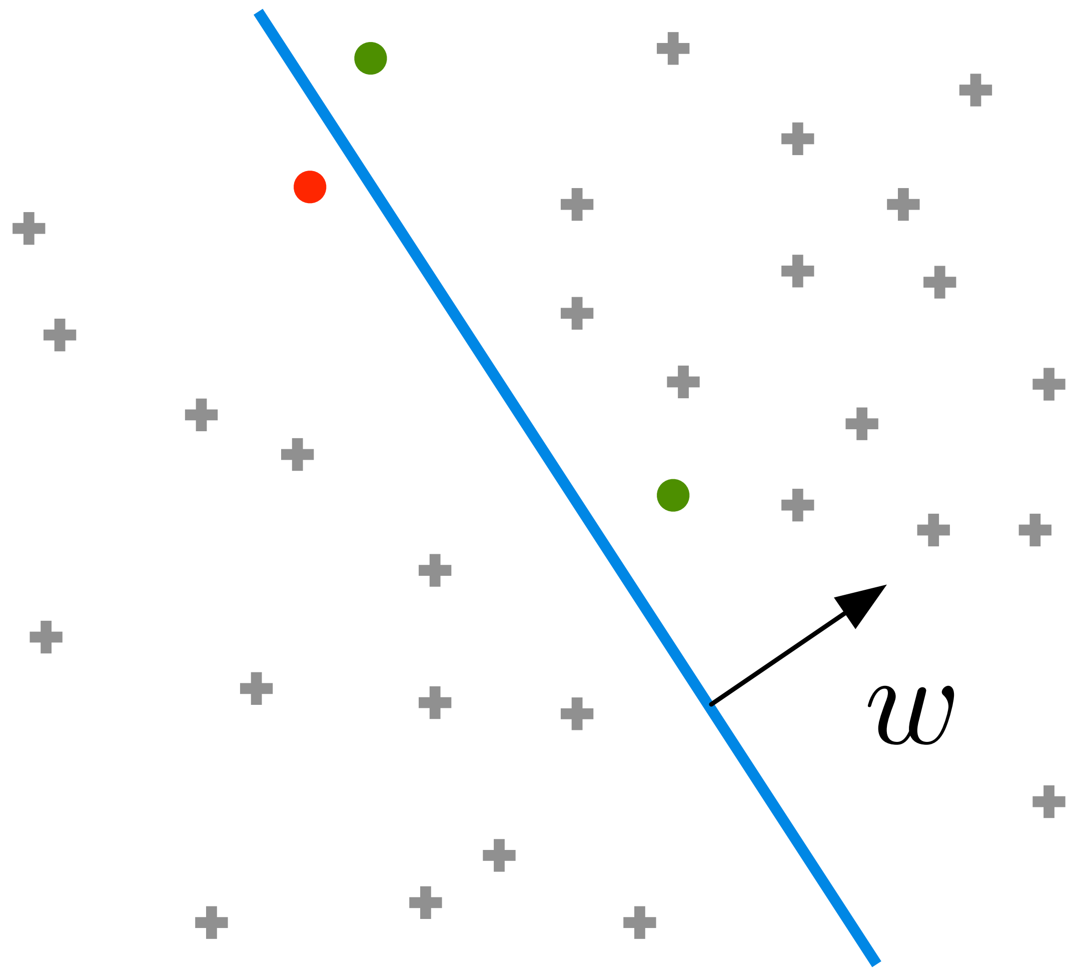
Active Learning

STAT 37710 / CMSC 35300

Rebecca Willett and Yuxin Chen

Example: learning under a budget



- Labels are expensive
- Want to minimize the number of labels

How can we obtain most "useful" labels at minimum cost?

How does “active” learning help?

Example: Learning threshold functions in 1-D

$$H = \{h : [0,1] \to \{0,1\}, h(x) = \text{sign}(x - \tau) \text{ for some } \tau \in [0,1]\}$$

Approaches to active learning

Stream-based active learning

- Data arrives one point at a time, and we must decide whether to request the label or not

Pool-based active learning

- We get an unlabeled data set, and must sequentially request points to label

Pool-based active learning

Pool-based active learning

- Obtain large pool of unlabeled data
- Selectively request a few labels, until we can infer all remaining labels

Resulting classifier as good as that obtained from complete data

- Reduction in labels
- In some cases, exponential reduction possible!

Uncertainty sampling

Algorithm: uncertainty sampling

Input: a pool of n unlabeled examples; learning algorithm $\mathcal{A}$

Repeat until we can infer all remaining labels with $\mathcal{A}$:

- Assign each unlabeled data x an "uncertainty score"

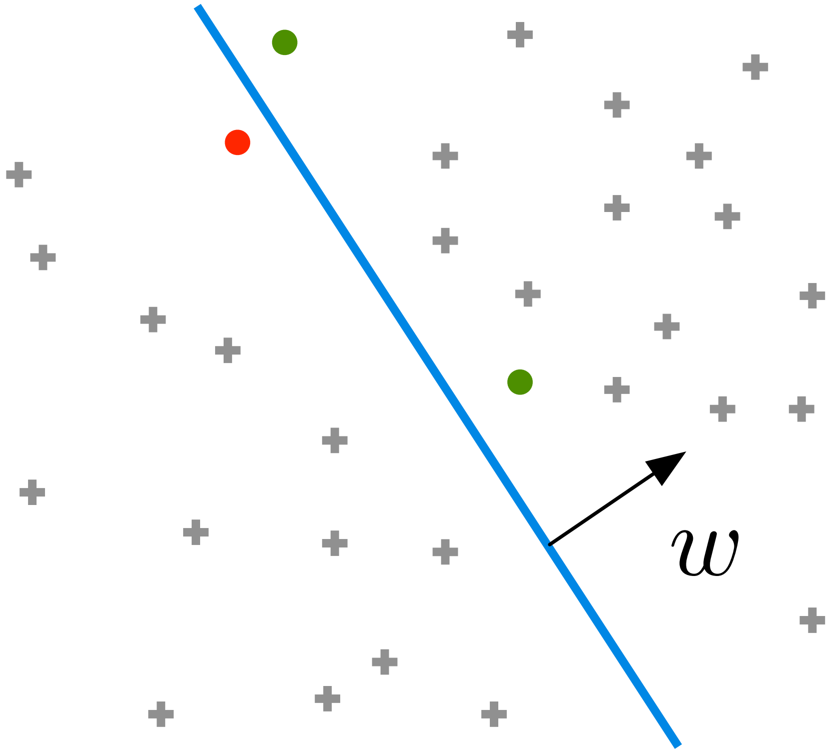
$$U_t(x) = U(x \mid x_{1:t-1}, y_{1:t-1})$$

