Semi-supervised and Active Learning

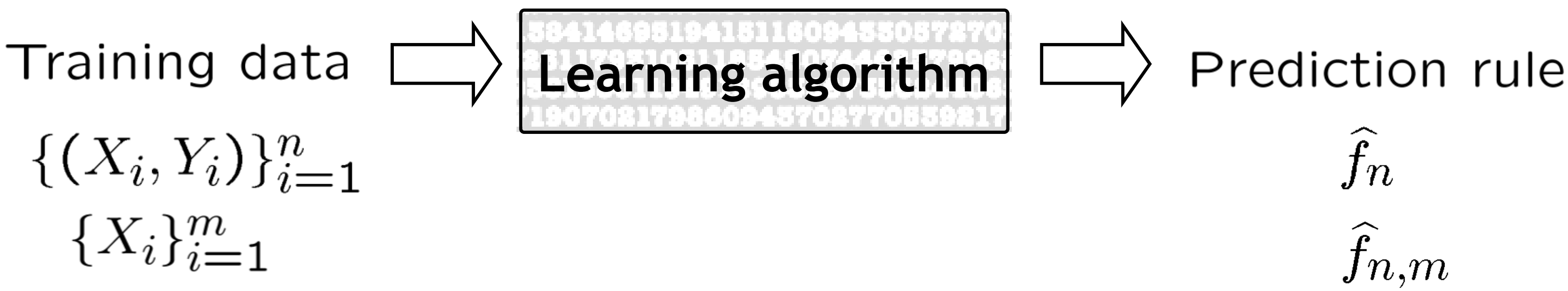
4/22

Amr

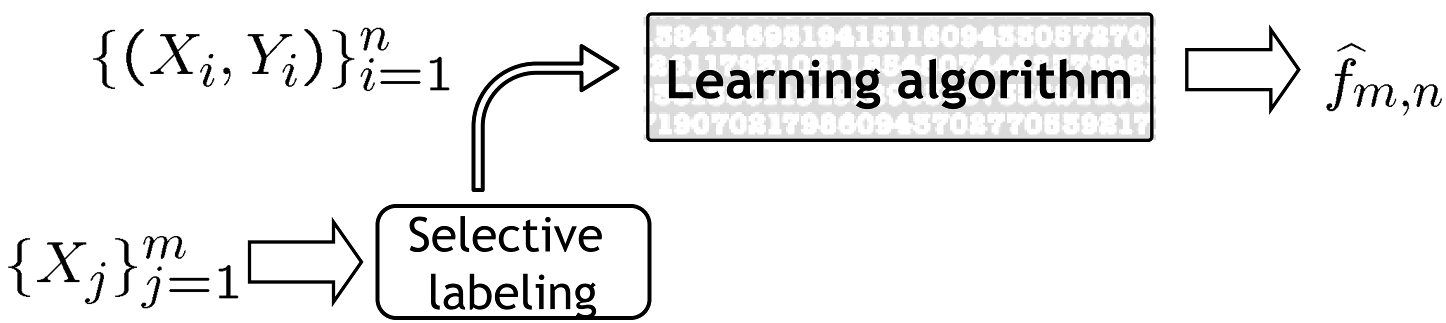
Credit: lecture slides

The big picture

- Semi-supervised Learning



- Active Learning



How?

- There is no free lunch!
- You need to make assumption
- Leverage them to construct an algorithm
- If assumption are correct we can improve

Assumption: Overview

both try to attack the same problem: making the most of unlabeled data $\mathcal{U}$

uncertainty sampling

query instances the model is least confident about

self-training
expectation-maximization (EM)

propagate confident labelings among unlabeled data

query-by-committee (QBC)

use ensembles to rapidly reduce the version space

co-training
multi-view learning

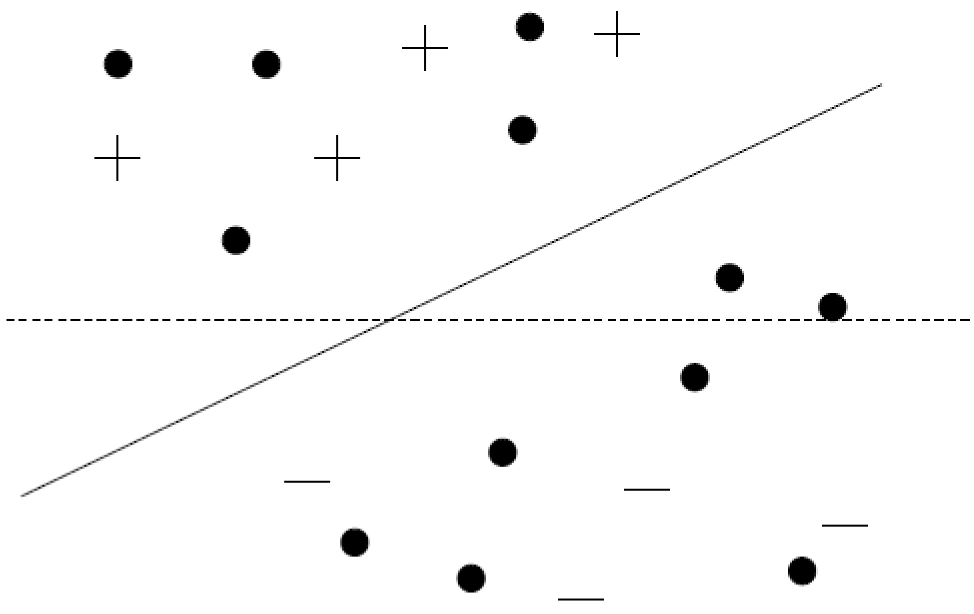
use ensembles with multiple views to constrain the version space

Semi-supervised

- If x and x’ are similar, then they are likely to have the same label
- Algorithm
  - Assume generative model
  - Cluster and label
  - Regularize the classifier using unlabeled data
  - Multi-view learning

