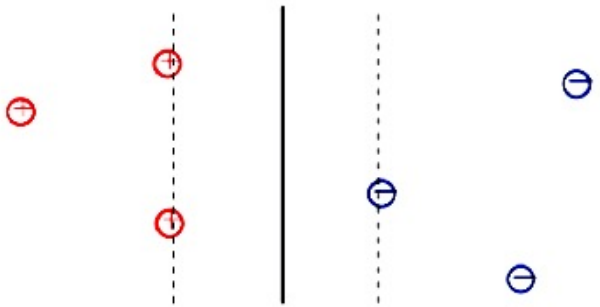
EE 559: Mathematical Pattern Recognition

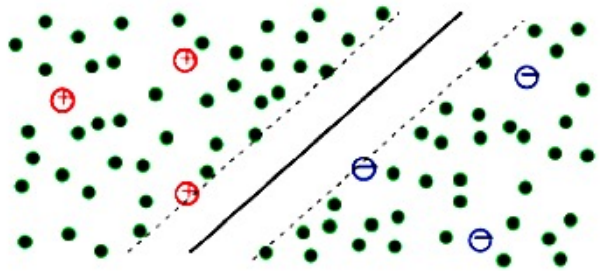
University of Southern California

M. R. Rajati, PhD

Lesson 9
Semi-Supervised Learning



(a) SVM decision boundary



(b) S3VM decision boundary

Introduction to Semi-Supervised Learning (SSL)

- Classifier based methods
  - Co-Training, Yarowsky, and Their Combination
- Data based methods (Not discussed here)
  - Manifold Regularization
  - Harmonic Mixtures
  - Information Regularization

Learning Problems Revisited

Supervised learning:

Given data consisting of feature-label pairs (x_i, y_i), find the predictive relationship between features and labels.

Unsupervised learning:

Given a sample consisting of only objects, look for interesting structures in the data, and group similar objects.

What is Semi-supervised learning?

Supervised learning + Additional unlabeled data

Unsupervised learning + Additional labeled data

Motivation for SSL

Pragmatic:

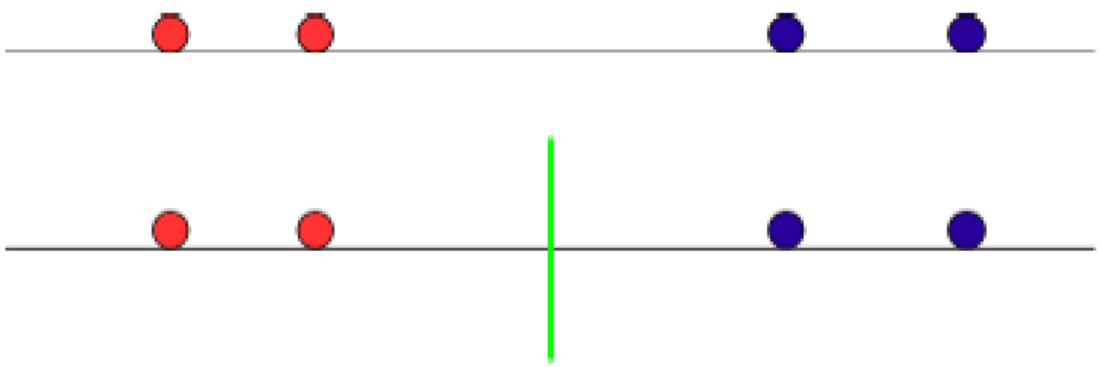
- Unlabeled data is cheap to collect.
- Example: Classifying web pages,
  - There are some annotated web pages.
  - A huge amount of un-annotated pages is easily available by crawling the web.

Methodological:

- The brain can exploit unlabeled data and learn from similarities.

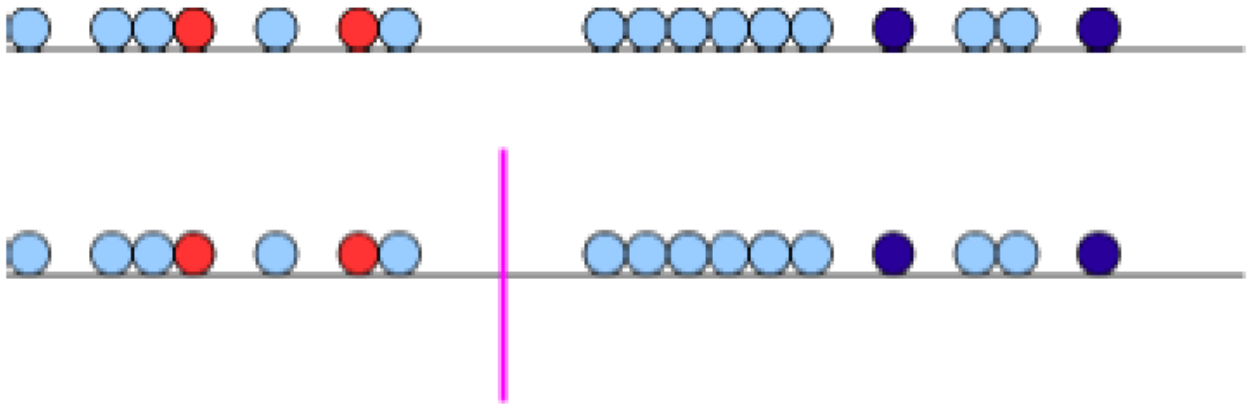
How can unlabeled data help ?

Red: + 1, Dark Blue: -1



How can unlabeled data help ?

