A tutorial on active learning

Sanjoy Dasgupta[1] John Langford[2]

UC San Diego[1]

Yahoo Labs[2]

Exploiting unlabeled data

A lot of unlabeled data is plentiful and cheap, eg.

- documents off the web
- speech samples
- images and video

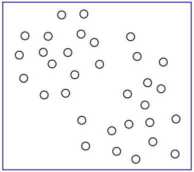
But labeling can be expensive.

Exploiting unlabeled data

A lot of unlabeled data is plentiful and cheap, eg.

- documents off the web
- speech samples
- images and video

But labeling can be expensive.



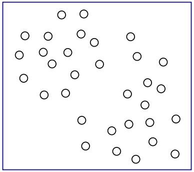
Unlabeled points

Exploiting unlabeled data

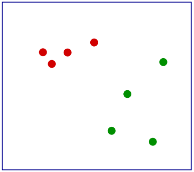
A lot of unlabeled data is plentiful and cheap, eg.

- documents off the web
- speech samples
- images and video

But labeling can be expensive.



Unlabeled points



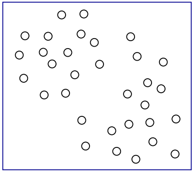
Supervised learning

Exploiting unlabeled data

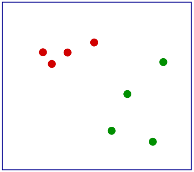
A lot of unlabeled data is plentiful and cheap, eg.

- documents off the web
- speech samples
- images and video

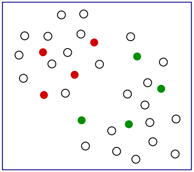
But labeling can be expensive.



Unlabeled points



Supervised learning



Semisupervised and active learning

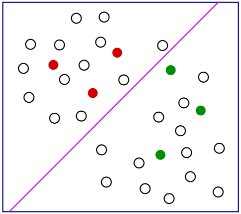
Typical heuristics for active learning

Start with a pool of unlabeled data

Pick a few points at random and get their labels

Repeat

- Fit a classifier to the labels seen so far
- Query the unlabeled point that is closest to the boundary (or most uncertain, or most likely to decrease overall uncertainty,...)



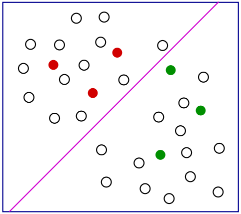
Typical heuristics for active learning

Start with a pool of unlabeled data

Pick a few points at random and get their labels

Repeat

Fit a classifier to the labels seen so far
Query the unlabeled point that is closest to the boundary
(or most uncertain, or most likely to decrease overall
uncertainty,...)



Biased sampling: the labeled points are not representative of the underlying distribution!

Sampling bias

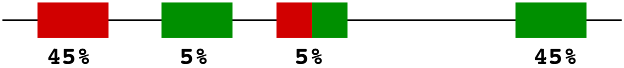
Start with a pool of unlabeled data

Pick a few points at random and get their labels

Repeat

Fit a classifier to the labels seen so far
Query the unlabeled point that is closest to the boundary
(or most uncertain, or most likely to decrease overall uncertainty,...)

Example:



Sampling bias

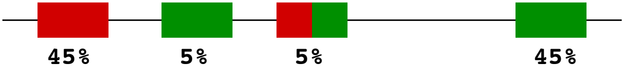
Start with a pool of unlabeled data

Pick a few points at random and get their labels

Repeat

Fit a classifier to the labels seen so far
Query the unlabeled point that is closest to the boundary
(or most uncertain, or most likely to decrease overall uncertainty,...)

Example:



Even with infinitely many labels, converges to a classifier with 5% error instead of the best achievable, 2.5%. *Not consistent!*

Sampling bias

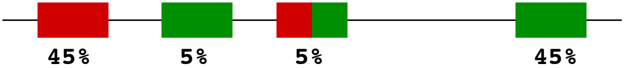
Start with a pool of unlabeled data

Pick a few points at random and get their labels

Repeat

Fit a classifier to the labels seen so far
Query the unlabeled point that is closest to the boundary
(or most uncertain, or most likely to decrease overall
uncertainty,...)