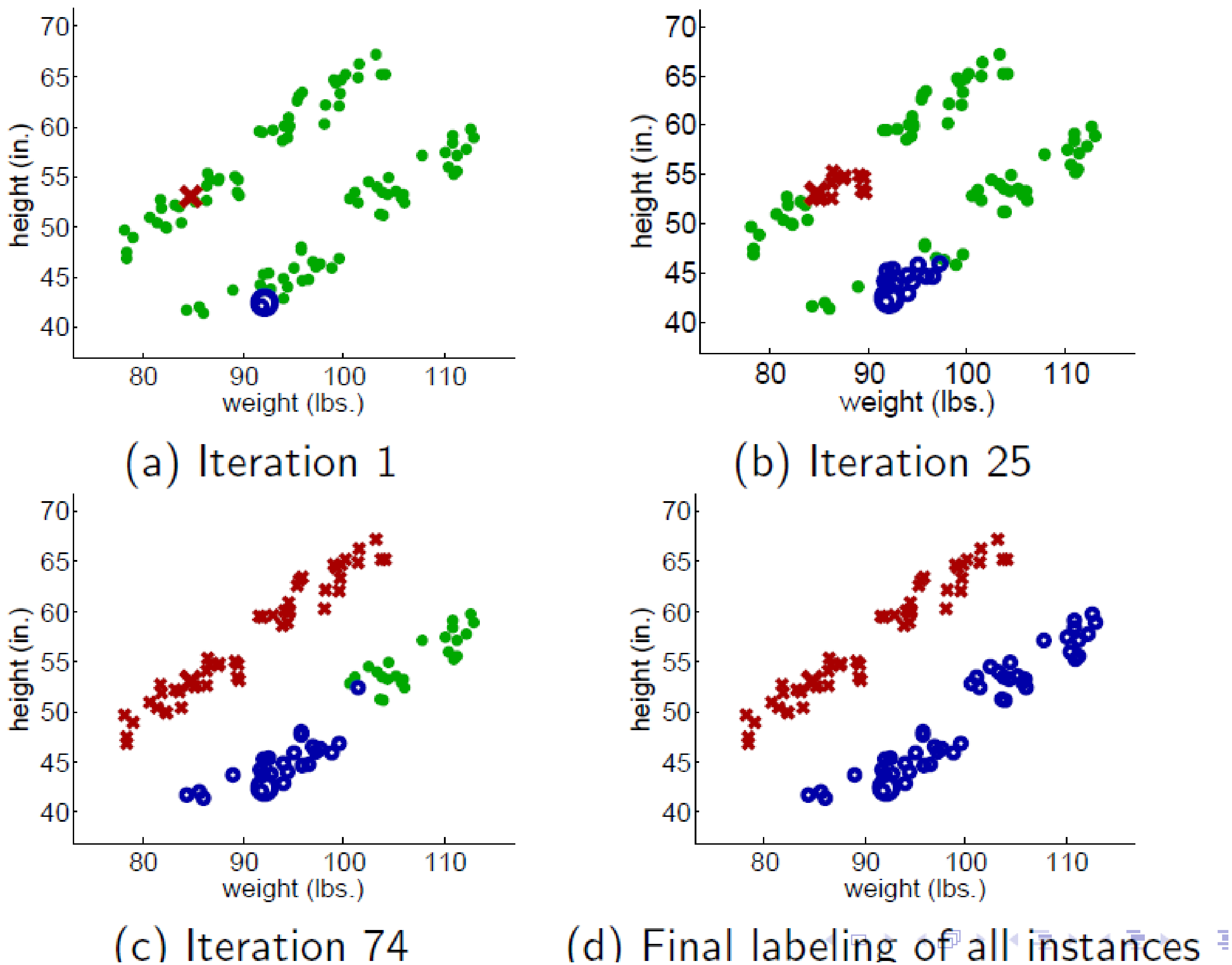
- Does it help?

Example



Examples: 1-NN, works!

Propagating 1-Nearest-Neighbor: now it works



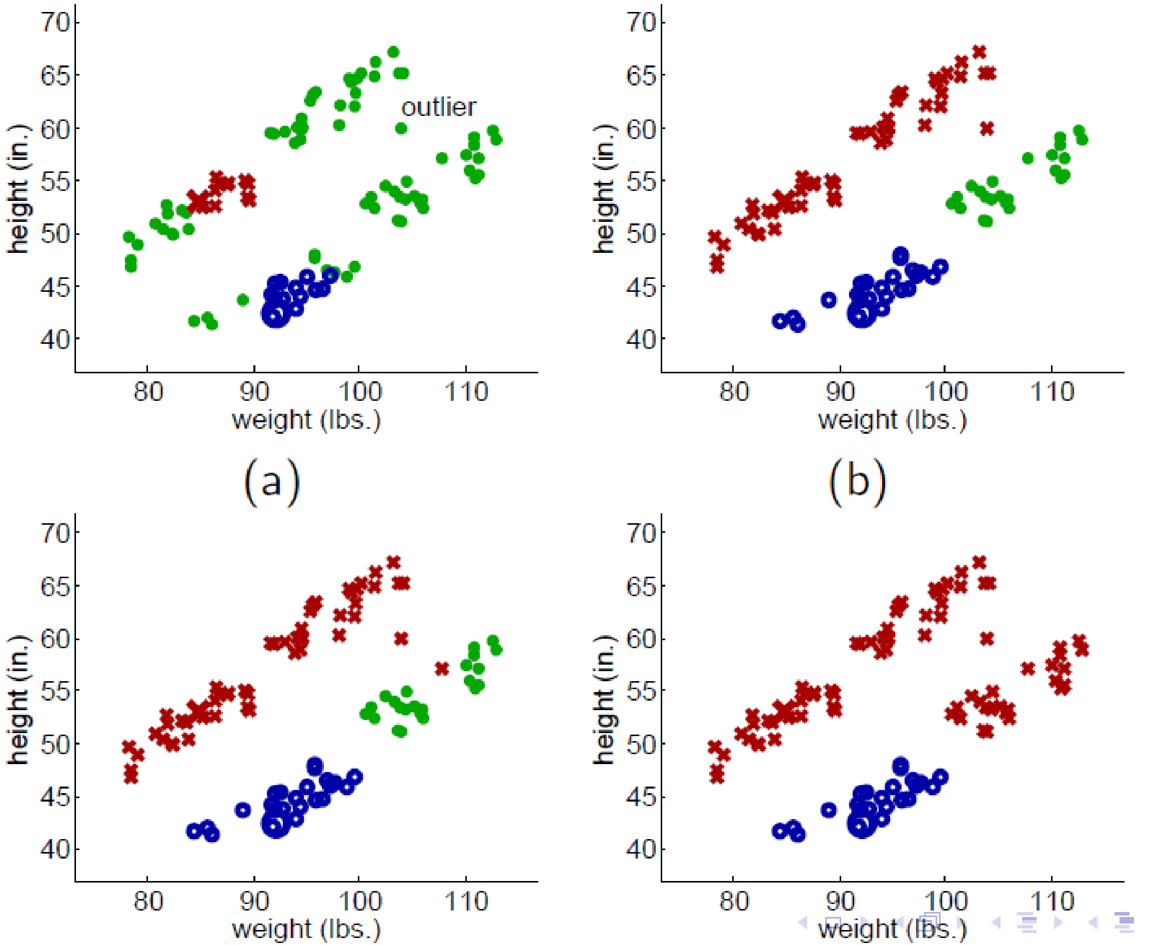
(a) Iteration 1 (b) Iteration 25

(c) Iteration 74 (d) Final labeling of all instances

Example: 1-NN, doesn't work

Propagating 1-Nearest-Neighbor: now it doesn't

But with a single outlier...

