Active Learning

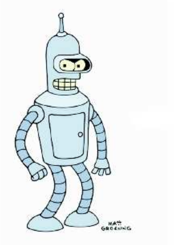
Piyush Rai

CS5350/6350: Machine Learning

November 10, 2011

(Passive) Supervised Learning

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



supervised learner
induces a classifier

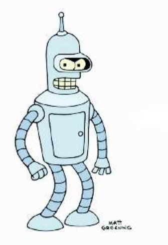
expert / oracle
analyzes experiments
to determine labels [1]

[1] Some figures from Burr Settles

(Passive) Supervised Learning

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

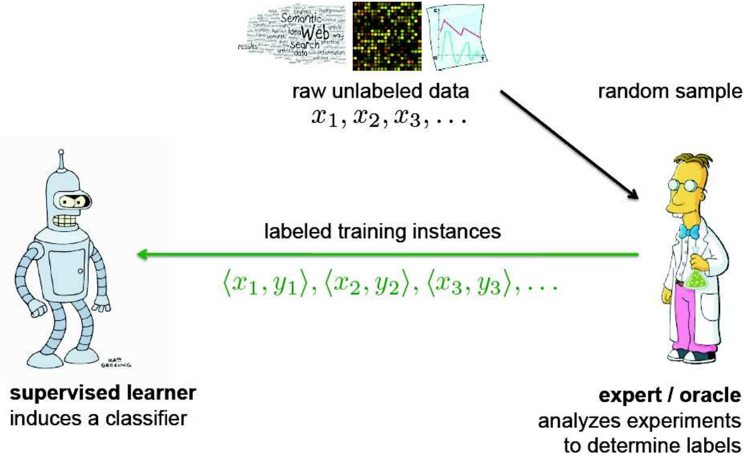
random sample



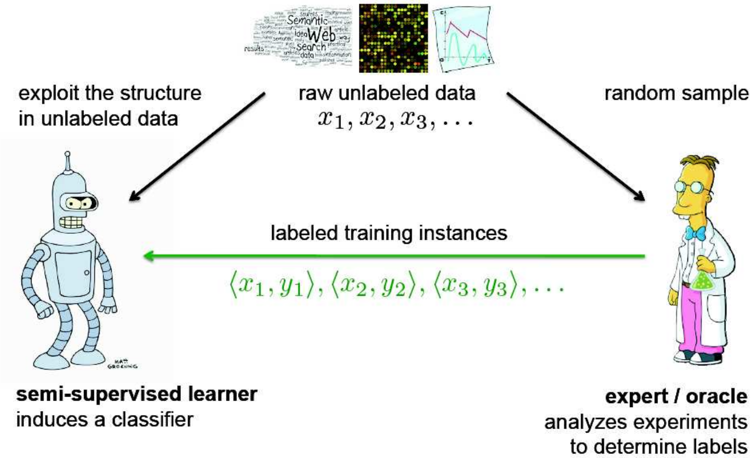
supervised learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

(Passive) Supervised Learning

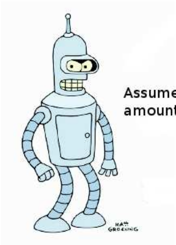


Semi-supervised Learning



Active Learning

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

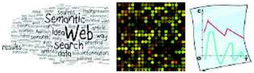


Assumes some small
amount of initial labeled training data

active learner
induces a classifier

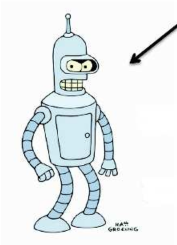
expert / oracle
analyzes experiments
to determine labels