Example:

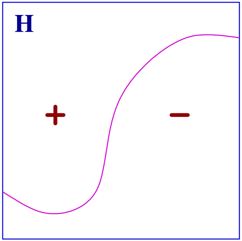


Even with infinitely many labels, converges to a classifier with 5% error instead of the best achievable, 2.5%. *Not consistent!*

Manifestation in practice, eg. Schutze et al 03.

Case II: Efficient search through hypothesis space

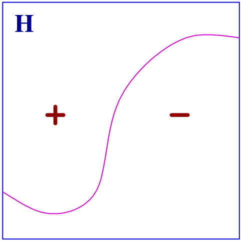
Ideal case: each query cuts the version space in two.



Then perhaps we need just $\log |H|$ labels to get a perfect hypothesis!

Case II: Efficient search through hypothesis space

Ideal case: each query cuts the version space in two.



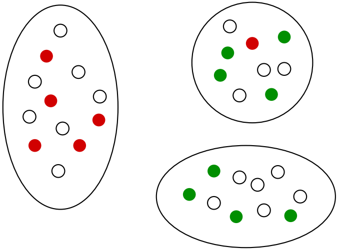
Then perhaps we need just $\log |H|$ labels to get a perfect hypothesis!

Challenges: (1) Do there always exist queries that will cut off a good portion of the version space? (2) If so, how can these queries be found? (3) What happens in the nonseparable case?

Exploiting cluster structure in data [DH 08]

Basic primitive:

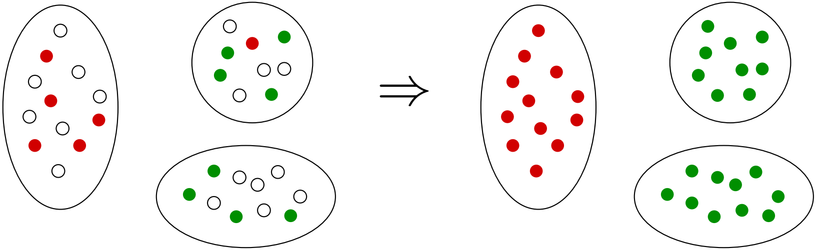
- Find a clustering of the data
- Sample a few *randomly-chosen* points in each cluster
- Assign each cluster its majority label
- Now use this fully labeled data set to build a classifier



Exploiting cluster structure in data [DH 08]

Basic primitive:

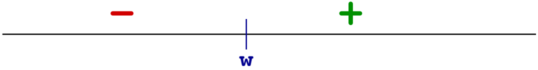
- Find a clustering of the data
- Sample a few *randomly-chosen* points in each cluster
- Assign each cluster its majority label
- Now use this fully labeled data set to build a classifier



Efficient search through hypothesis space

Threshold functions on the real line:

$H = \{h_w : w \in \mathbb{R}\}$
$h_w(x) = 1(x \geq w)$



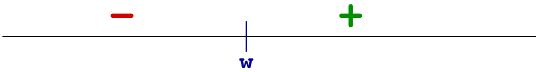
Supervised: for misclassification error $\leq \epsilon$, need $\approx 1/\epsilon$ labeled points.

Efficient search through hypothesis space

Threshold functions on the real line:

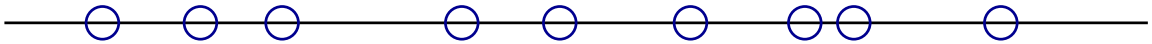
$H = \{h_w : w \in \mathbb{R}\}$

$h_w(x) = 1(x \geq w)$



Supervised: for misclassification error $\leq \epsilon$, need $\approx 1/\epsilon$ labeled points.

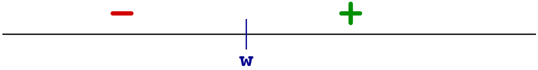
Active learning: instead, start with $1/\epsilon$ *unlabeled* points.



Efficient search through hypothesis space

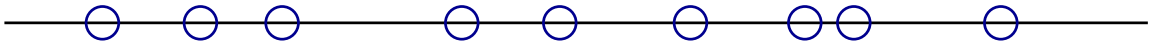
Threshold functions on the real line:

$H = \{h_w : w \in \mathbb{R}\}$
$h_w(x) = 1(x \geq w)$



Supervised: for misclassification error $\leq \epsilon$, need $\approx 1/\epsilon$ labeled points.

Active learning: instead, start with $1/\epsilon$ *unlabeled* points.