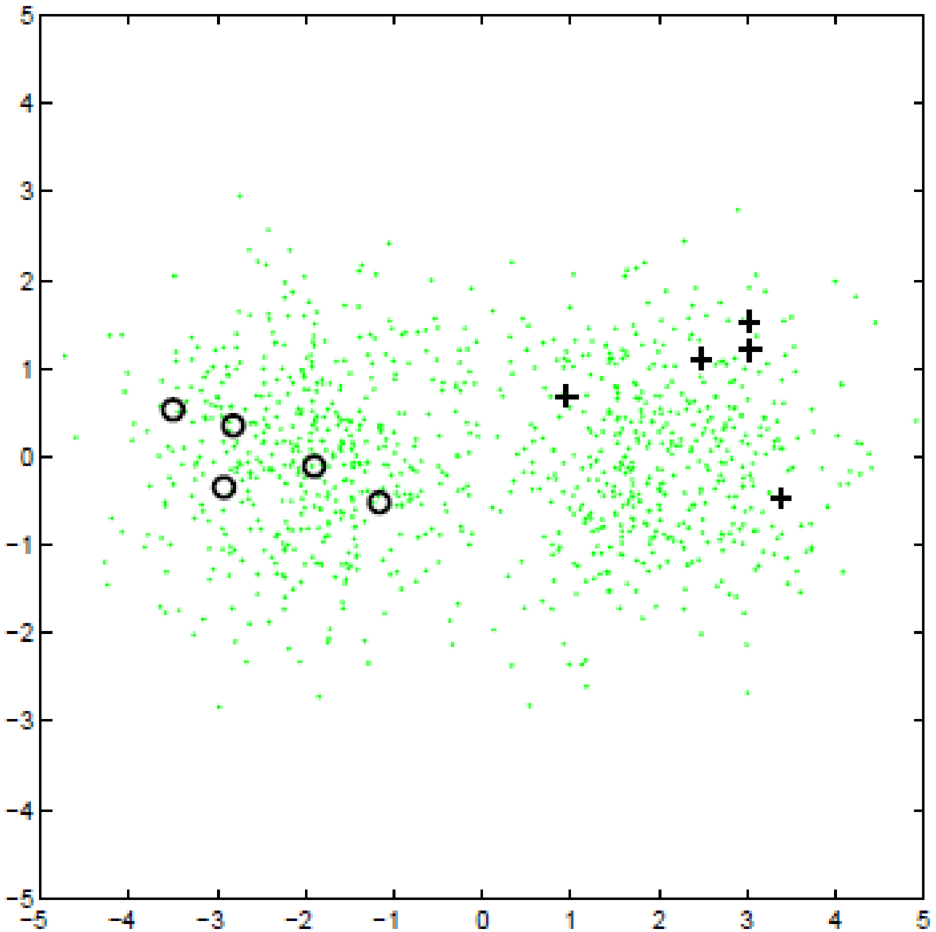


Can we be more robust?

- So in general how to deal with this problem?
  - Generative model
  - Regularization

SSL using Mixture Models

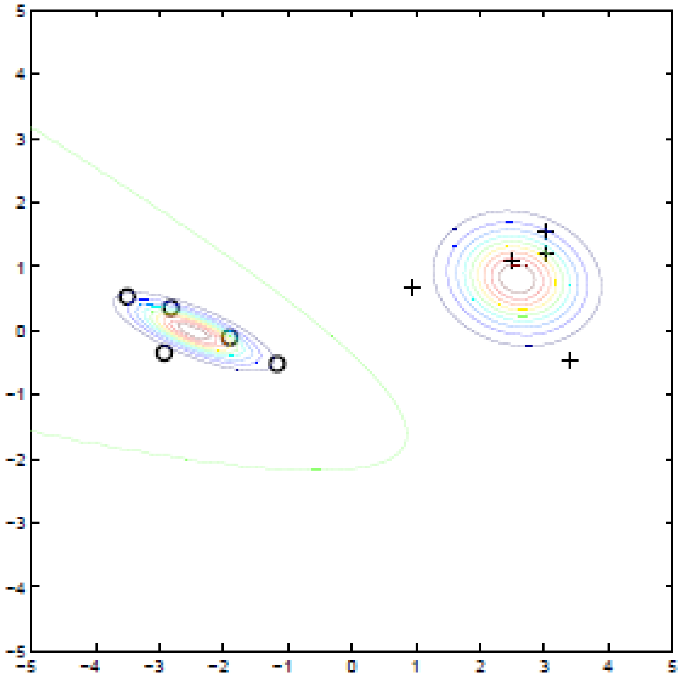
- Use all data not one at a time!



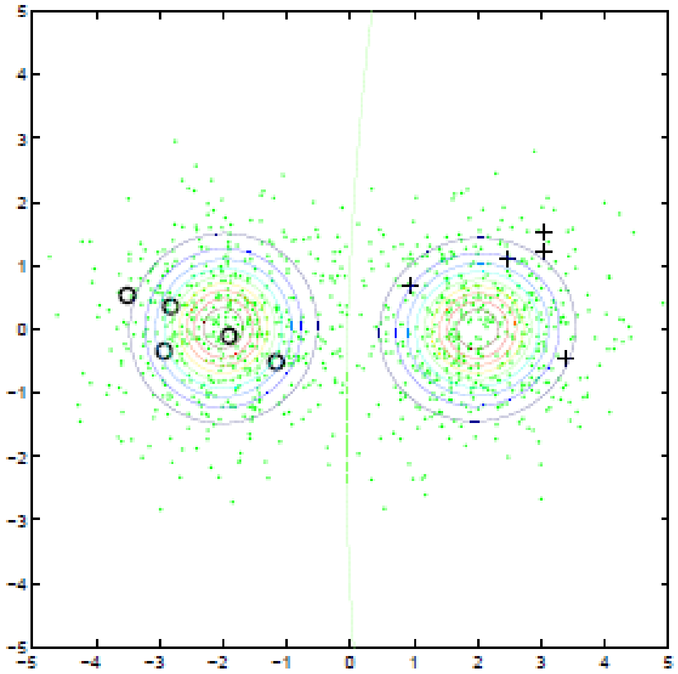
SSL using Mixture Models

They are different because they maximize different quantities.