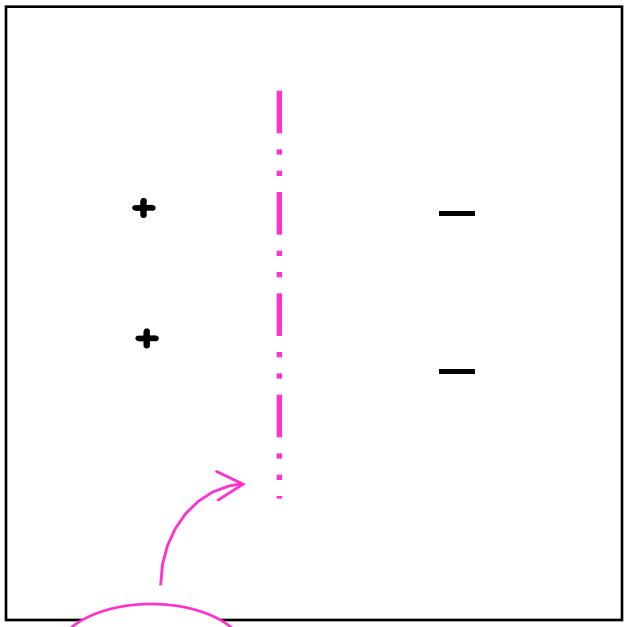
Let's include some additional unlabeled data (Light Blue points)



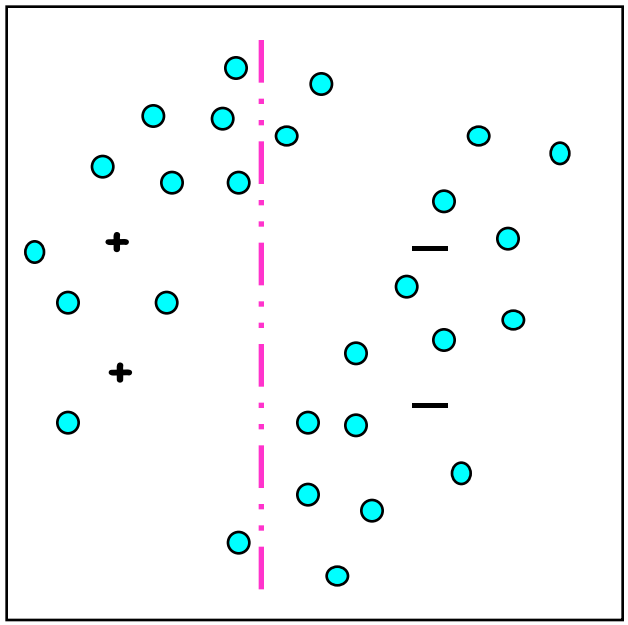
How can unlabeled data help ?

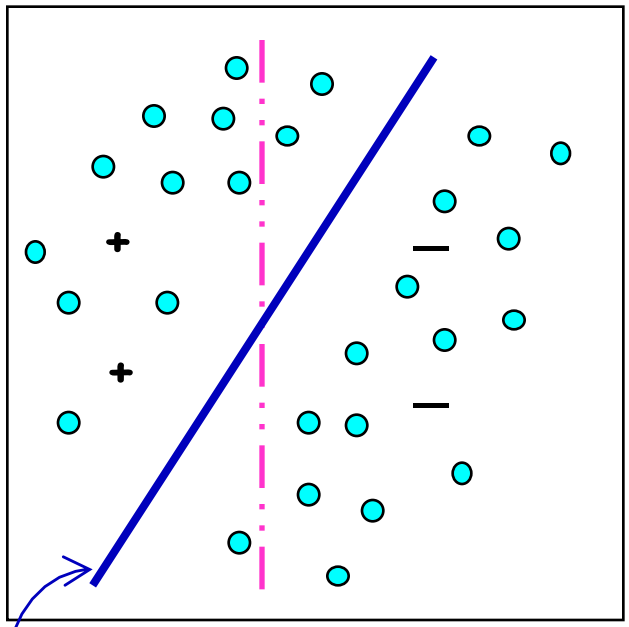
Assumption: Examples from the same class follow a coherent distribution
Unlabeled data can give a better sense of the class separation boundary

Intuition



Labeled data only





Transductive SVM

Inductive vs.Transductive

- Transductive: Produce label only for the available unlabeled data.
  - The output of the method is not a classifier.
- Inductive: Not only produce label for unlabeled data, but also produce a classifier.
- Let's first focus on inductive semi-supervised learning..

Two Algorithmic Approaches

- Classifier based methods (Self-Training):
  - Start from initial classifier(s), and iteratively enhance it (them)
- Data based methods:
  - Discover an inherent geometry in the data, and exploit it in finding a good classifier.

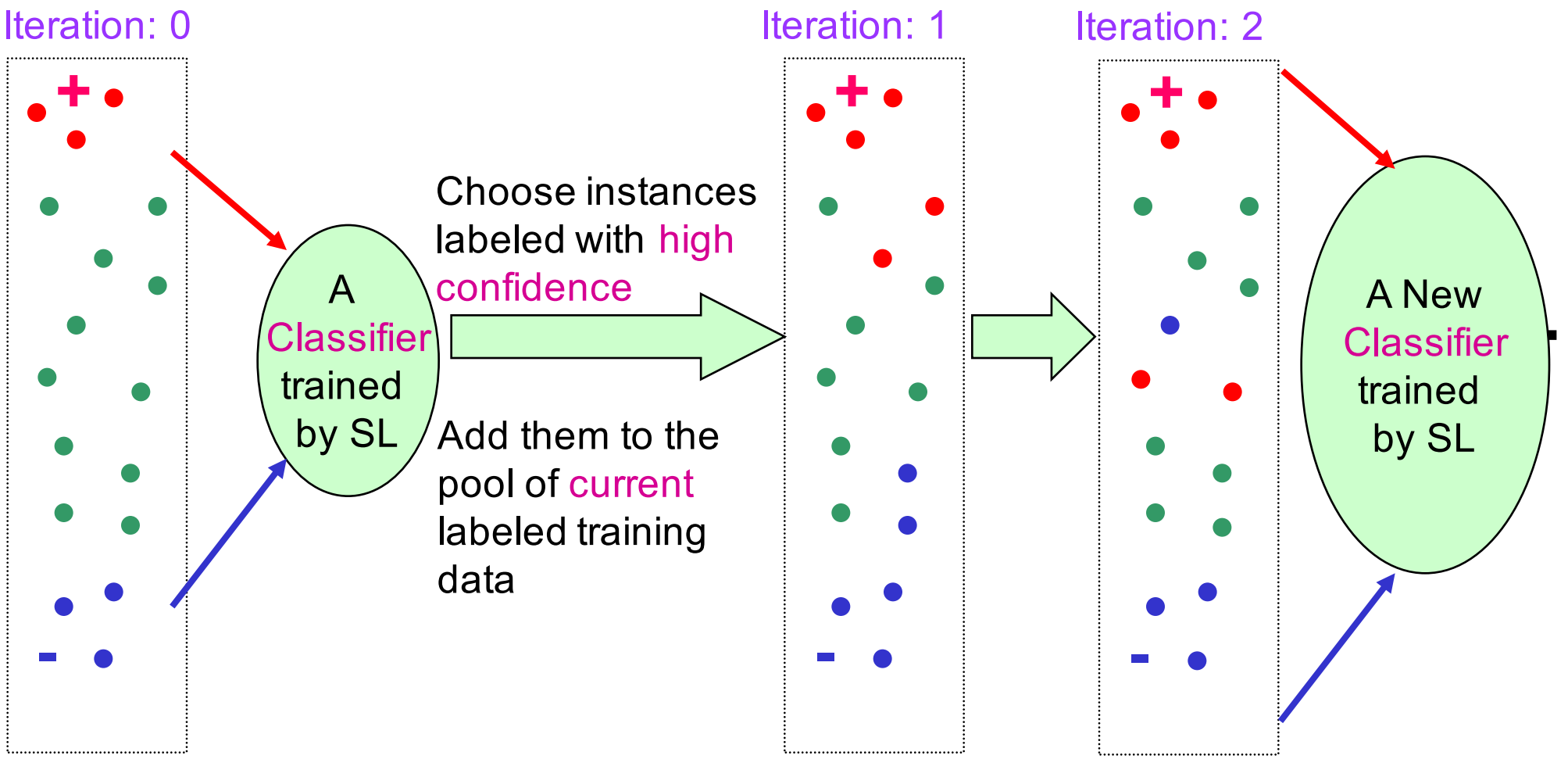
Self-Training (Bootstrap)