- Greedily pick the most uncertain example and request label

$$x_t = \arg\max_x U_t(x)$$

- Retrain classifier: $A(x_{1:t}, y_{1:t})$

Uncertainty sampling in SVM



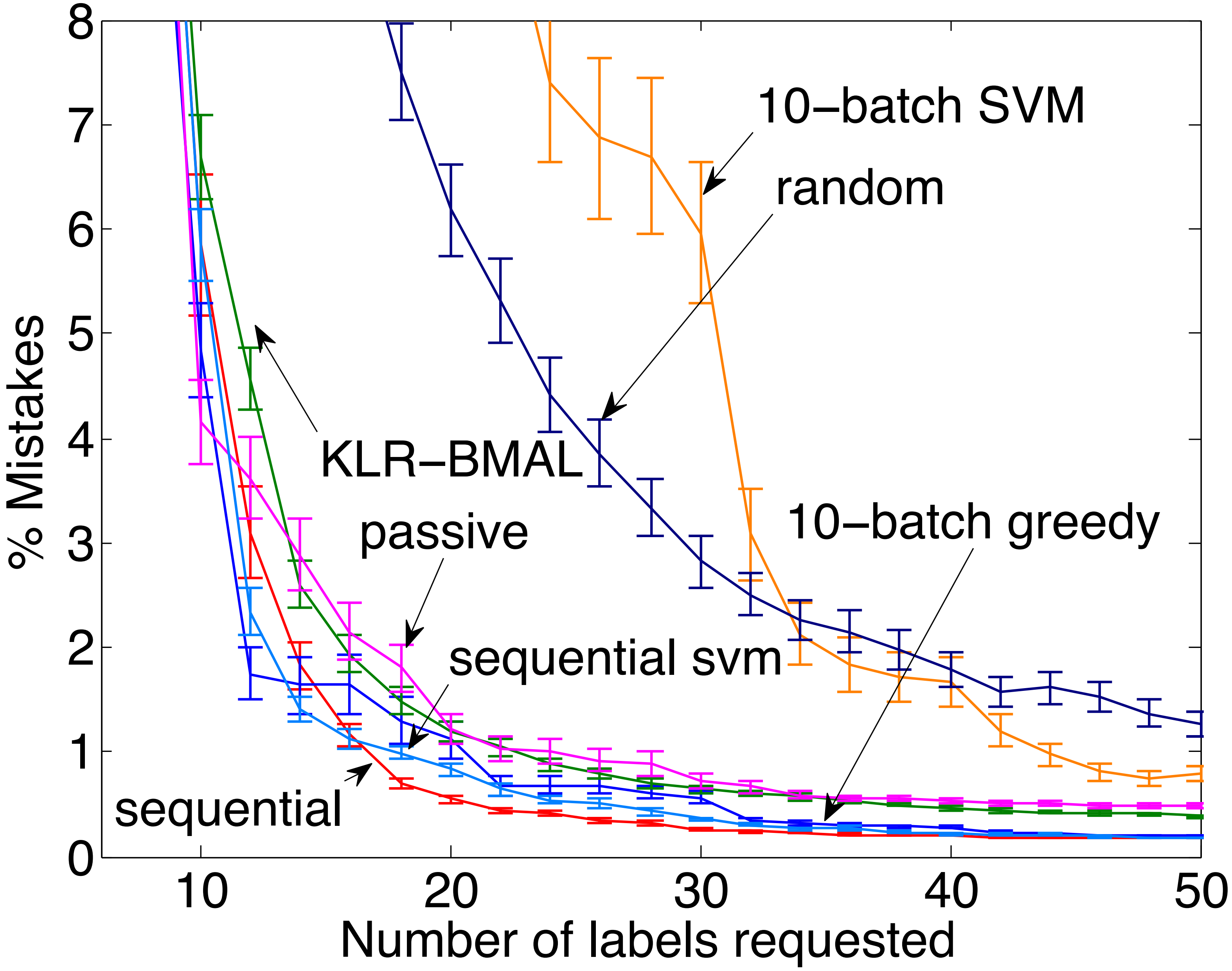
Select point nearest to decision boundary

$$U_t(x) = U(x \mid x_{1:t-1}, y_{1:t-1}) = \frac{1}{|w_{t-1}^\top x|}$$

$$x_t = \arg\max_x U_t(x) = \arg\min \left| w_{t-1}^\top x \right|$$

[Tong & Koller, 2000; Schohn & Cohn, 2000]..

Experimental results: uncertainty sampling



[Chen & Krause, 2013]

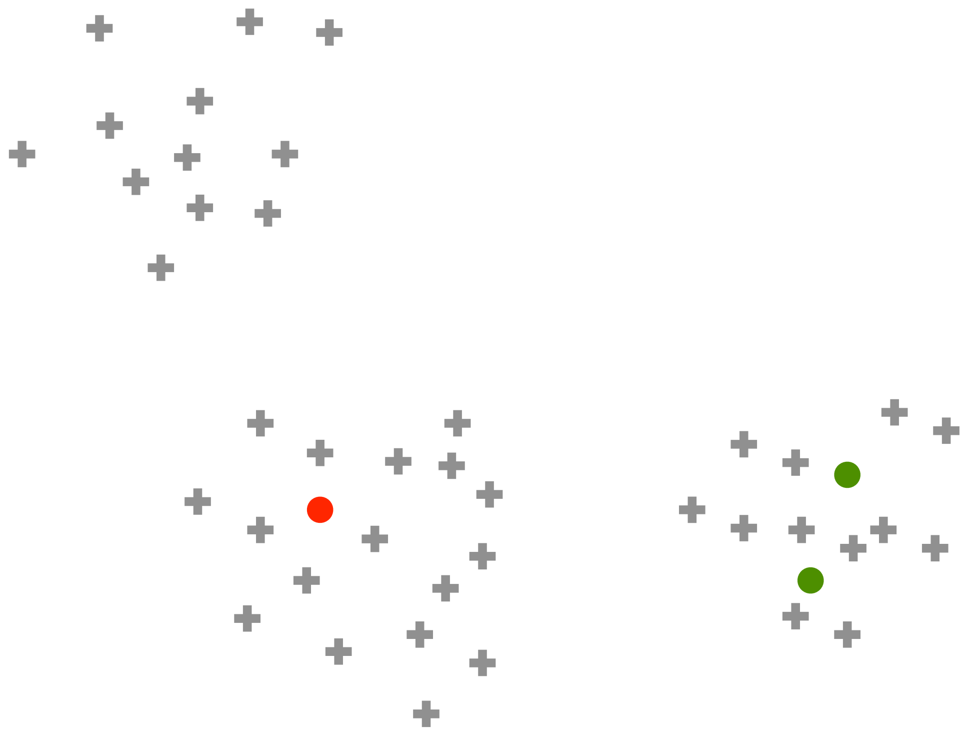
Issues with uncertainty sampling

Computational issue

- What is the computational cost to pick m labels over n unlabled data points?