$p(X_l, Y_l|\theta)$



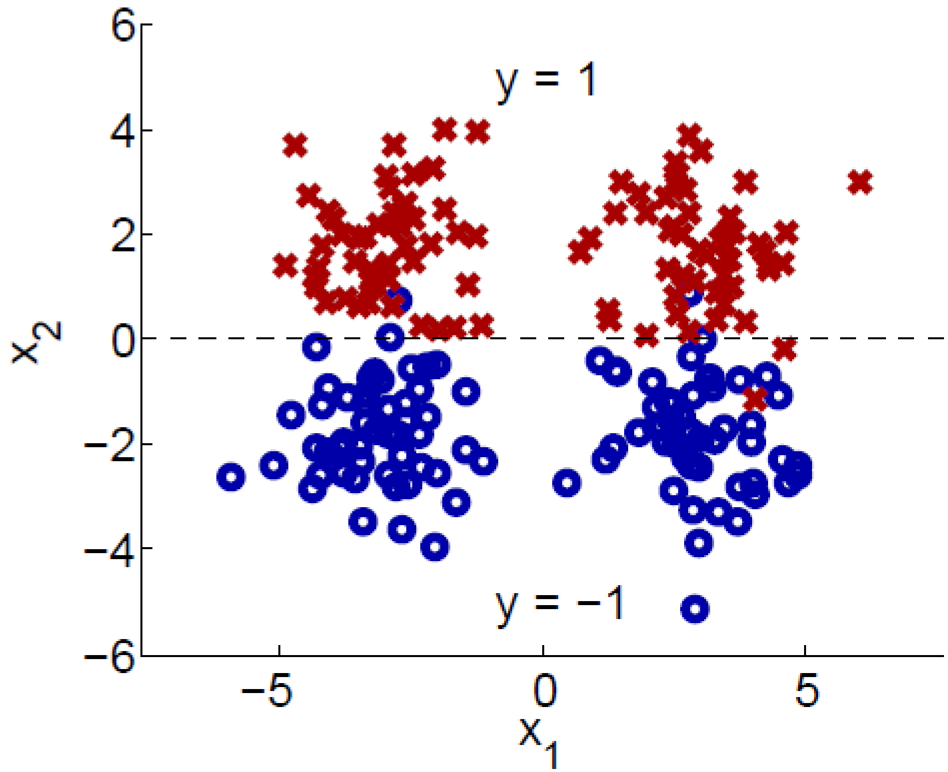
$p(X_l, Y_l, X_u|\theta)$



SSL using Mixture Models

- Inference and learning
  - This was your midterm problem!
  - You know more than you think you do!
- Is this robust to noise?
  - At least you can get Bayes optimal if assumption is correct
  - What if assumption are wrong?

- When the assumption is wrong:



Can we be more robust?

- So in general how to deal with this problem?
  - Generative model
  - Regularization

So why a new method

- As we said earlier
- Different kind of assumption
- What if data is not Gaussian?
  - Remember spectral clustering

Graph Regularization

- Regularized classifier
- Learn a classifier that minimize
  - Loss term + regularize
  - Example: ridge regression
- Can we use unlabeled data for regularization?

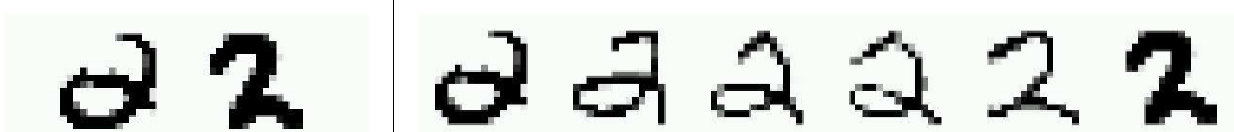
$$\min_f \underbrace{\sum_{i\in l}(y_i - f_i)^2}_{\substack{\text{Loss on labeled data} \\ \text{(mean square,0-1)}}} + \lambda \underbrace{\sum_{i,j\in l,u} w_{ij}(f_i - f_j)^2}_{\substack{\text{Graph based smoothness prior} \\ \text{on labeled and unlabeled data}}}$$

Is it robust?

$$\min_f \underbrace{\sum_{i \in l} (y_i - f_i)^2}_{\substack{\text{Loss on labeled data} \\ \text{(mean square,0-1)}}} + \lambda \underbrace{\sum_{i,j \in l,u} w_{ij}(f_i - f_j)^2}_{\substack{\text{Graph based smoothness prior} \\ \text{on labeled and unlabeled data}}}$$

- You can play with the regularization parameter
- Sensitive to graph construction

Handwritten digits recognition with pixel-wise Euclidean distance



not similar	'indirectly' similar with stepping stones

The big picture

- Semi-supervised Learning

There is no free lunch

Training data → **Learning algorithm** → Prediction rule

$\{(X_i, Y_i)\}_{i=1}^{n}$ $\qquad$ $\widehat{f}_n$

$\{X_i\}_{i=1}^{m}$ $\qquad$ $\widehat{f}_{n,m}$

- Active Learning

$\{(X_i, Y_i)\}_{i=1}^{n}$

$\{X_j\}_{j=1}^{m}$ → Selective labeling → **Learning algorithm** → $\widehat{f}_{m,n}$

Pay a little than passive

Active Learning

- Passive learning
  - Input a set of example
  - Output a classifier
- Observation:
  - Labels are expensive
  - Sometime you can get the same classifier with subset of the data
    - Example?

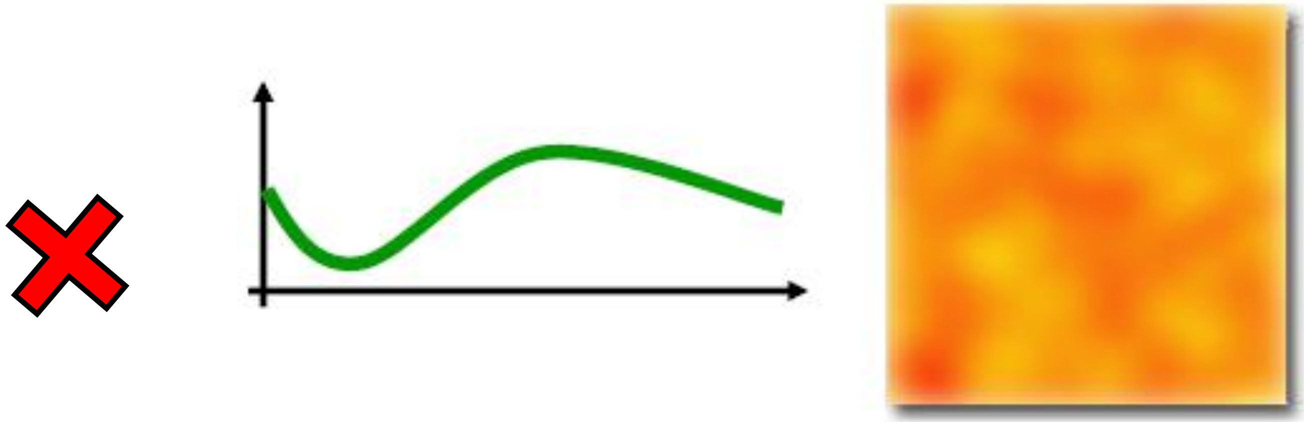
Active Learning

- SVM
  - Only need support vector
- Is it that easy?
- What assumption are we making here?
  - Noise free environment
- In general, we need a localized function

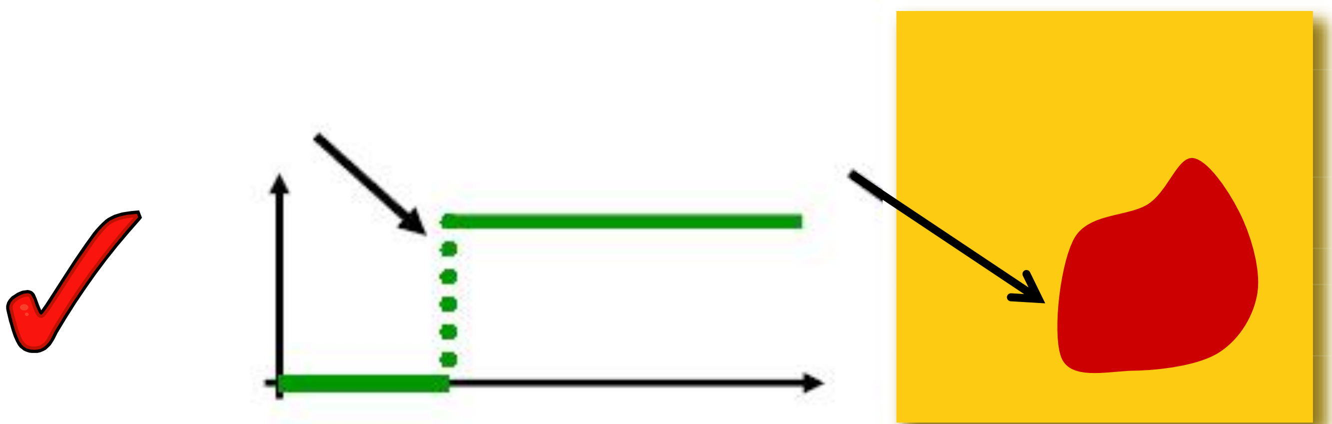
Active Learning

- When does it help?