Self-Training/ The Yarowsky Algorithm

- Train supervised model on labeled data *L*
- Test on unlabeled data *U*
- Add the most confidently classified members of *U* to *L*
- Repeat

Classifier Based Methods: The Yarowsky Algorithm

(Yarowsky 1995)



Classifier-Based Methods: Refinement

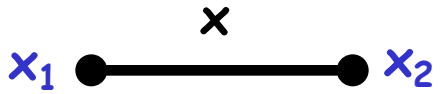
- Refinement:
  - Reduce weight of unlabeled data to increase power of more accurate labeled data

Advantages and Disadvantages of Self-Training

- **Advantages:**
  - The simplest semi-supervised learning method.
  - A wrapper method, applies to existing (complex) classifiers.
  - Often used in real tasks like natural language processing.
- **Disadvantages**
  - Early mistakes could reinforce themselves. Heuristic solutions, e.g. “un-label” an instance if its confidence falls below a threshold.

Co-Training

- Instances contain two sufficient sets of features
  - i.e. an instance is $x=(x_1,x_2)$
  - Each set of features is called a View



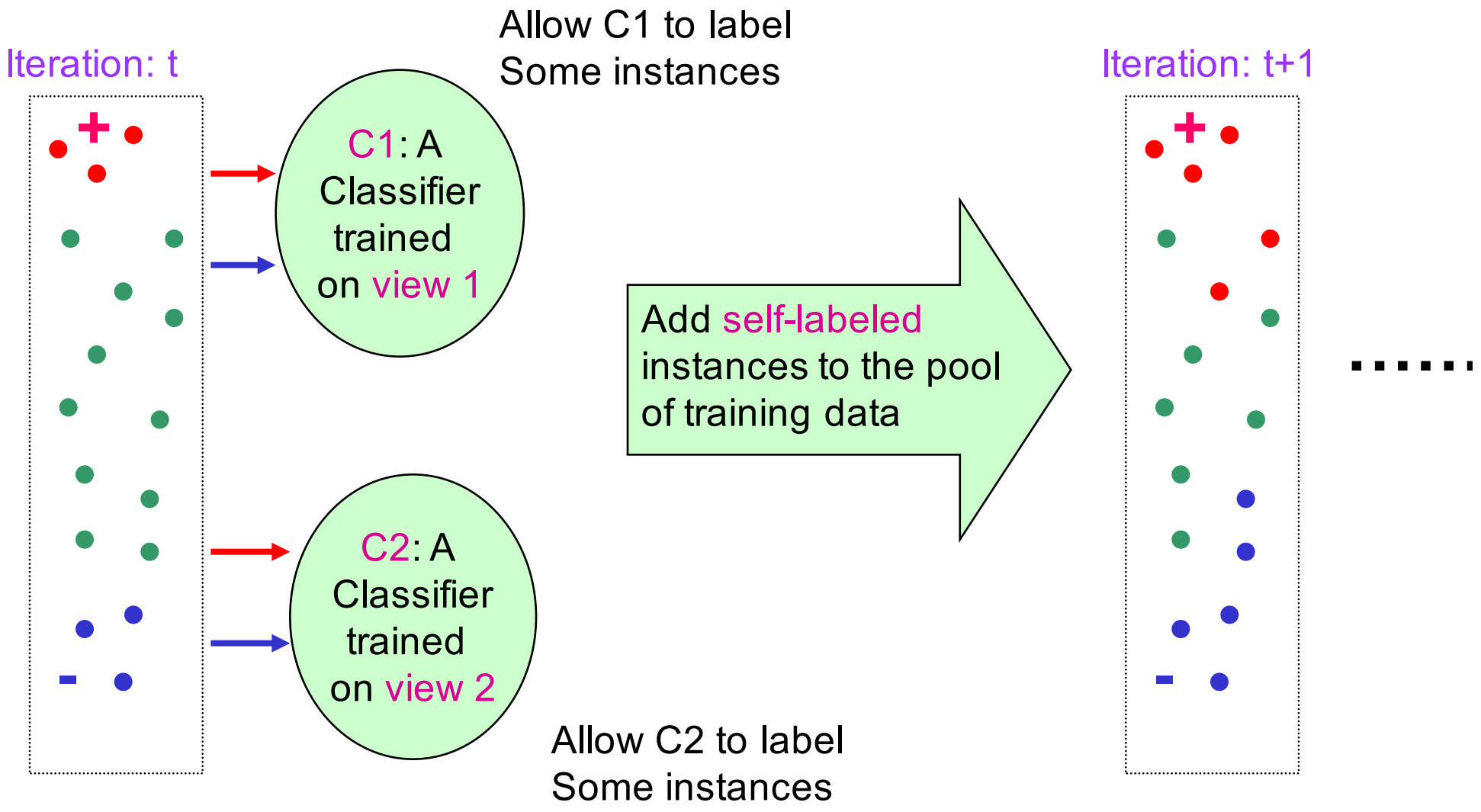
- Two views are independent given the label:

$$P(x_1|x_2,y) = P(x_1|y)$$
$$P(x_2|x_1,y) = P(x_2|y)$$

- Two views are consistent:

$$\exists c_1, c_2 : c^{opt}(x) = c_1(x_1) = c_2(x_2)$$

Co-Training


Iteration: t
Allow C1 to label
Some instances
C1: A
Classifier
trained
on view 1
Add self-labeled
instances to the pool
of training data
Iteration: t+1
C2: A
Classifier
trained
on view 2
Allow C2 to label
Some instances