Active Learning



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

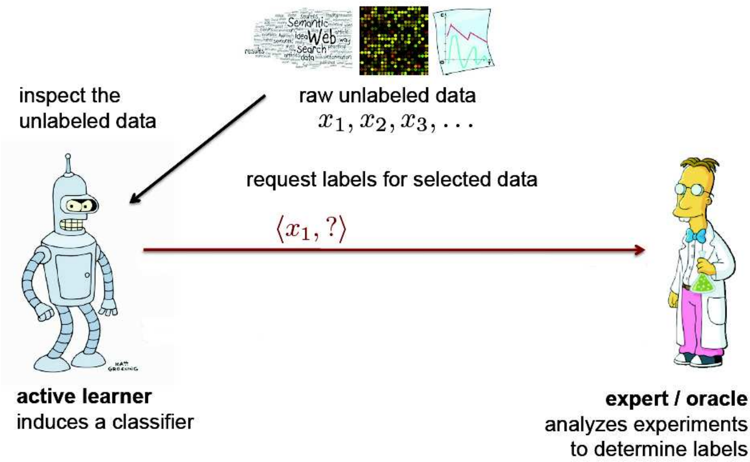
inspect the
unlabeled data



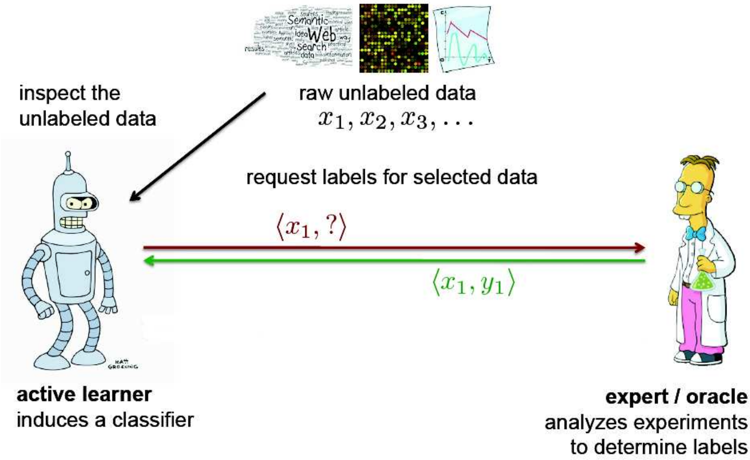
active learner
induces a classifier

expert / oracle
analyzes experiments
to determine labels

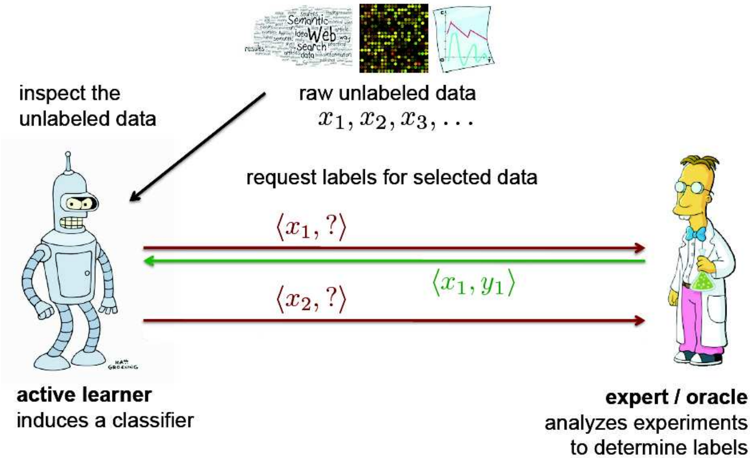
Active Learning



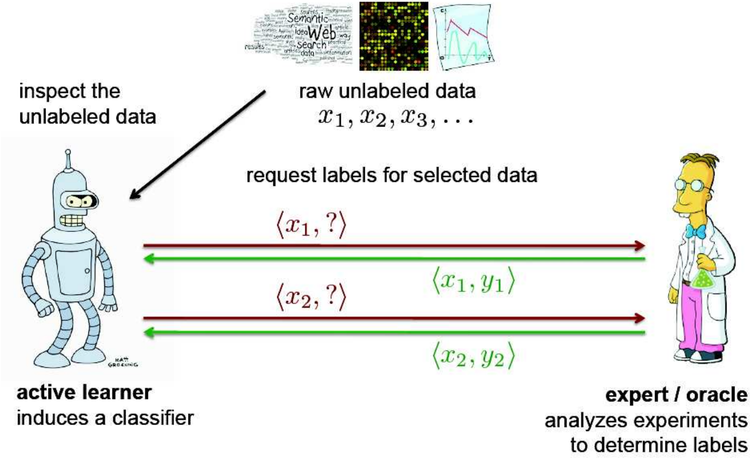
Active Learning



Active Learning

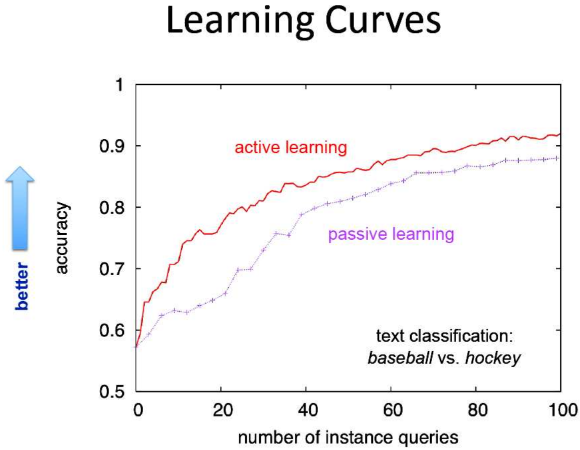


Active Learning



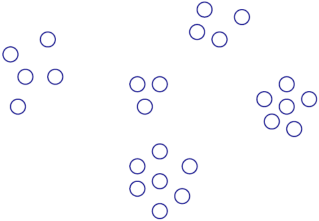
Active Learning vs Random Sampling

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



A Naïve Approach

Suppose the unlabeled data looks like this.



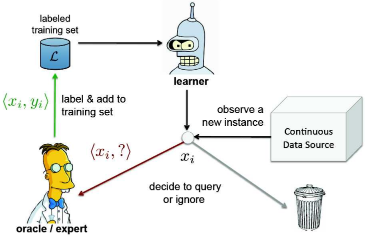
Then perhaps we just need five labels!

- Of course, thing could go wrong..

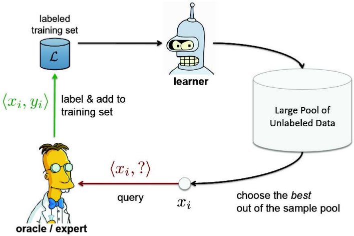
Types of Active Learning

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
- 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