[Castro et al.,'05]



Passive = Active

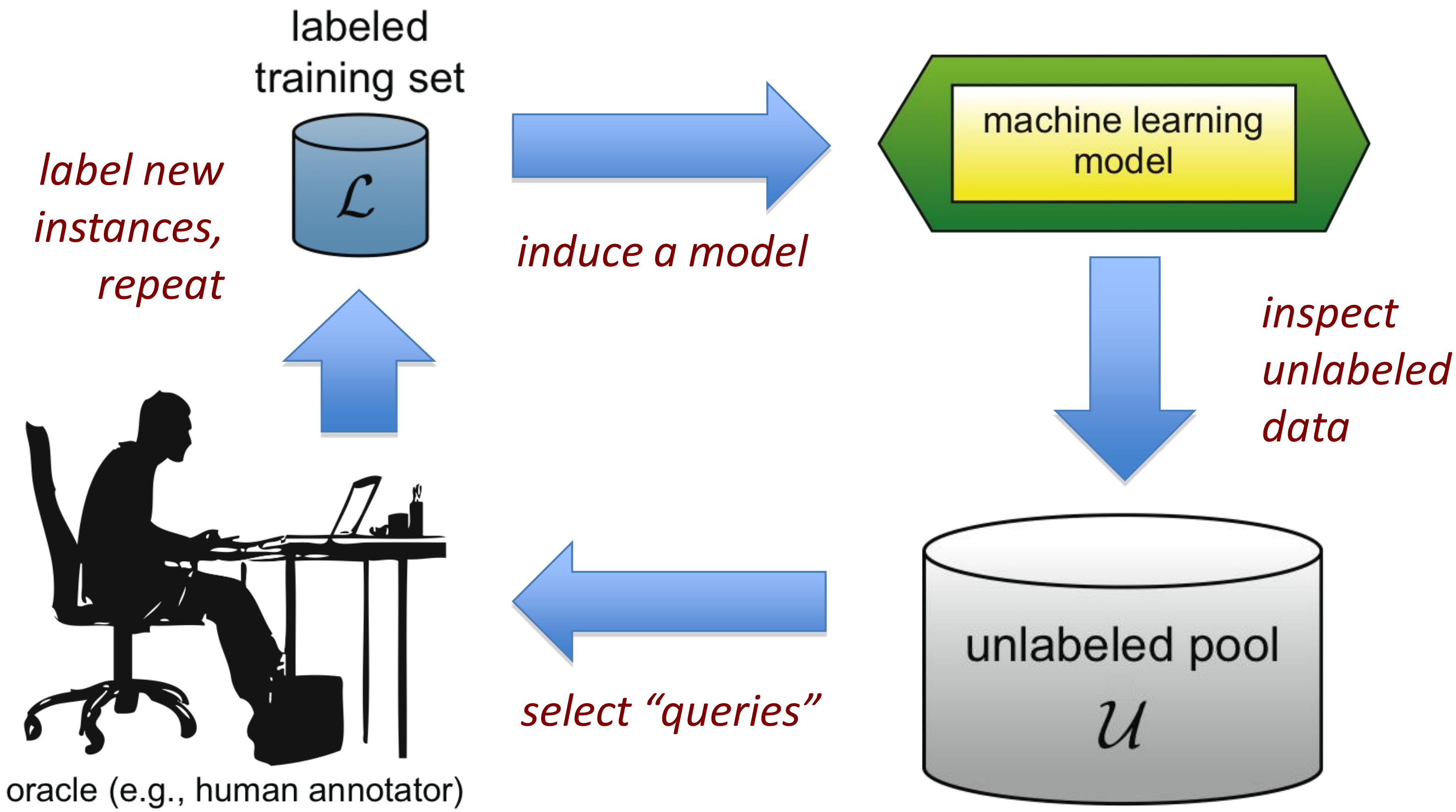


$$\epsilon \sim n^{-\frac{1}{d}}$$

$$\epsilon \sim n^{-\frac{1}{d-1}}$$

Active learning is useful if complexity of target function is localized – labels of some data points are more informative than others.

Active Learning setup

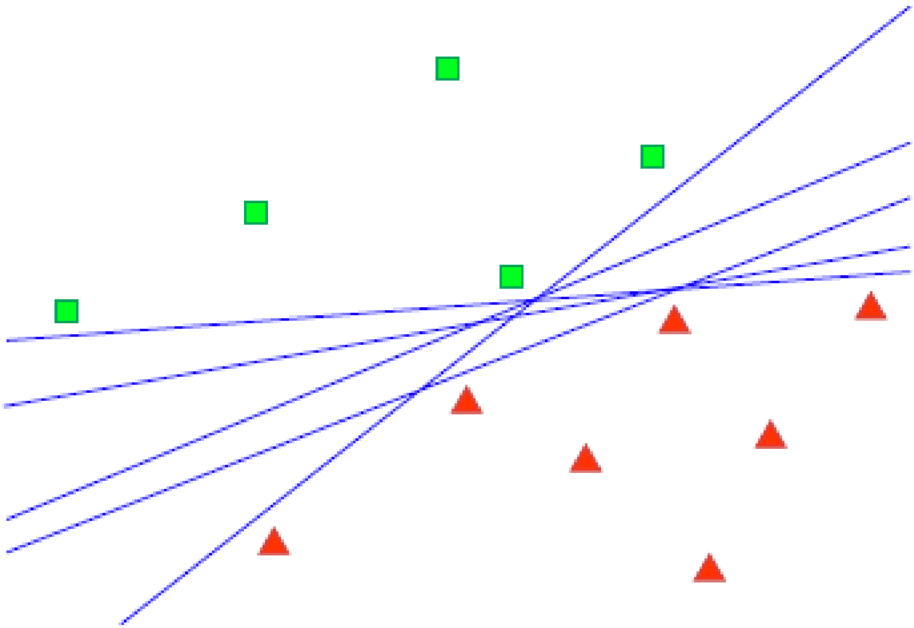


Algorithms from Insights

- We need to learn a decision boundary
- Classification uncertainty
  - Query example closer to decision boundary
  - We become more confident if we get them right
  - Somehow this is still local decisions
- Version-Space uncertainty
  - Some how makes global decision