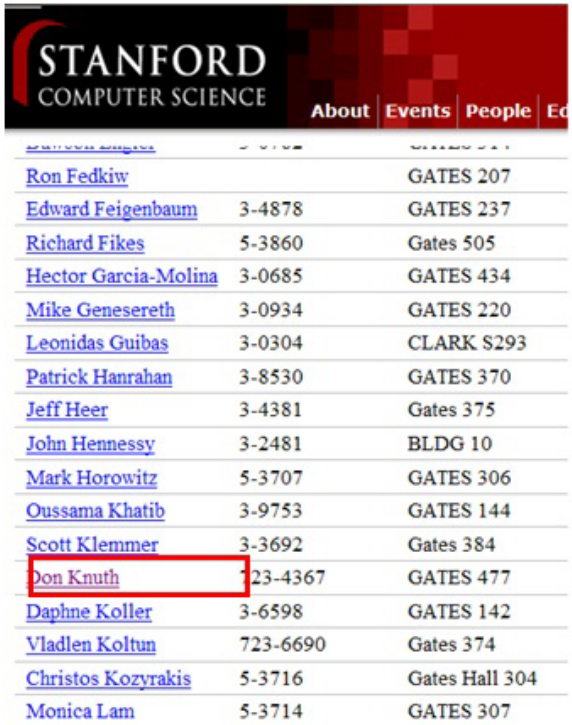
Co-Training

- Example of learning from *multiple views* (multiple sets of attributes): classifying webpages
  - First set of attributes describes content of web page
  - Second set of attributes describes links from other pages
- Independence Assumption:
  - Reduces the probability of the models agreeing on incorrect labels

Yarowsky+ Co-training

- Like Yarowsky Algorithm for semi-supervised learning, but view is switched in each iteration
  - Uses all the unlabeled data (probabilistically labeled) for training
- Has also been used successfully with neural networks and support vector machines

Example



Ron Fedkiw		GATES 207
Edward Feigenbaum	3-4878	GATES 237
Richard Fikes	5-3860	Gates 505
Hector Garcia-Molina	3-0685	GATES 434
Mike Genesereth	3-0934	GATES 220
Leonidas Guibas	3-0304	CLARK S293
Patrick Hanrahan	3-8530	GATES 370
Jeff Heer	3-4381	Gates 375
John Hennessy	3-2481	BLDG 10
Mark Horowitz	5-3707	GATES 306
Oussama Khatib	3-9753	GATES 144
Scott Klemmer	3-3692	Gates 384
Don Knuth	723-4367	GATES 477
Daphne Koller	3-6598	GATES 142
Vladlen Koltun	723-6690	Gates 374
Christos Kozyrakis	5-3716	Gates Hall 304
Monica Lam	5-3714	GATES 307

Classify web pages into category for students and category for professors

Two views of web page

Content: I am currently a professor of ...

Hyperlinks: a link to the faculty list of computer science department

General Multiview Learning

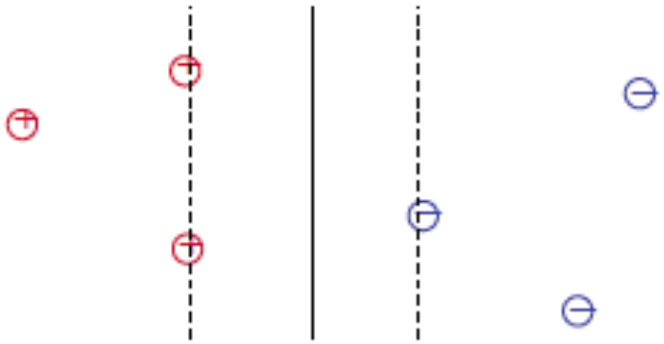
Train multiple diverse models on *L*.
Those instances in *U* which most models agree on are placed in *L*.

Semi-Supervised SVM

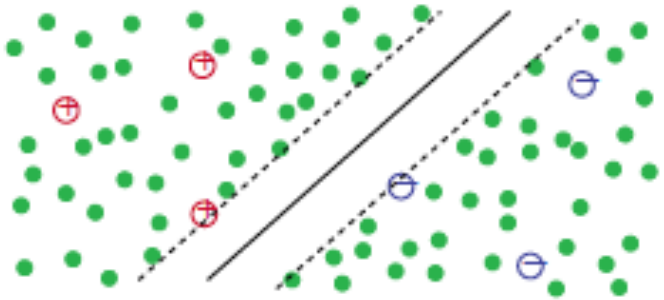
Semi-Supervised SVM (S3VM)

Maximize margin of both *L* and *U*. Decision surface placed in non-dense spaces

- Assumes classes are "well-separated"
- Can also try to simultaneously maintain class proportion on both sides similar to labeled proportion



(a) SVM decision boundary



(b) S3VM decision boundary

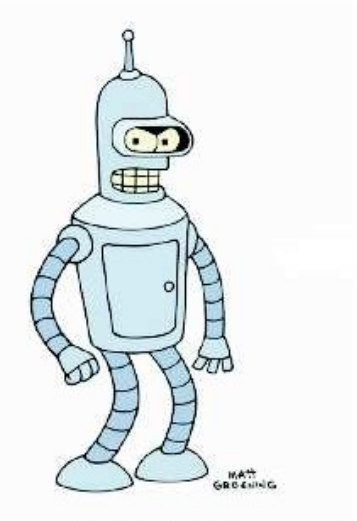
Cluster-and-Label Approach

- Assumption: Clusters coincide with decision boundaries
- Poor results if this assumption is wrong
  - Cluster labeled and unlabeled data
  - For each cluster, train a classifier based on the labeled points within that cluster
  - Label all data in each cluster using the classifier designed for that cluster.
  - Train a model based on the whole data (that is now labeled)