$$O(n \cdot m \cdot \mathsf{cost}(U)) + O(m \cdot \mathsf{cost}(\mathcal{A}))$$

- Query in (mini-)batches of size k
- Sublinear time active selection via hashing [Jain, Vijayanarasimhan & Grauman, NeurIPS 2010]



Active learning bias

- uncertain $\neq$ **informative**

Defining “informativeness”

Need to capture how much “information” we gain about the true classifier for each label

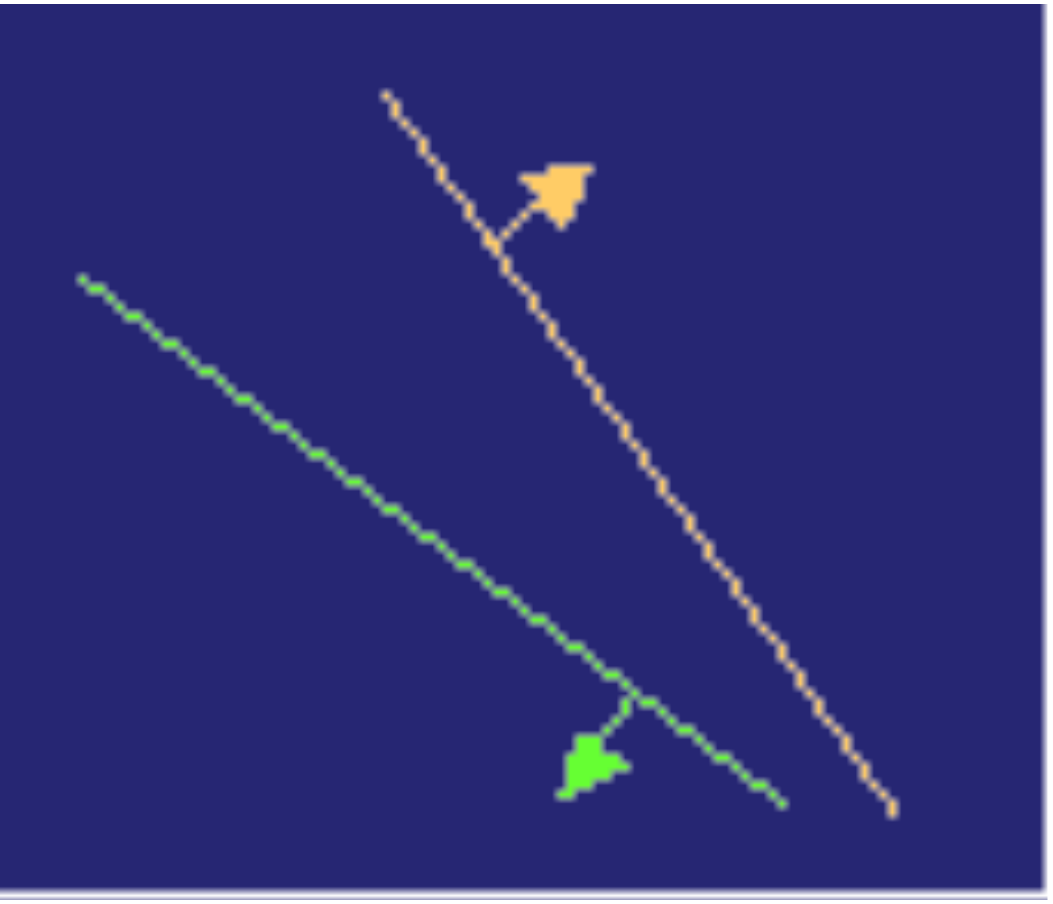
Version space: set of all classifiers consistent with the data

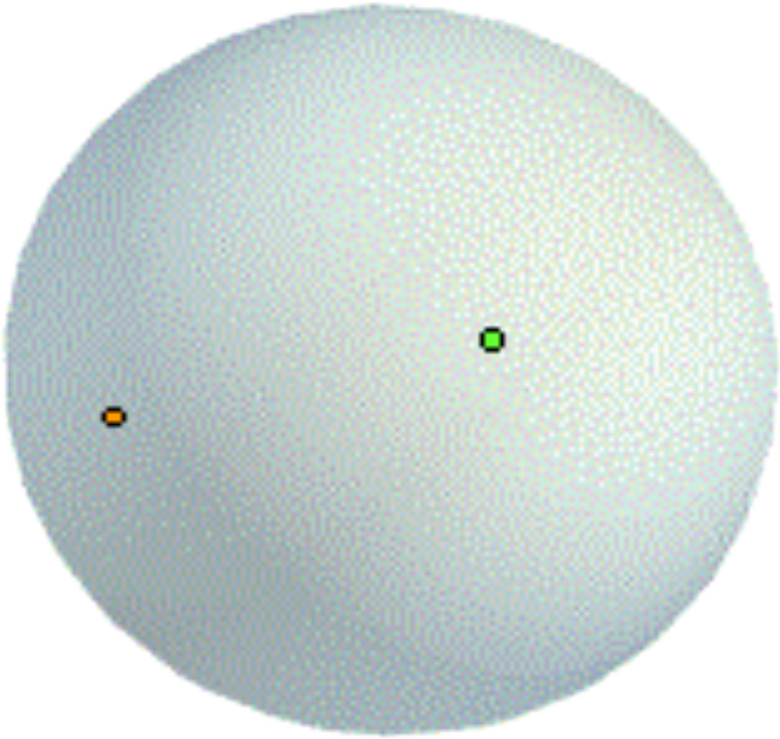
$$\mathcal{V}(D) = \{w : \forall (x,y) \in D, \mathsf{sign}(w^\top x) = y\}$$

- would like to shrink version space as quickly as possible

Version space for SVM (I)

