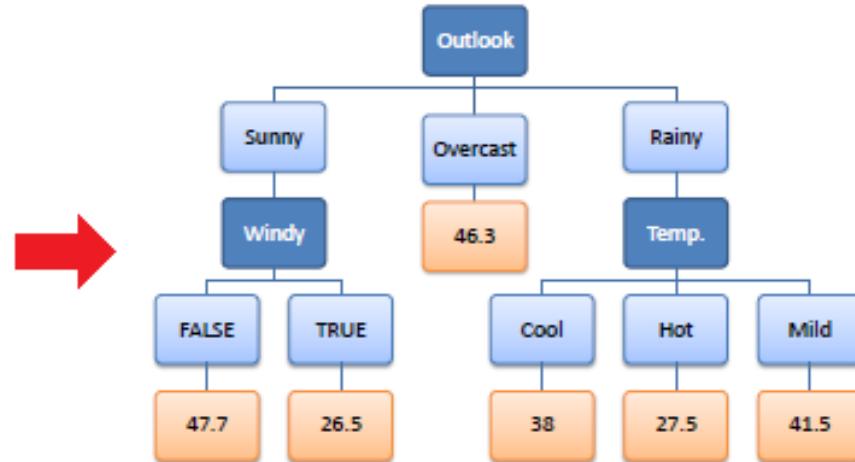


POLYNOMIAL REGRESSION



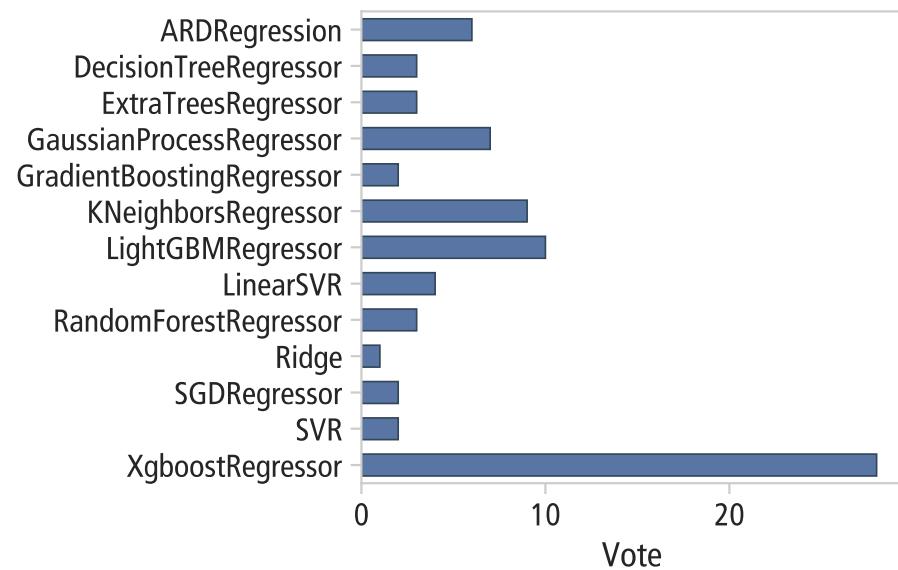
DECISION TREE - REGRESSION

Predictors				Target
Outlook	Temp.	Humidity	Windy	Hours Played
Rainy	Hot	High	False	26
Rainy	Hot	High	True	30
Overcast	Hot	High	False	48
Sunny	Mild	High	False	46
Sunny	Cool	Normal	False	62
Sunny	Cool	Normal	True	23
Overcast	Cool	Normal	True	43
Rainy	Mild	High	False	36
Rainy	Cool	Normal	False	38
Sunny	Mild	Normal	False	48
Rainy	Mild	Normal	True	48
Overcast	Mild	High	True	52
Overcast	Hot	Normal	False	44
Sunny	Mild	High	True	30



PERFORMANCE

How often ranked 1st



Relative Training Time