(Passive) Supervised Learning

raw unlabeled data

$x_1, x_2, x_3, \ldots$



supervised learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

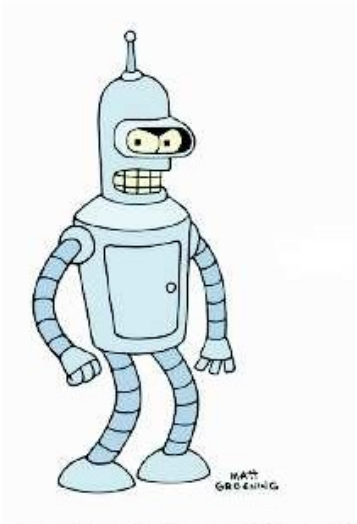
Some figures from Burr Settles

(Passive) Supervised Learning

raw unlabeled data

$x_1, x_2, x_3, \ldots$

random sample



supervised learner
induces a classifier

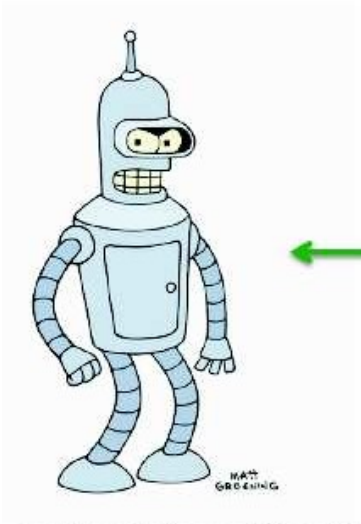
expert / oracle
analyzes experiments
to determine labels

(Passive) Supervised Learning

raw unlabeled data

$x_1, x_2, x_3, \ldots$

random sample



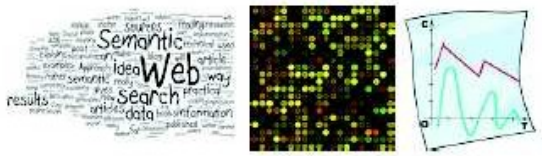
labeled training instances

$\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \langle x_3, y_3 \rangle, \ldots$

supervised learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

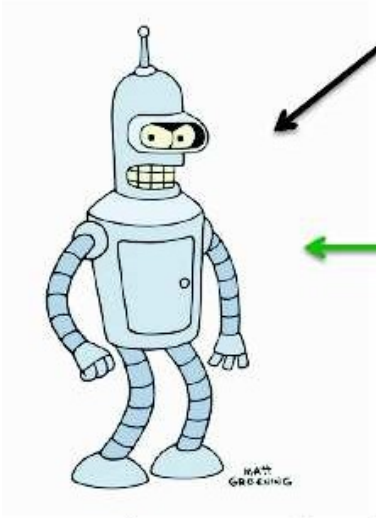
Semi-supervised Learning



exploit the structure
in unlabeled data

raw unlabeled data
$x_1, x_2, x_3, \ldots$

random sample



labeled training instances

$\langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle, \langle x_3, y_3 \rangle, \ldots$

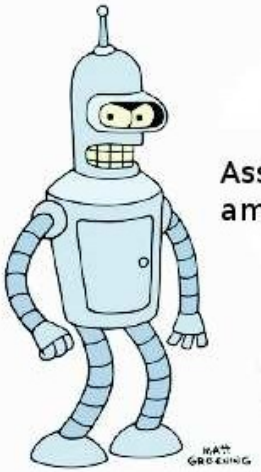
semi-supervised learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