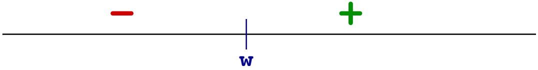


Binary search: need just $\log 1/\epsilon$ labels, from which the rest can be inferred. *Exponential improvement in label complexity!*

Efficient search through hypothesis space

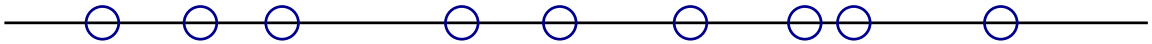
Threshold functions on the real line:

$H = \{h_w : w \in \mathbb{R}\}$
$h_w(x) = 1(x \geq w)$



Supervised: for misclassification error $\leq \epsilon$, need $\approx 1/\epsilon$ labeled points.

Active learning: instead, start with $1/\epsilon$ *unlabeled* points.



Binary search: need just $\log 1/\epsilon$ labels, from which the rest can be inferred. *Exponential improvement in label complexity!*

Challenges: Nonseparable data? Other hypothesis classes?

Some results of active learning theory

	Separable data	General (nonseparable) data
Aggressive	Query by committee (Freund, Seung, Shamir, Tishby, 97) Splitting index (D, 05)	
Mellow	Generic active learner (Cohn, Atlas, Ladner, 91)	A^2 algorithm (Balcan, Beygelzimer, L, 06) Disagreement coefficient (Hanneke, 07) Reduction to supervised (D, Hsu, Monteleoni, 2007) Importance-weighted approach (Beygelzimer, D, L, 2009)

Some results of active learning theory

	Separable data	General (nonseparable) data
Aggressive	Query by committee (Freund, Seung, Shamir, Tishby, 97) Splitting index (D, 05)	
Mellow	Generic active learner (Cohn, Atlas, Ladner, 91)	A^2 algorithm (Balcan, Beygelzimer, L, 06) Disagreement coefficient (Hanneke, 07) Reduction to supervised (D, Hsu, Monteleoni, 2007) Importance-weighted approach (Beygelzimer, D, L, 2009)

Issues:

Computational tractability

Are labels being used as efficiently as possible?

A generic mellow learner [CAL '91]

For *separable* data that is streaming in.

H_1 = hypothesis class

Repeat for $t = 1, 2, \ldots$

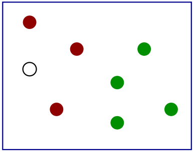
Receive unlabeled point x_t

If there is any disagreement within H_t about x_t's label:

query label y_t and set $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$

else

$H_{t+1} = H_t$



Is a label needed?

A generic mellow learner [CAL '91]

For *separable* data that is streaming in.

H_1 = hypothesis class

Repeat for $t = 1, 2, \ldots$

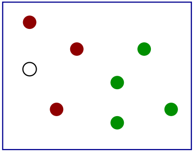
Receive unlabeled point x_t

If there is any disagreement within H_t about x_t's label:

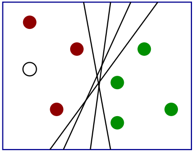
query label y_t and set $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$

else

$H_{t+1} = H_t$



Is a label needed?



H_t = current candidate hypotheses

A generic mellow learner [CAL '91]

For *separable* data that is streaming in.

H_1 = hypothesis class

Repeat for $t = 1, 2, \ldots$

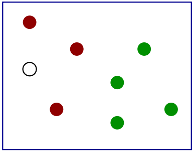
Receive unlabeled point x_t

If there is any disagreement within H_t about x_t's label:

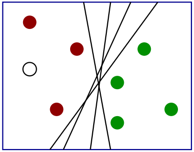
query label y_t and set $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$

else

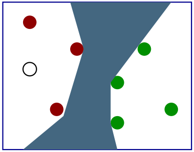
$H_{t+1} = H_t$



Is a label needed?



H_t = current candidate hypotheses



Region of uncertainty

A generic mellow learner [CAL '91]

For *separable* data that is streaming in.

H_1 = hypothesis class

Repeat for $t = 1, 2, \ldots$

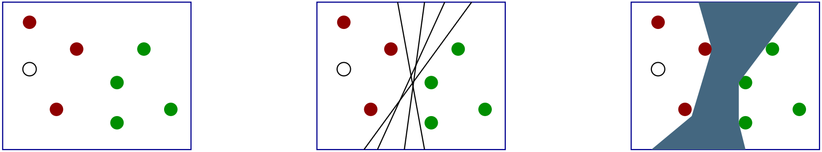
Receive unlabeled point x_t

If there is any disagreement within H_t about x_t's label:

query label y_t and set $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$

else

$H_{t+1} = H_t$



Is a label needed? | H_t = current candidate hypotheses | Region of uncertainty

Problems: (1) intractable to maintain H_t; (2) nonseparable data.