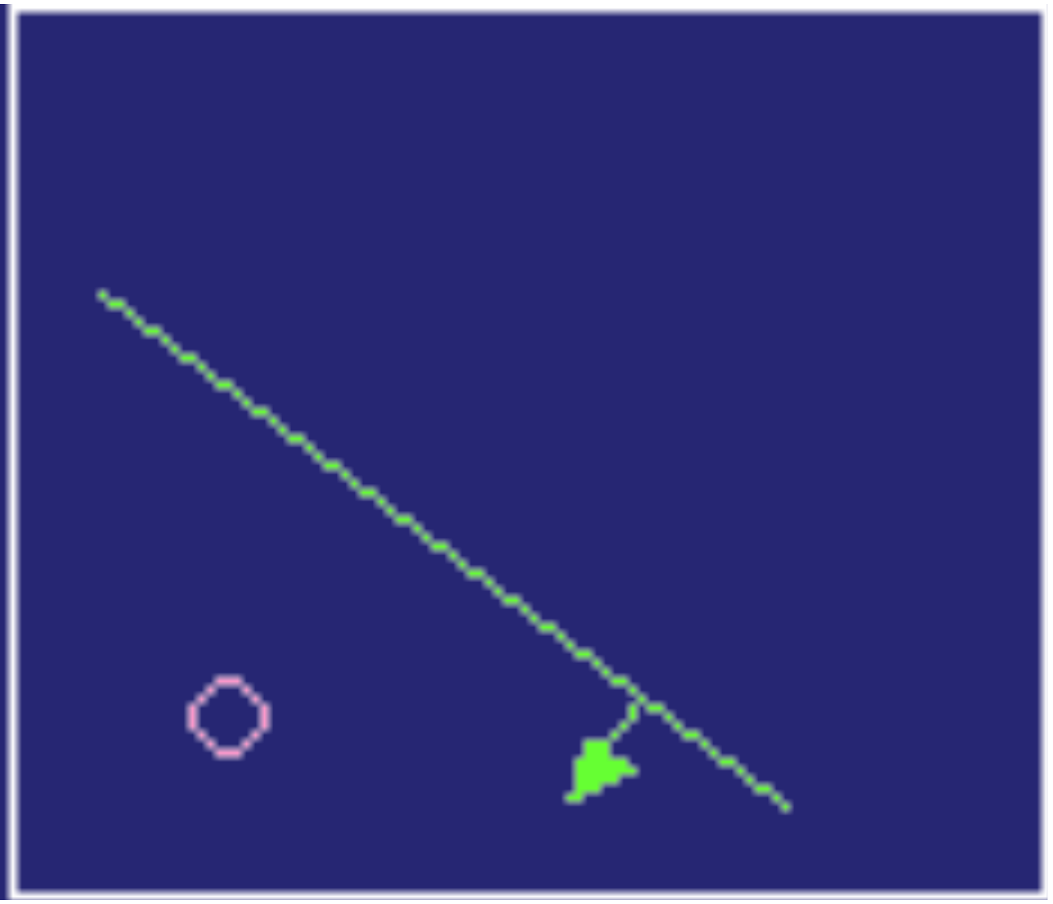
[Tong & Koller]



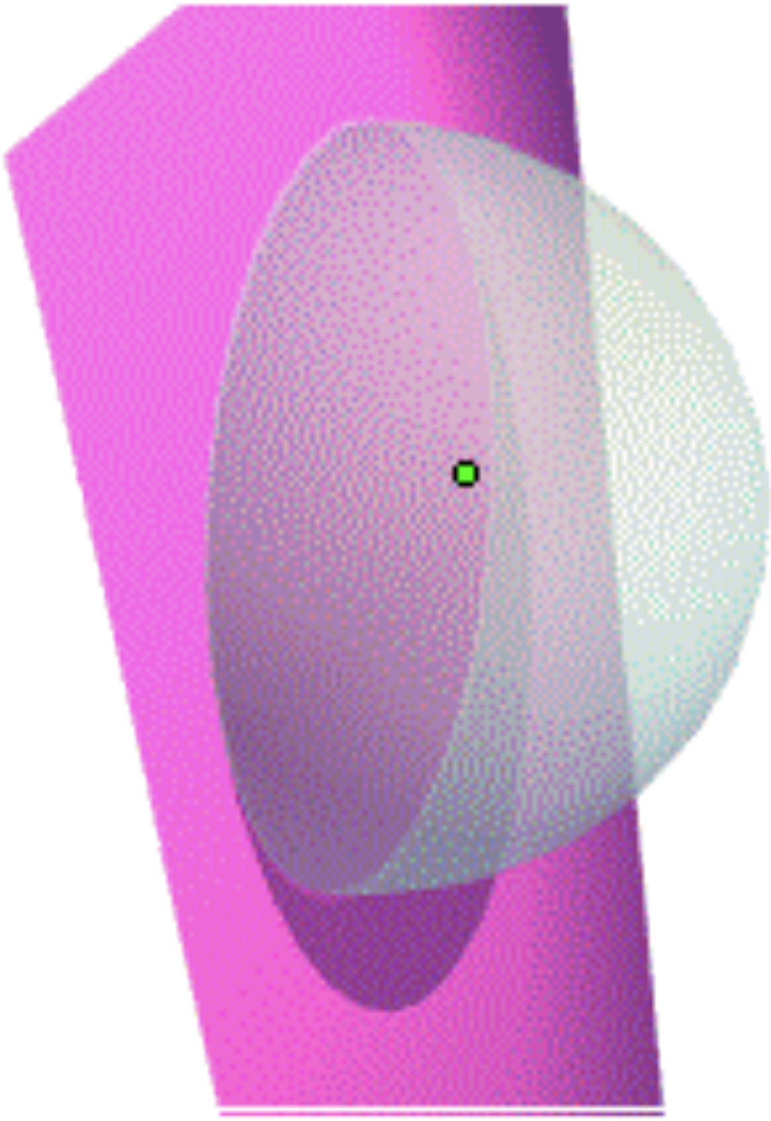


Src: https://las.inf.ethz.ch/courses/datamining-f17/

Version space for SVM (II)