Active Learning

raw unlabeled data

$x_1, x_2, x_3, \ldots$

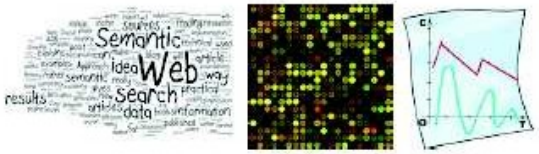


Assumes some small
amount of initial labeled training data

active learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

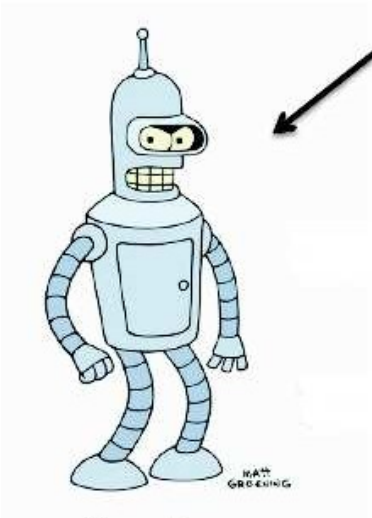
Active Learning



raw unlabeled data

$x_1, x_2, x_3, \ldots$

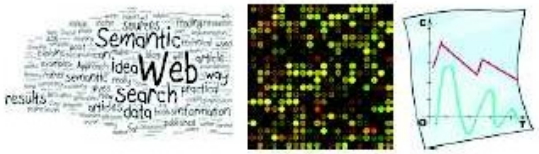
inspect the unlabeled data



active learner
induces a classifier

expert / oracle
analyzes experiments to determine labels

Active Learning



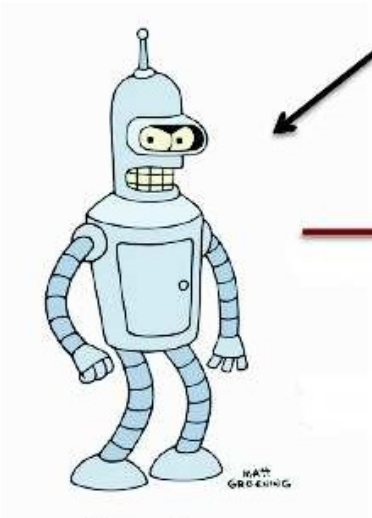
raw unlabeled data

$x_1, x_2, x_3, \ldots$

inspect the unlabeled data

request labels for selected data

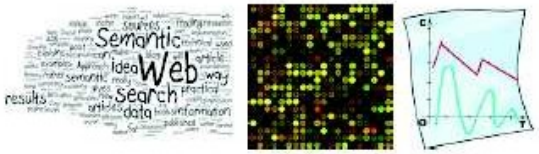
$\langle x_1, ? \rangle$



active learner
induces a classifier

expert / oracle
analyzes experiments to determine labels

Active Learning



raw unlabeled data

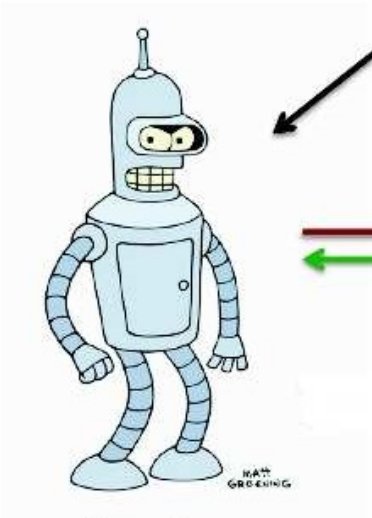
$x_1, x_2, x_3, \ldots$

inspect the unlabeled data

request labels for selected data

$\langle x_1, ? \rangle$

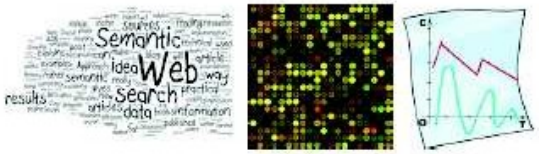
$\langle x_1, y_1 \rangle$



active learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

Active Learning



inspect the unlabeled data

raw unlabeled data

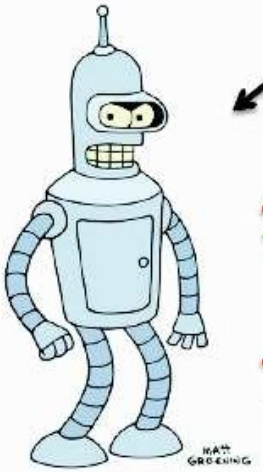
$x_1, x_2, x_3, \ldots$

request labels for selected data

$\langle x_1, ? \rangle$

$\langle x_1, y_1 \rangle$

$\langle x_2, ? \rangle$



active learner
induces a classifier

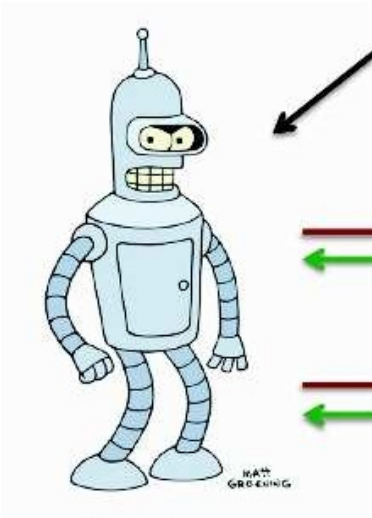
expert / oracle
analyzes experiments
to determine labels