Maintaining H_t

Explicitly maintaining H_t is intractable. Do it implicitly, by reduction to supervised learning.

Explicit version

H_1 = hypothesis class
For $t = 1, 2, \ldots$:
  Receive unlabeled point x_t
  If disagreement in H_t about x_t's label:
    query label y_t of x_t
    $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$
  else:
    $H_{t+1} = H_t$

Implicit version

$S = \{\}$ (points seen so far)
For $t = 1, 2, \ldots$:
  Receive unlabeled point x_t
  If learn($S \cup (x_t, 1)$) and learn($S \cup (x_t, 0)$) both return an answer:
    query label y_t
  else:
    set y_t to whichever label succeeded
  $S = S \cup \{(x_t, y_t)\}$

Maintaining H_t

Explicitly maintaining H_t is intractable. Do it implicitly, by reduction to supervised learning.

Explicit version

H_1 = hypothesis class
For $t = 1, 2, \ldots$:
  Receive unlabeled point x_t
  If disagreement in H_t about x_t's label:
    query label y_t of x_t
    $H_{t+1} = \{h \in H_t : h(x_t) = y_t\}$
  else:
    $H_{t+1} = H_t$

Implicit version

$S = \{\}$ (points seen so far)
For $t = 1, 2, \ldots$:
  Receive unlabeled point x_t
  If learn$(S \cup (x_t, 1))$ and learn$(S \cup (x_t, 0))$
  both return an answer:
    query label y_t
  else:
    set y_t to whichever label succeeded
  $S = S \cup \{(x_t, y_t)\}$

This scheme is no worse than straight supervised learning. But can one bound the number of labels needed?

Label complexity [Hanneke]

The label complexity of CAL (mellow, separable) active learning can be captured by the the VC dimension d of the hypothesis and by a parameter θ called the *disagreement coefficient*.

Label complexity [Hanneke]

The label complexity of CAL (mellow, separable) active learning can be captured by the the VC dimension d of the hypothesis and by a parameter θ called the *disagreement coefficient*.

- Regular supervised learning, separable case.

  Suppose data are sampled iid from an underlying distribution. To get a hypothesis whose misclassification rate (on the underlying distribution) is $\leq \epsilon$ with probability ≥ 0.9, it suffices to have

  $$\frac{d}{\epsilon}$$

  labeled examples.

Label complexity [Hanneke]

The label complexity of CAL (mellow, separable) active learning can be captured by the the VC dimension d of the hypothesis and by a parameter θ called the *disagreement coefficient*.

- Regular supervised learning, separable case.

  Suppose data are sampled iid from an underlying distribution. To get a hypothesis whose misclassification rate (on the underlying distribution) is $\leq \epsilon$ with probability ≥ 0.9, it suffices to have

  $$\frac{d}{\epsilon}$$

  labeled examples.

- CAL active learner, separable case.

  Label complexity is

  $$\theta d \log \frac{1}{\epsilon}$$

Label complexity [Hanneke]

The label complexity of CAL (mellow, separable) active learning can be captured by the the VC dimension d of the hypothesis and by a parameter θ called the *disagreement coefficient*.

- Regular supervised learning, separable case.

  Suppose data are sampled iid from an underlying distribution. To get a hypothesis whose misclassification rate (on the underlying distribution) is $\leq \epsilon$ with probability ≥ 0.9, it suffices to have

  $$\frac{d}{\epsilon}$$

  labeled examples.

- CAL active learner, separable case.

  Label complexity is

  $$\theta d \log \frac{1}{\epsilon}$$

- There is a version of CAL for nonseparable data. (More to come!)

  If best achievable error rate is ν, suffices to have

  $$\theta \left(d \log^2 \frac{1}{\epsilon} + \frac{d\nu^2}{\epsilon^2} \right)$$

  labels. Usual supervised requirement: d/ϵ^2.