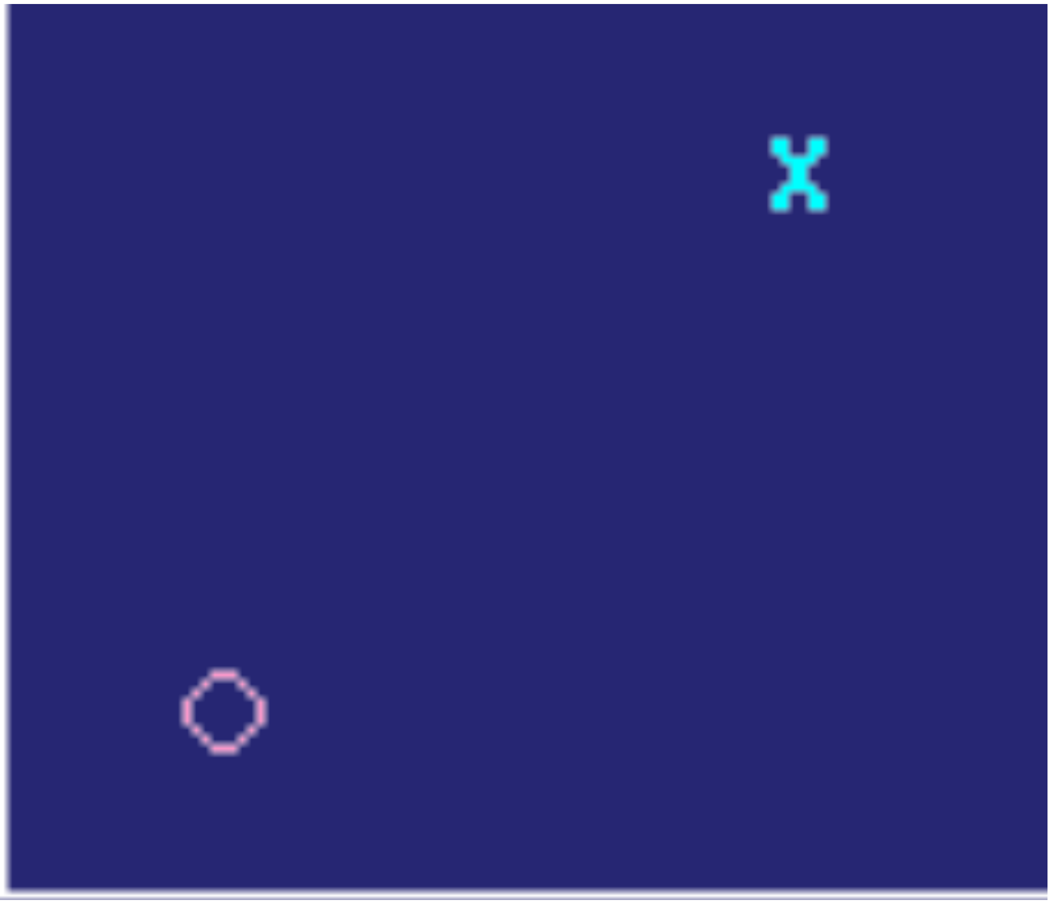


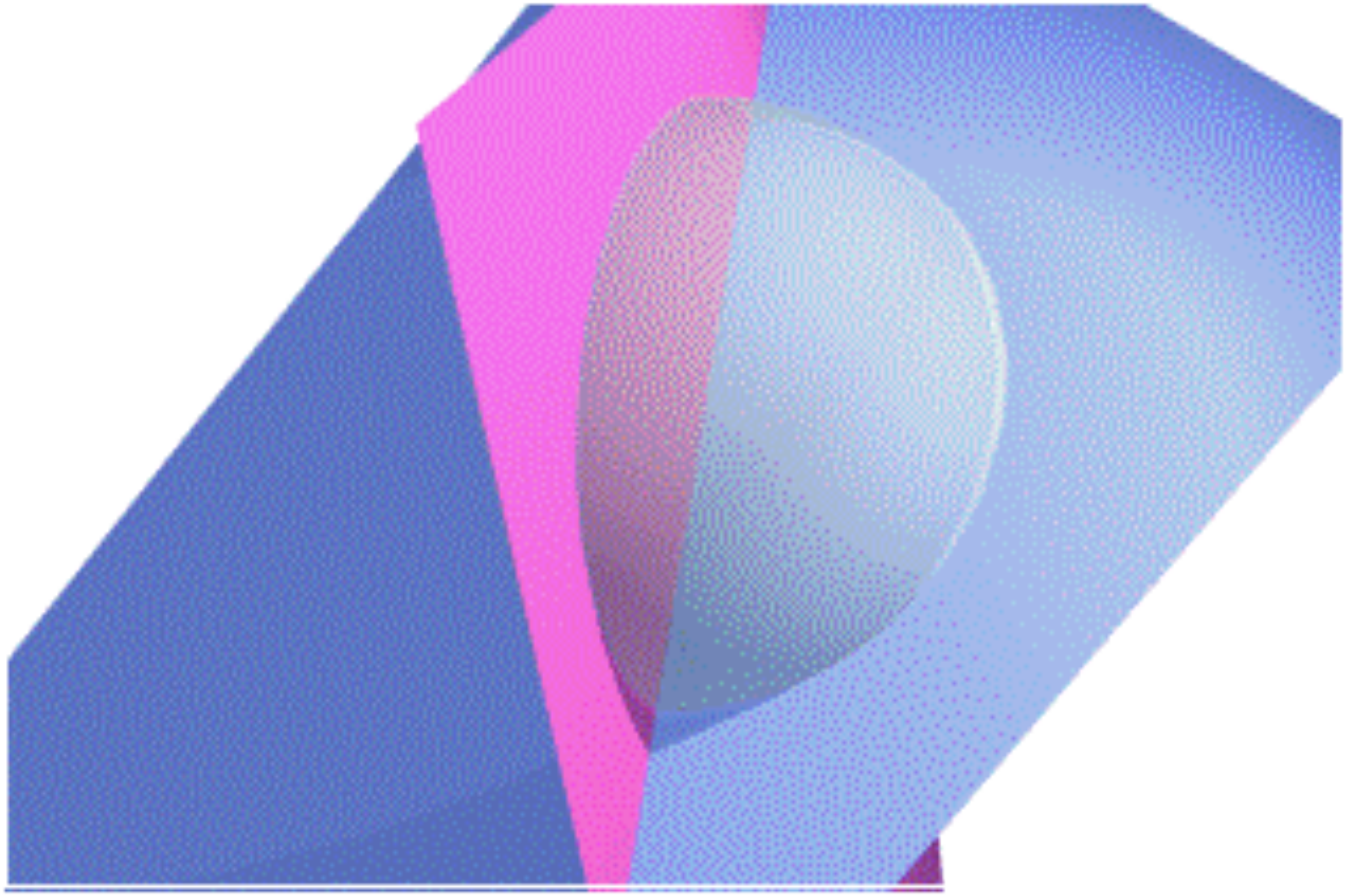
[Tong & Koller]



Version space for SVM (III)