Version Space

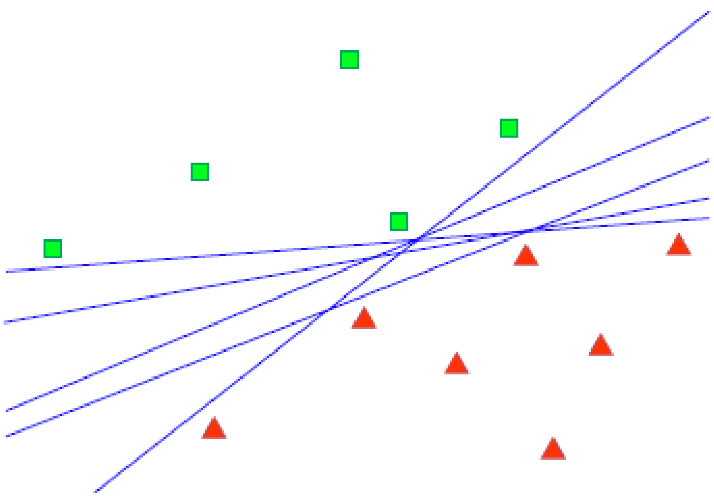
- Set of hypothesis consistent with labeled examples



(a)

Version Space

- Our goal: get a single hypothesis
- Select example that results in maximum reduction of hypothesis space
- What is the problem with that?



(a)

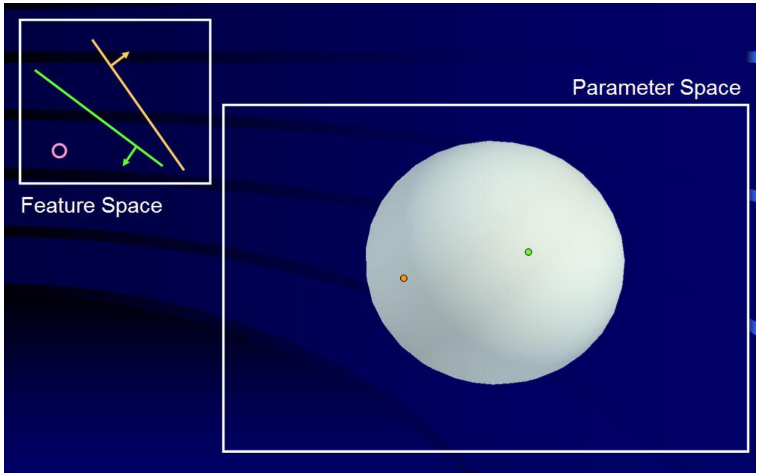
Version space: Algorithm

- Query by committee
  - Keep an ensemble of classifiers to approximate
- Goal reduce “entropy” over their contributions
- Idea
  - Sample from P(parameters| data)

Case study: SVM

- How to represent version space

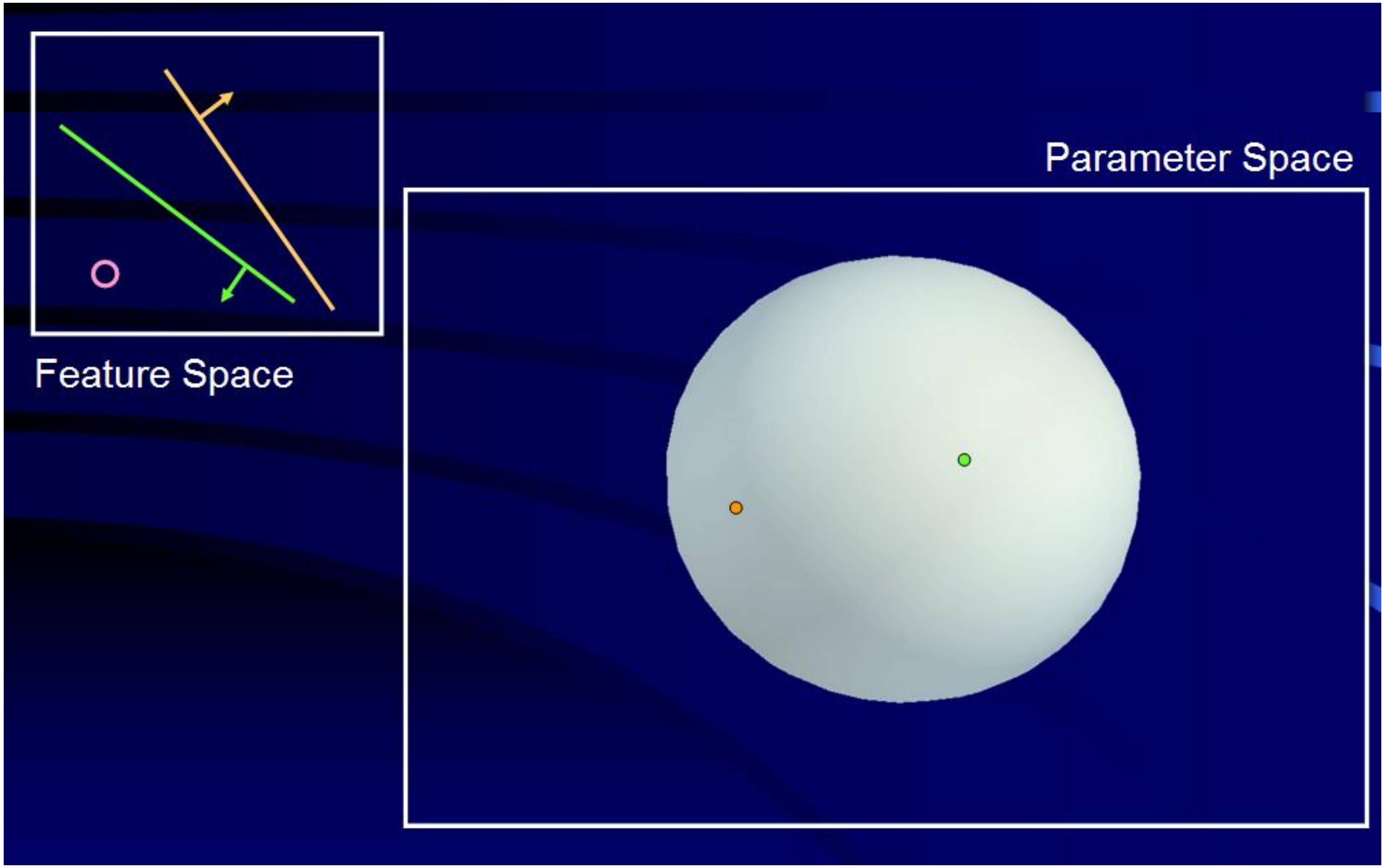
$$
\begin{array}{ll}
\text{maximize}_{\mathbf{w}\in\mathcal{F}} & \min_i\{y_i(\mathbf{w}\cdot\Phi(\mathbf{x}_i))\} \\
\text{subject to:} & \|\mathbf{w}\| = 1 \\
 & y_i(\mathbf{w}\cdot\Phi(\mathbf{x}_i)) > 0 \quad i = 1\ldots n.
\end{array}
$$