Disagreement coefficient [Hanneke]

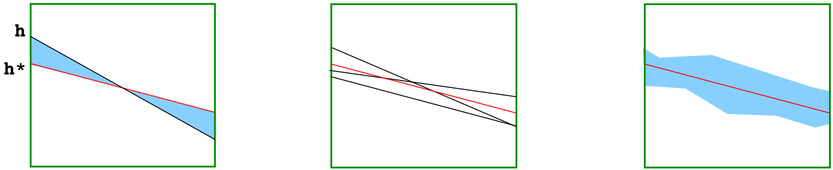
Let $\mathbb{P}$ be the underlying probability distribution on input space $\mathcal{X}$.
Induces (pseudo-)metric on hypotheses: $d(h, h') = \mathbb{P}[h(X) \neq h'(X)]$.
Corresponding notion of *ball* $B(h, r) = \{h' \in H : d(h, h') < r\}$.

Disagreement region of any set of candidate hypotheses $V \subseteq H$:

$$\mathrm{DIS}(V) = \{x : \exists h, h' \in V \text{ such that } h(x) \neq h'(x)\}.$$

Disagreement coefficient for target hypothesis $h^* \in H$:

$$\theta = \sup_r \frac{\mathbb{P}[\mathrm{DIS}(B(h^*, r))]}{r}.$$



$d(h^*, h) = \mathbb{P}[\text{shaded region}]$ | Some elements of $B(h^*, r)$ | $\mathrm{DIS}(B(h^*, r))$

Disagreement coefficient: separable case

Let $\mathbb{P}$ be the underlying probability distribution on input space $\mathcal{X}$.
Let H_ϵ be all hypotheses in H with error $\leq \epsilon$. Disagreement region:

$$\mathrm{DIS}(H_\epsilon) = \{x : \exists h, h' \in H_\epsilon \text{ such that } h(x) \neq h'(x)\}.$$

Then disagreement coefficient is

$$\theta = \sup_\epsilon \frac{\mathbb{P}[\mathrm{DIS}(H_\epsilon)]}{\epsilon}.$$

Disagreement coefficient: separable case

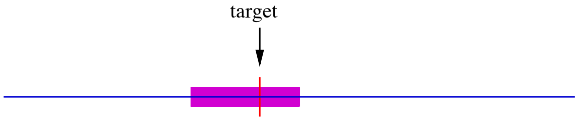
Let $\mathbb{P}$ be the underlying probability distribution on input space $\mathcal{X}$.
Let H_ϵ be all hypotheses in H with error $\leq \epsilon$. Disagreement region:

$$\mathsf{DIS}(H_\epsilon) = \{x : \exists h, h' \in H_\epsilon \text{ such that } h(x) \neq h'(x)\}.$$

Then disagreement coefficient is

$$\theta = \sup_\epsilon \frac{\mathbb{P}[\mathsf{DIS}(H_\epsilon)]}{\epsilon}.$$

Example: $H = \{\text{thresholds in } \mathbb{R}\}$, any data distribution.



Therefore $\theta = 2$.

Disagreement coefficient: examples [H '07, F '09]

Disagreement coefficient: examples [H '07, F '09]

- Thresholds in $\mathbb{R}$, any data distribution.

$$\theta = 2.$$

Label complexity $O(\log 1/\epsilon)$.

Disagreement coefficient: examples [H '07, F '09]

- Thresholds in $\mathbb{R}$, any data distribution.

$$\theta = 2.$$

Label complexity $O(\log 1/\epsilon)$.

- Linear separators through the origin in $\mathbb{R}^d$, uniform data distribution.

$$\theta \leq \sqrt{d}.$$

Label complexity $O(d^{3/2} \log 1/\epsilon)$.

Disagreement coefficient: examples [H '07, F '09]

- Thresholds in $\mathbb{R}$, any data distribution.

  $$\theta = 2.$$

  Label complexity $O(\log 1/\epsilon)$.

- Linear separators through the origin in $\mathbb{R}^d$, uniform data distribution.

  $$\theta \leq \sqrt{d}.$$

  Label complexity $O(d^{3/2} \log 1/\epsilon)$.

- Linear separators in $\mathbb{R}^d$, smooth data density bounded away from zero.

  $$\theta \leq c(h^*)d$$

  where $c(h^*)$ is a constant depending on the target h^*.
  Label complexity $O(c(h^*)d^2 \log 1/\epsilon)$.