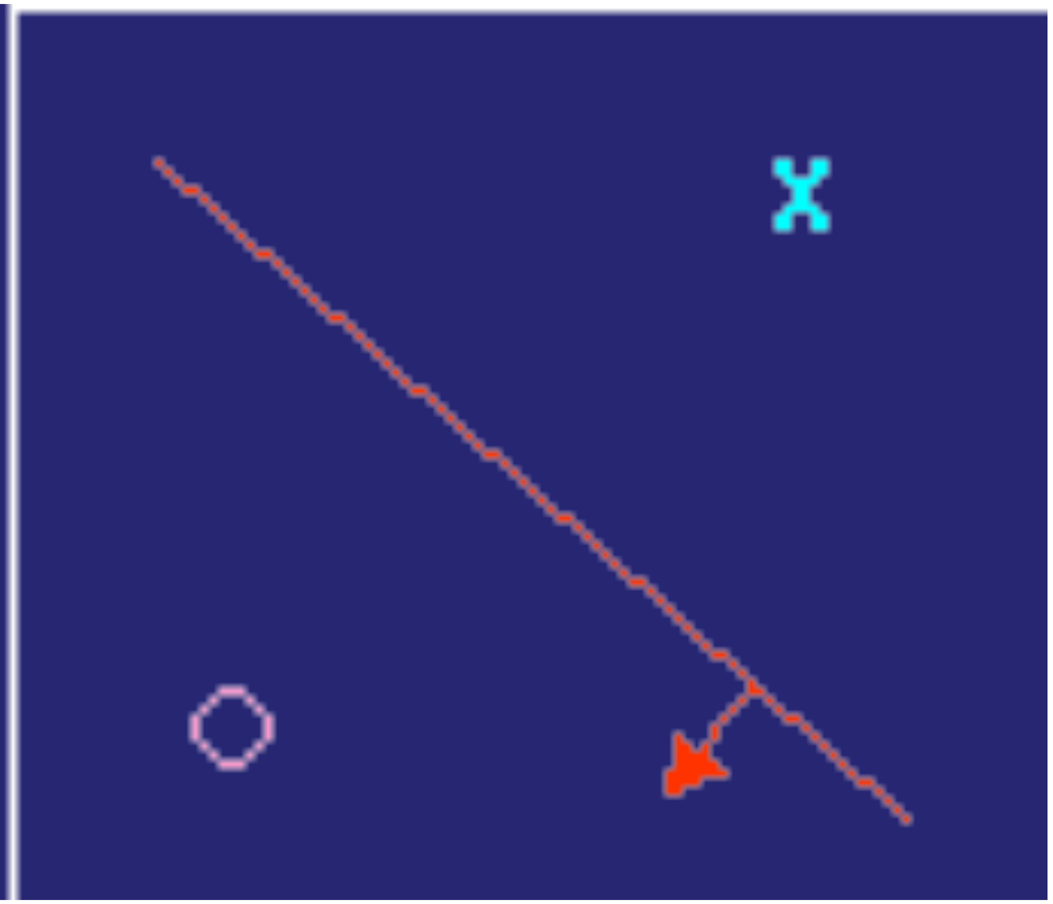
[Tong & Koller]