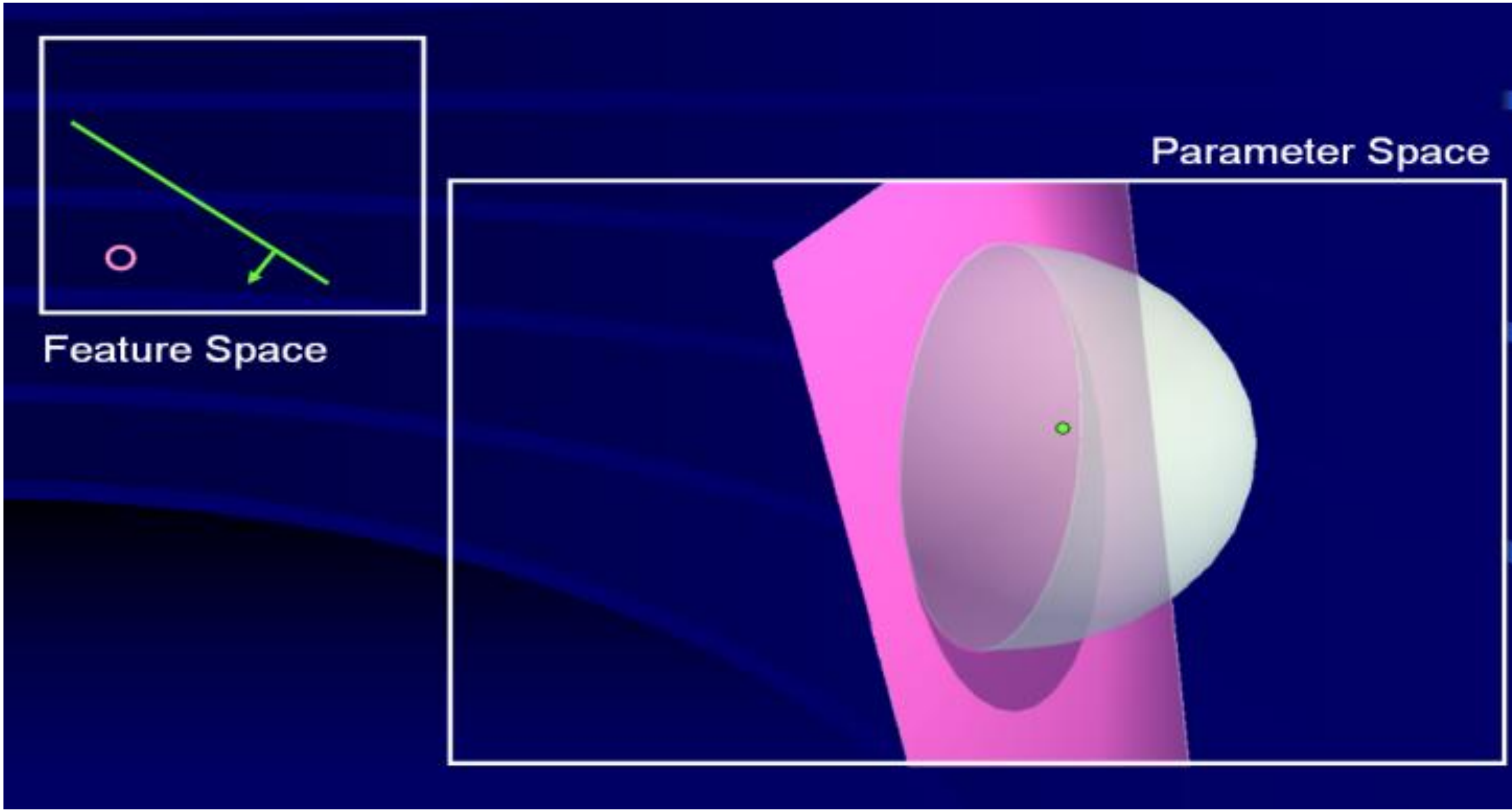


- This is slightly re-parameterized SVM objective but it is the same

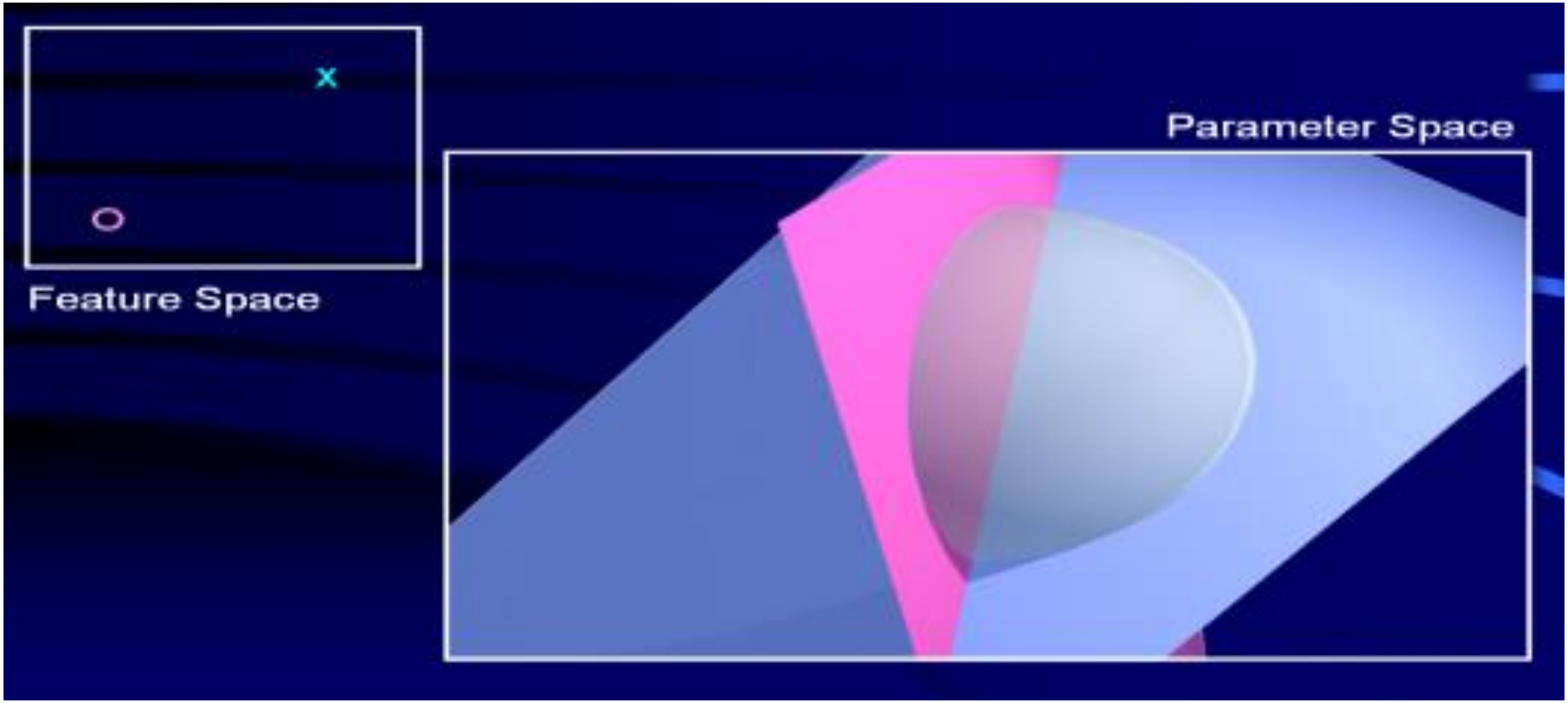
Case study: SVM

- How to represent version space





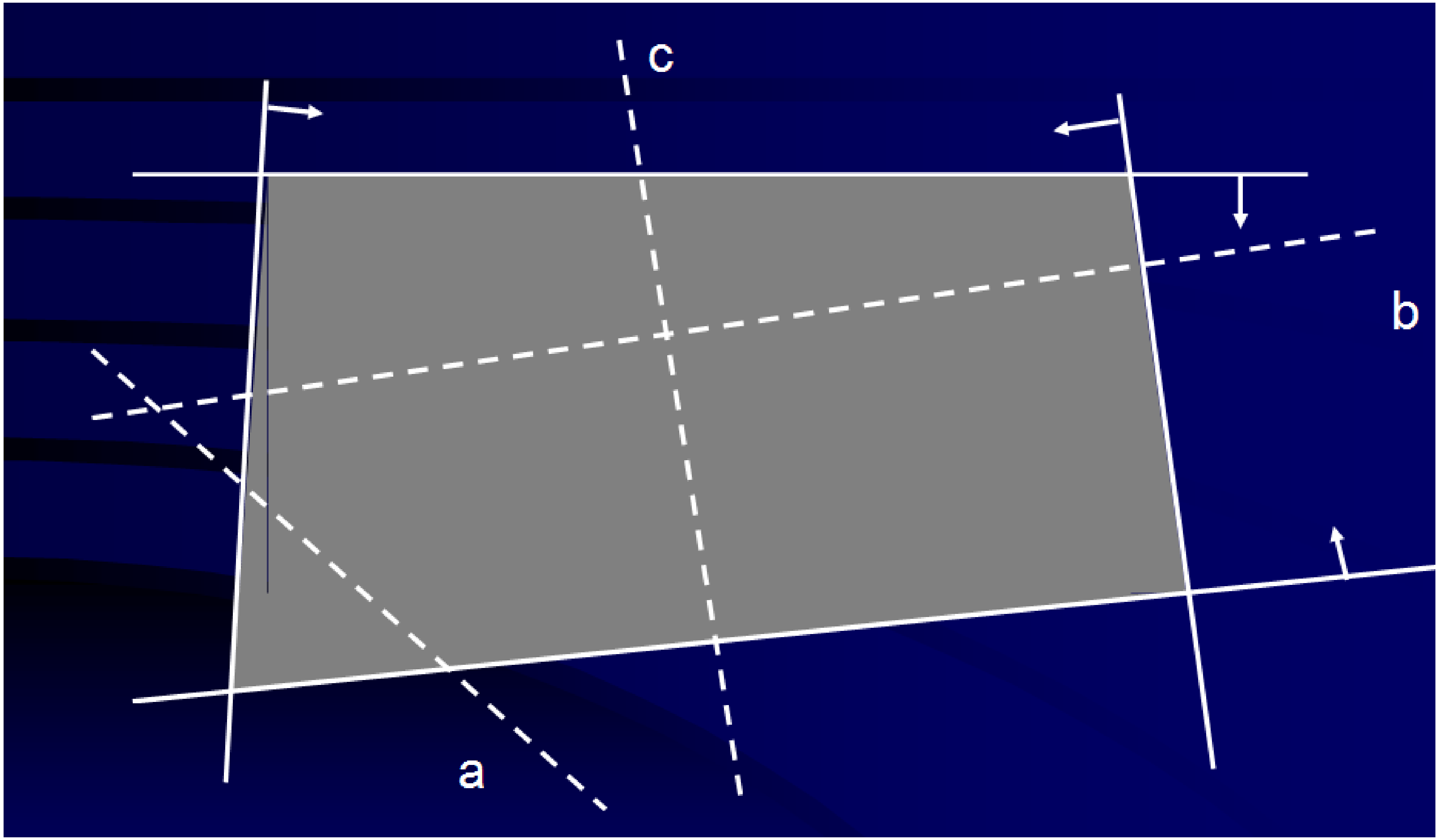
Parameter Space
Feature Space



Given the current labeled data we have an explicit representation of the version space

Query point

- Halving the version space (query point c)



Is it the End?

- Supervised
- Semi-supervised
- Active
- Transductive
  - You still get to see unlabeled data
  - But these are also your test data
  - What can you do with that?