Active Learning

inspect the unlabeled data

raw unlabeled data

$x_1, x_2, x_3, \ldots$



request labels for selected data

$\langle x_1, ? \rangle$

$\langle x_1, y_1 \rangle$

$\langle x_2, ? \rangle$

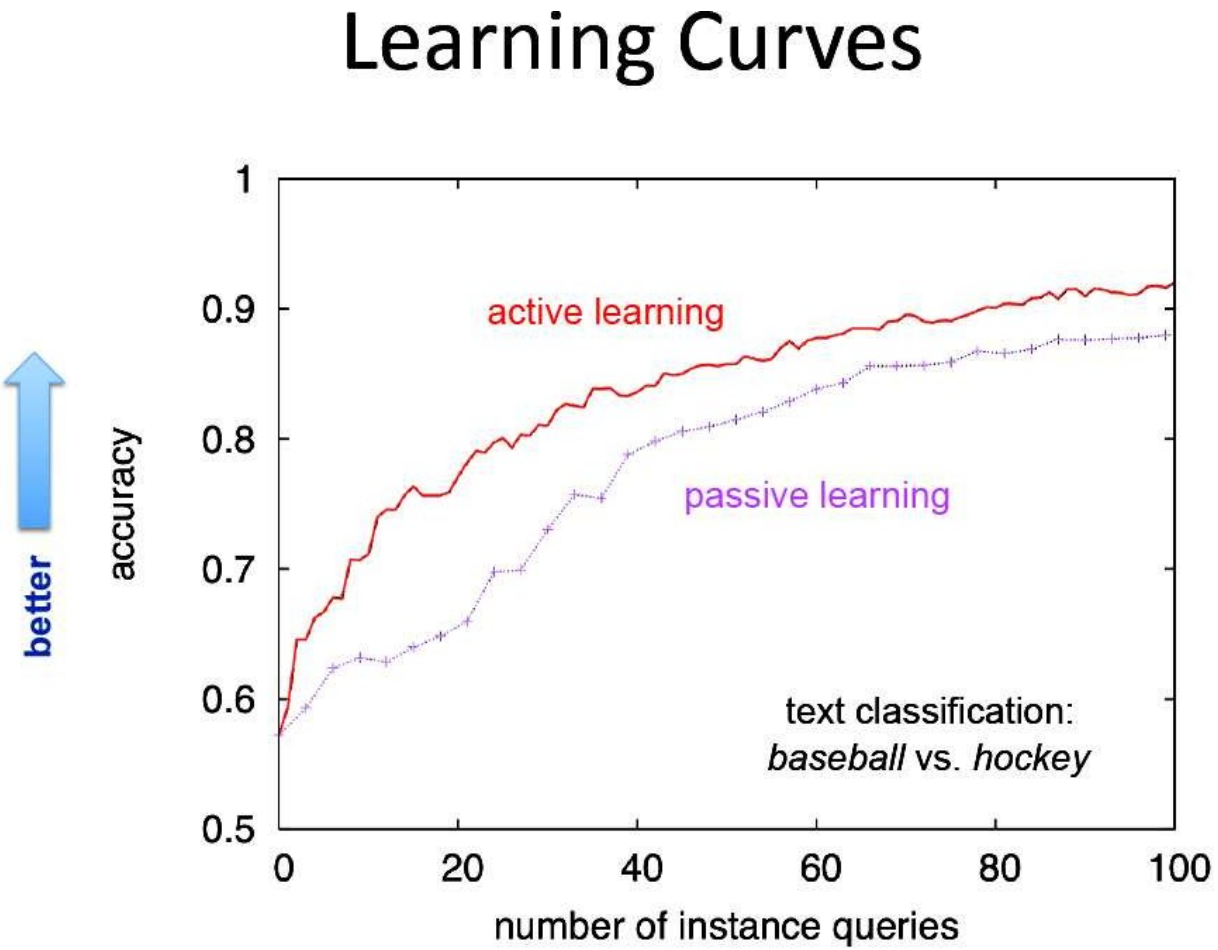
$\langle x_2, y_2 \rangle$

active learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

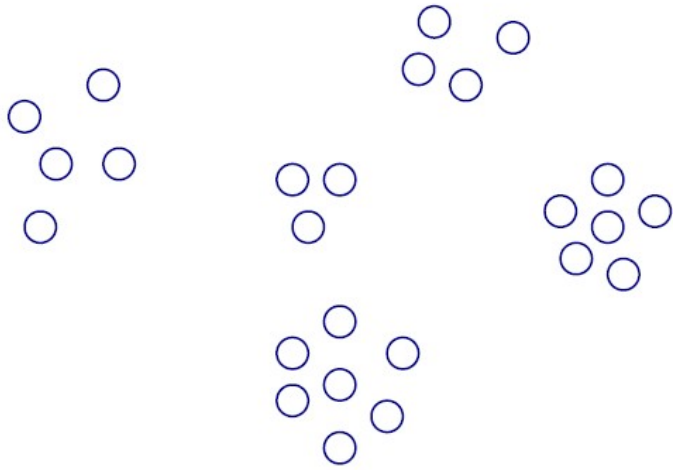
Active Learning vs Random Sampling

Passive Learning curve: Randomly selects examples to get labels for Active Learning curve: Active learning selects examples to get labels for



A Naive Approach

Suppose the unlabeled data looks like this.



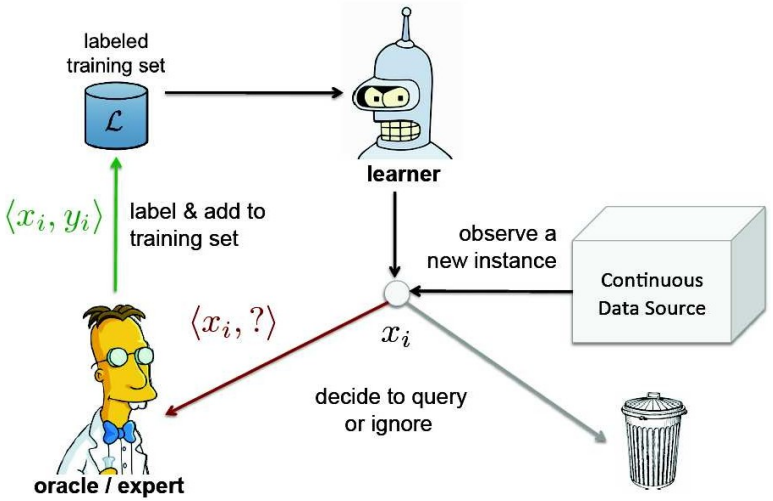
Then perhaps we just need five labels!

- Of course, things could go wrong..

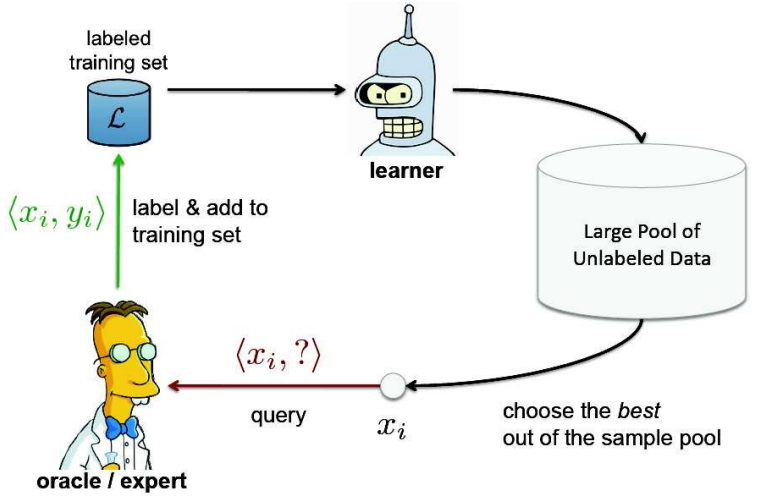
Types of Active Learning

Largely falls into one of these two types:

- Stream-Based Active Learning
  - Consider one unlabeled example at a time
  - Decide whether to query its label or ignore it



- Pool-Based Active Learning Given:
  - a large unlabeled pool of examples
  - Rank examples in order of informativeness
  - Query the labels for the most informative example(s)



Query Selection Strategies

Any Active Learning algorithm requires a query selection strategy

Some examples:

- Uncertainty Sampling
- Query By Committee (QBC)
- Expected Model Change
- Expected Error Reduction
- Variance Reduction
- Density Weighted Methods

How Active Learning Operates

- Active Learning proceeds in rounds
- Each round has a current model (learned using the labeled data seen so far)
- The current model is used to assess informativeness of unlabeled examples
  - ... using one of the query selection strategies
- The most informative example(s) is/ are selected

How Active Learning Operates

- The labels are obtained (by the labeling oracle)
- The (now) labeled example(s)

  is/are included in the training data

  The model is re-trained using the

  new training data
- The process repeat until we have budget left for getting labels