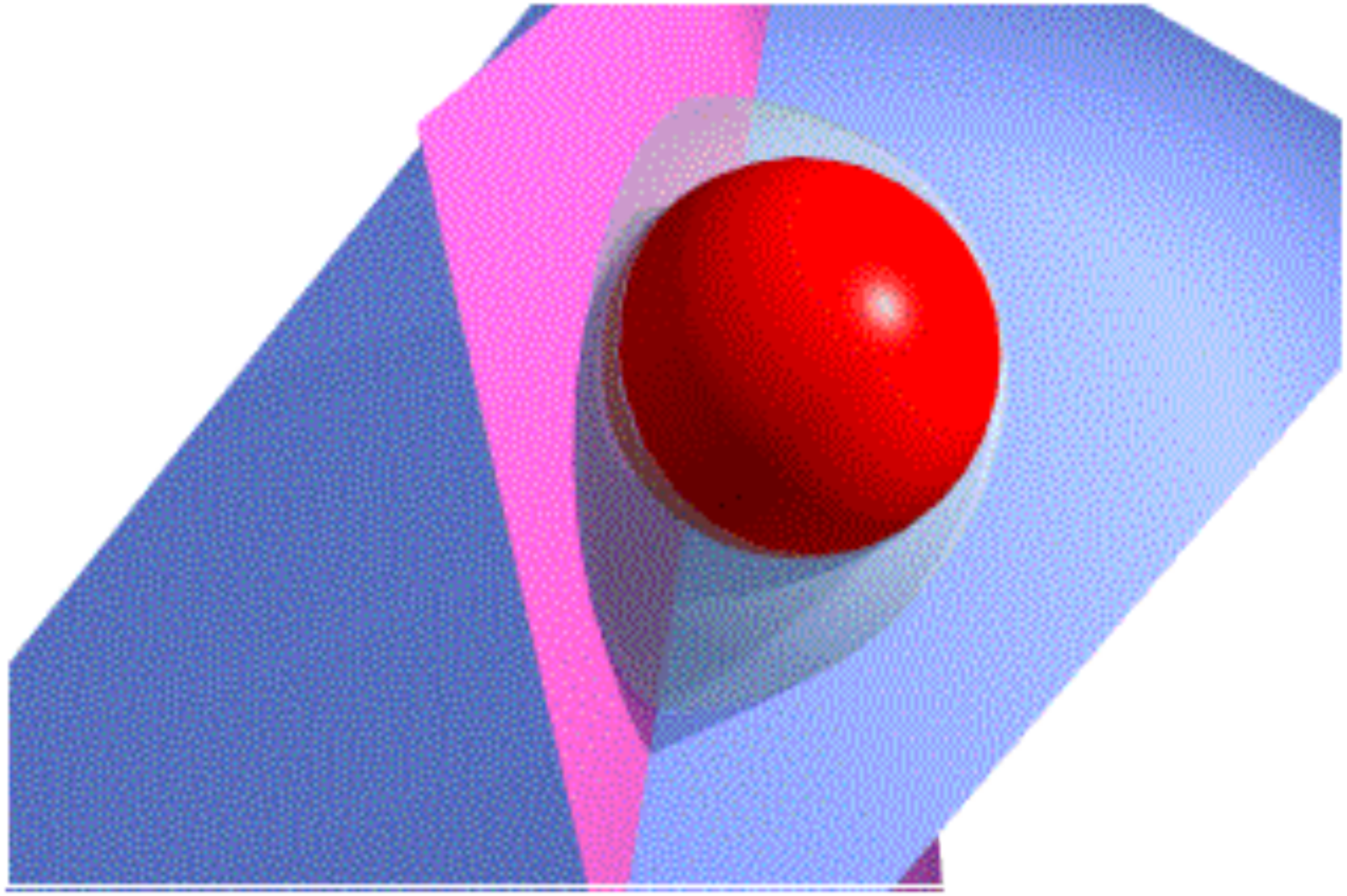




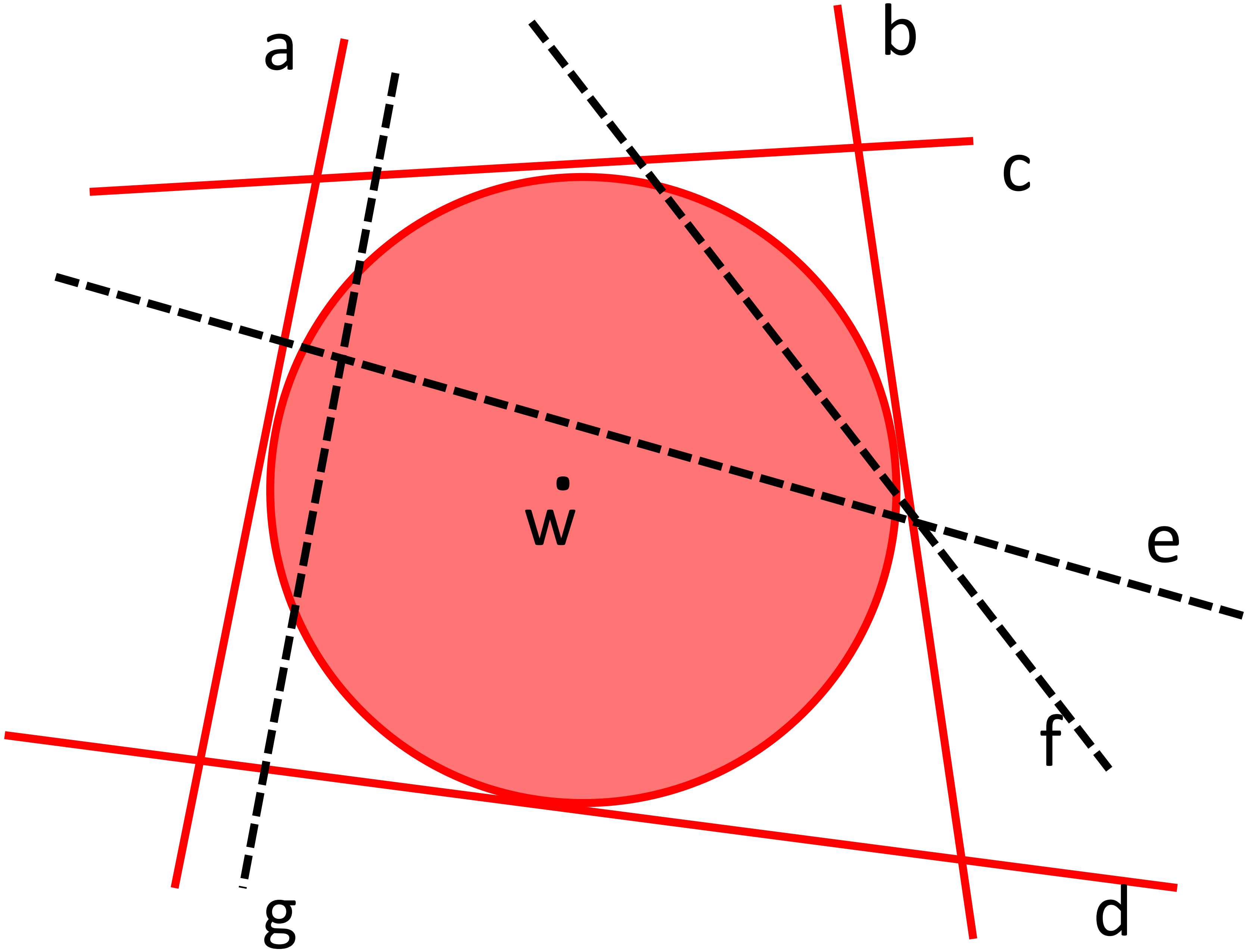
Version space for SVM (IV)

[Tong & Koller]





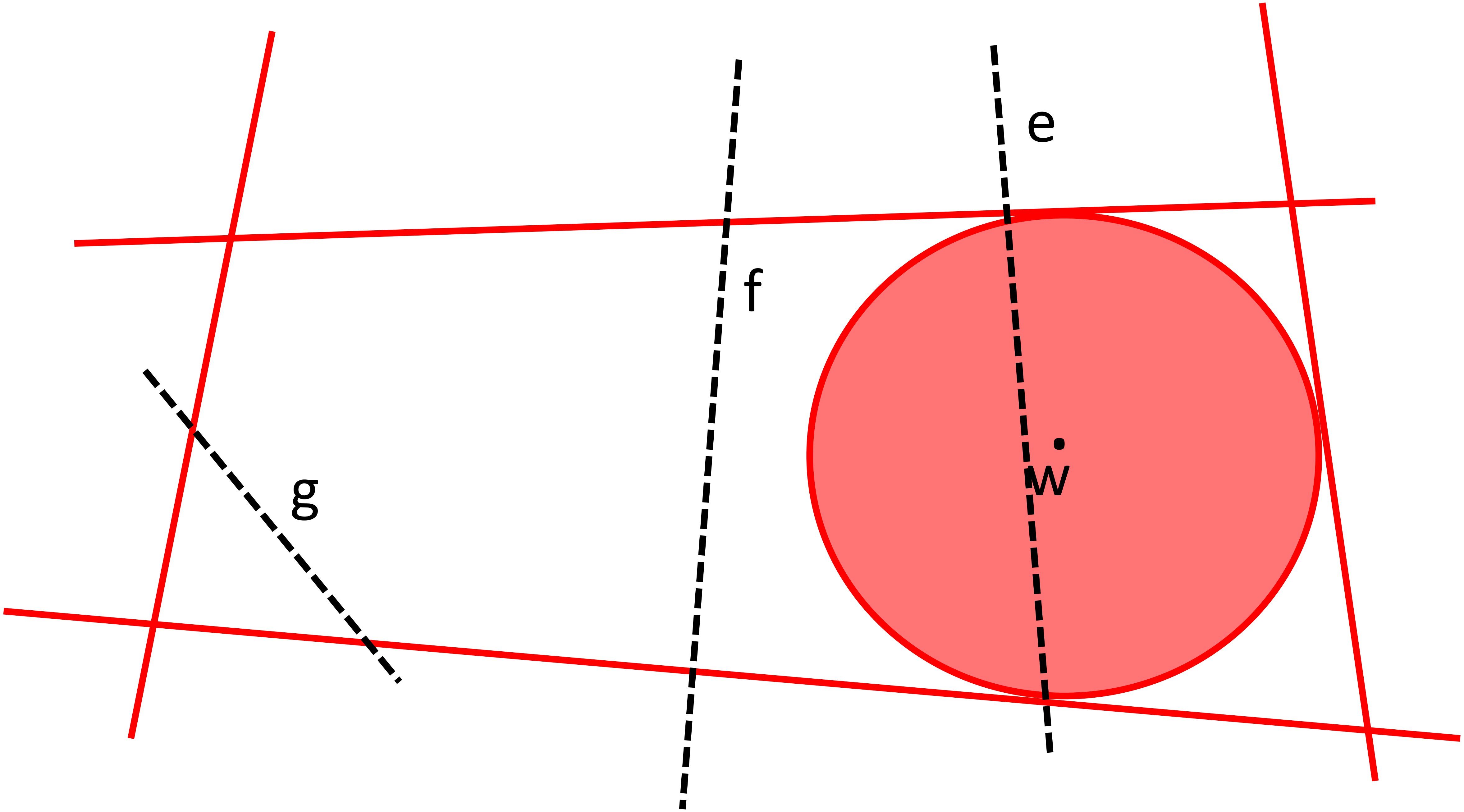
Uncertainty sampling in SVM: a version space view