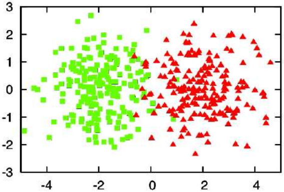
- Select examples which the current model θ is the most uncertain about
- Various ways to measure uncertainty. For example:
  - Based on the distance from the hyperplane
  - Using the label probability $P_\theta(y|\mathbf{x})$ (for probabilistic models)
- Some typically used measures based on label probabilities:
  - **Least Confident:** $x^*_{LC} = \text{argmax}_x\, 1 - P_\theta(\hat{y}|x)$
    where $\hat{y}$ is the most probable label for x under the current model θ
  - **Smallest Margin:** $x^*_{SM} = \text{argmin}_x\, P_\theta(y_1|x) - P_\theta(y_2|x)$
    y_1, y_2 are the two most probable labels for x under the current model
  - **Label Entropy:** choose example whose label entropy is maximum

$$x^*_{LE} = \underset{x}{\text{argmax}} - \sum_i P_\theta(y_i|x) \log P_\theta(y_i|x)$$

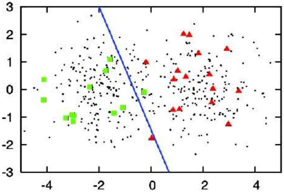
    where y_i ranges over all possible labels