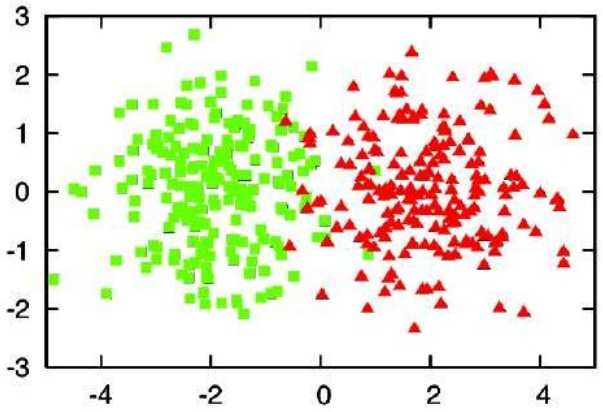
Uncertainty Sampling

Select examples which the current model is the most uncertain about Various ways to measure uncertainty. For example:

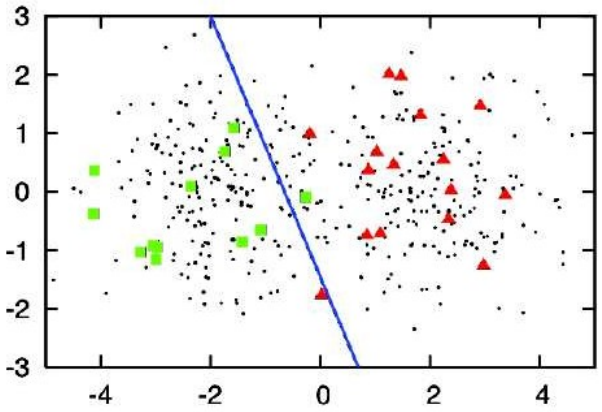
- Based on the distance from the hyperplane
- Using the label probability $P(y \mid x)$ (for probabilistic models)

Uncertainty Sampling

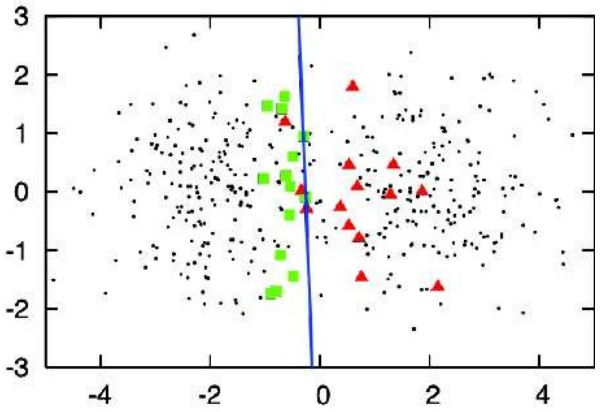
A simple illustration of uncertainty sampling based on the distance from the hyperplane (i.e., margin based)



400 instances sampled
from 2 class Gaussians



random sampling
30 labeled instances
(accuracy=0.7)



uncertainty sampling
30 labeled instances
(accuracy=0.9)

Query By Committee (QBC)

- QBC uses a committee of models
- All models trained using the currently available labeled data *L*
- How is the committee constructed?
  - Some possible ways: Sampling different models from the model distribution
  - Using ensemble methods (bagging/boosting, etc.)

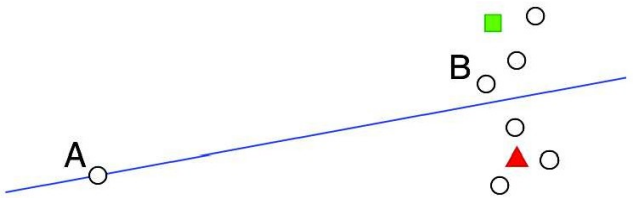
Query By Committee (QBC)

- All models vote their predictions on the unlabeled pool
- The example(s) with maximum disagreement is/are chosen for labeling
- Each model in the committee is re-trained after including the new example(s)

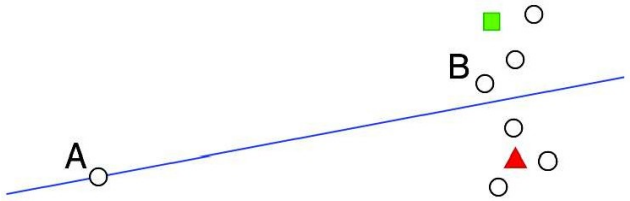
Effect of Outlier Examples

Uncertainty Sampling or QBC may wrongly think an outlier to be an informative example

Such examples won't really help (and can even be misleading)



Effect of Outlier Examples



Other robust query selection methods exist to deal with outliers

Idea: Instead of using the confidence of a model on an example, see how a labeled example affects the model itself (various ways to quantify this)

The example(s) that affects the model the most is probably the most informative

Other Query Selection Methods

Expected Model Change

Select the example whose inclusion brings about the maximum change in the model (e.g., the gradient of the loss function w.r.t. the parameters)

Expected Error Reduction

Select example that reduces the expected generalization error the most

.. measured w.r.t. the remaining unlabeled examples (using the *expected* labels)

Other Query Selection Methods

Variance Reduction

Select example(s) that reduces the model variance by the most

.. by maximizing Fisher information of model parameters (e.g., by *minimizing* the trace or determinant of the *inverse* Fisher information matrix)

Fisher information matrix: computed using the log-likelihood

Density Weighting

Weight the informativeness of an example by its average similarity to the entire unlabeled pool of examples

An outlier will not get a substantial weight!

Active Learning with Selective Sampling

Looking at one example at a time witha margin-based classifier

Input: Parameter $b > 0$ (dictates how aggressively we want to query labels)

For $n = 1 : N$

Get x_n, compute p_n

Predict $\hat{y}_n = \text{sign}(p_n)$

Draw Bernoulli random variable $Z \in \{0, 1\}$ with probability $b/b+|p|$

If $Z == 1$, query the true label y_n

Else if $Z == 0$, ignore the example x_n and don't update w

Comments:

$|p_n|$ is the absolute margin of x_n

Large $|p_n| \Rightarrow$ Small label query probability

Interesting Observations and Questions

- Active Learning: Label efficient learning strategy
  Based on judging the informativeness of examples
- Several variants possible. E.g.,
  - Different examples having different labeling costs
  - Access to multiple labeling oracles (possibly noisy)
  - Active Learning on features instead of labels (e.g., if features are expensive)
- Being “actively” used in industry (IBM, Microsoft, Siemens, Google, etc.)

Interesting Observations and Questions

Can an actively labeled dataset be reused to train a new different model?

Sampling is biased. The actively labeled dataset doesn't reflect the true training/test data distribution. What could be the consequences? How could this be accounted for?