Transductive SVM

- Chose a confident labeling of unlabeled data

Minimize over $(y_1^*, \ldots, y_n^*, \vec{w}, b)$:

$$\frac{1}{2}||\vec{w}||^2$$

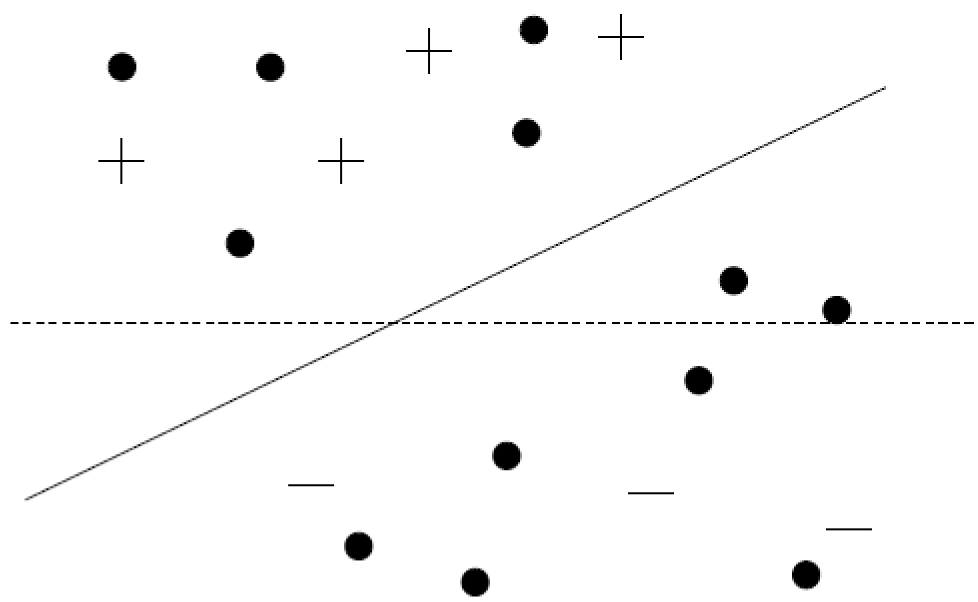
subject to: $\forall_{i=1}^{n} : y_i[\vec{w} \cdot \vec{x}_i + b] \geq 1$

$$\forall_{j=1}^{k} : y_j^*[\vec{w} \cdot \vec{x}_j^* + b] \geq 1$$

Unlabeled data

Transductive SVM

- Why does it make sense?



Transductive SVM

- When is it useful?
- News filtering
  - Labeled data: news users liked in the past
  - Test data (unlabeled): today's news
  - We only need to do well on those test data