Uncertainty Sampling

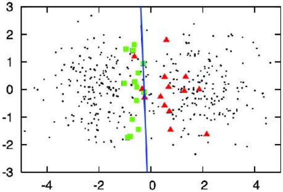
A simple illustration of uncertainty sampling based on the distance from the hyperplane (i.e., margin based)



400 instances sampled
from 2 class Gaussians



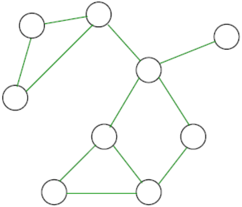
random sampling
30 labeled instances
(accuracy=0.7)



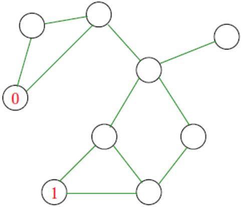
uncertainty sampling
30 labeled instances
(accuracy=0.9)

Uncertainty Sampling based on Label-Propagation

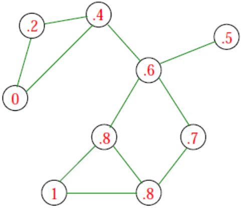
(1) Build neighborhood graph



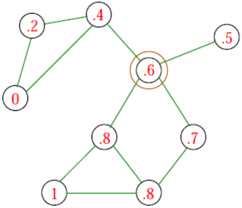
(2) Query some random points



(3) Propagate labels



(4) Make query and go to (3)



Query By Committee (QBC)

- QBC uses a committee of models $\mathcal{C} = \{\theta^{(1)}, \ldots, \theta^{(C)}\}$
- All models trained using the currently available labeled data $\mathcal{L}$
- How is the committee constructed? Some possible ways:
  - Sampling different models from the model distribution $P(\theta|\mathcal{L})$
  - Using ensemble methods (bagging/boosting, etc.)
- All models vote their predictions on the unlabeled pool
- The example(s) with maximum disagreement is/are chosen for labeling
- One way of measuring disagreement is the Vote Entropy
  - Vote Entropy

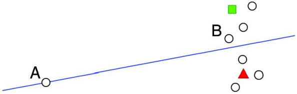
$$x^*_{VE} = \underset{x}{\text{argmax}} - \sum_i \frac{V(y_i)}{C} \log \frac{V(y_i)}{C}$$

  y_i ranges over all possible labels, $V(y_i)$: number of votes received to label y_i

- 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)



- 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)
- **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!

A Perceptron Based Active Learner

Based on Selective Sampling (looking at one example at a time)

- Input: Parameter $b > 0$ (dictates how aggressively we want to query labels)
- Initialization: $\mathbf{w} = [0\ 0\ 0 \ldots 0]$
- For $n = 1 : N$
  1. Get $\mathbf{x}_n$, compute $p_n = \mathbf{w}^\top \mathbf{x}_n$
  2. Predict $\hat{y}_n = \text{sign}(p_n)$
  3. Draw Bernoulli random variable $Z \in \{0, 1\}$ with probability $\frac{b}{b+|p_n|}$
  4. If $Z == 1$, query the true label y_n
     - If $y_n \neq \hat{y}_n$ **then** update $\mathbf{w}$, **else** don't update $\mathbf{w}$
  5. Else if $Z == 0$, ignore the example $\mathbf{x}_n$ and don't update $\mathbf{w}$

Comments:

- $|p_n|$ is the absolute margin of $\mathbf{x}_n$
- Large $|p_n| \Rightarrow$ Small label query probability
- Expected number of labels queried $= \sum_{n=1}^{N} \mathbb{E}[\frac{b}{b+|p_n|}]$

Concluding Thoughts..

- 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.)
- Some questions worth thinking about (read the Active Learning survey)
  1. Can I reuse an actively labeled dataset to train a new different model?
  2. 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?