[Krause]

Uncertainty sampling picks data point closest to current solution

Query selection



Uncertainty sampling picks data point closest to current solution

Quantifying “balanced” splits

Select example that splits the version space as equally as possible

In general, halving may not be possible, thus the goal is to find “balanced” split

How do we quantify how “balanced” a split is?

Version space for data set $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$:

$$\mathcal{V}(D) = \{w : \forall (x, y) \in D, \mathsf{sign}(w^\top x) = y\}$$

Relevant version space for D and unlabled pool U:

$$\widehat{\mathcal{V}}(D; U) = \{h : U \to \{+1, -1\} : \\ \exists w \in \mathcal{V}(D), \forall x \in U \ \ \mathsf{sign}(w^T x) = h(x)\}$$

- Labelings of pool consistent with the data

Generalized binary search

Algorithm: generalized binary search

```
Initialize $D = \{\}$
While $\left|\widehat{\mathcal{V}}(D;U)\right| > 1$
    ▸ For each unlabeled example $x$ in $U$ compute
          $v^+(x) = \left|\widehat{\mathcal{V}}(D \cup \{(x,+)\};U)\right| \quad v^-(x) = \left|\widehat{\mathcal{V}}(D \cup \{(x,-)\};U)\right|$
    ▸ Pick example $x$ where $\min\left(v^-(x), v^+(x)\right)$ is largest, request
      label and add to $D$
```

Near-optimality of GBS [Dasgupta '04, Golovin & Krause, '11]

GBS requires only $O\left(\log\left|\widehat{\mathcal{V}}(\{\};U)\right|\right)$ more labels than any other active learning strategy, both on average and in worst-case.

Version space reduction

Select example that splits the version space as equally as possible

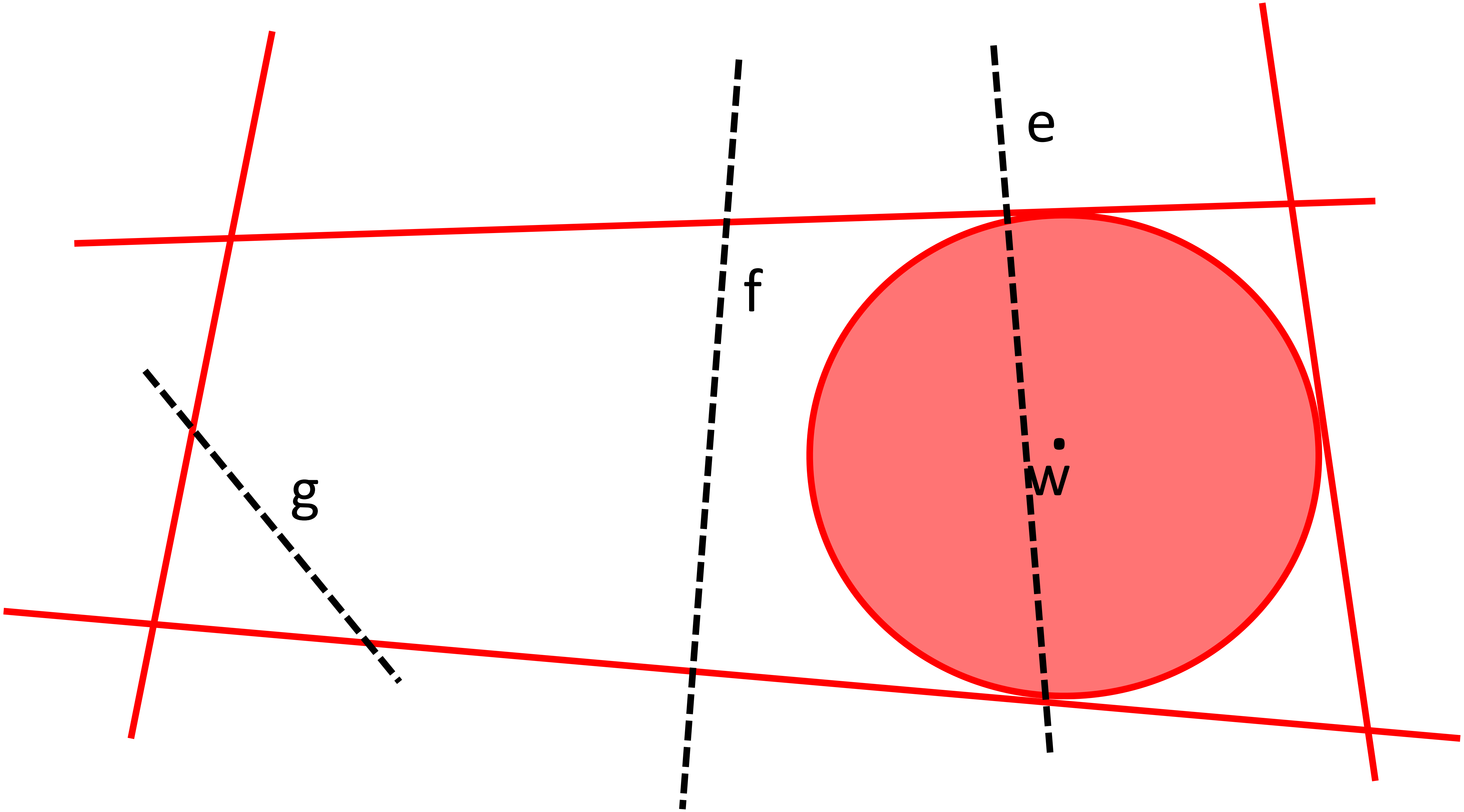
In general, halving may not be possible

- find balanced split
- generalized binary search
- competitive with optimal active learning scheme

Size of the (relevant) version space difficult to calculate

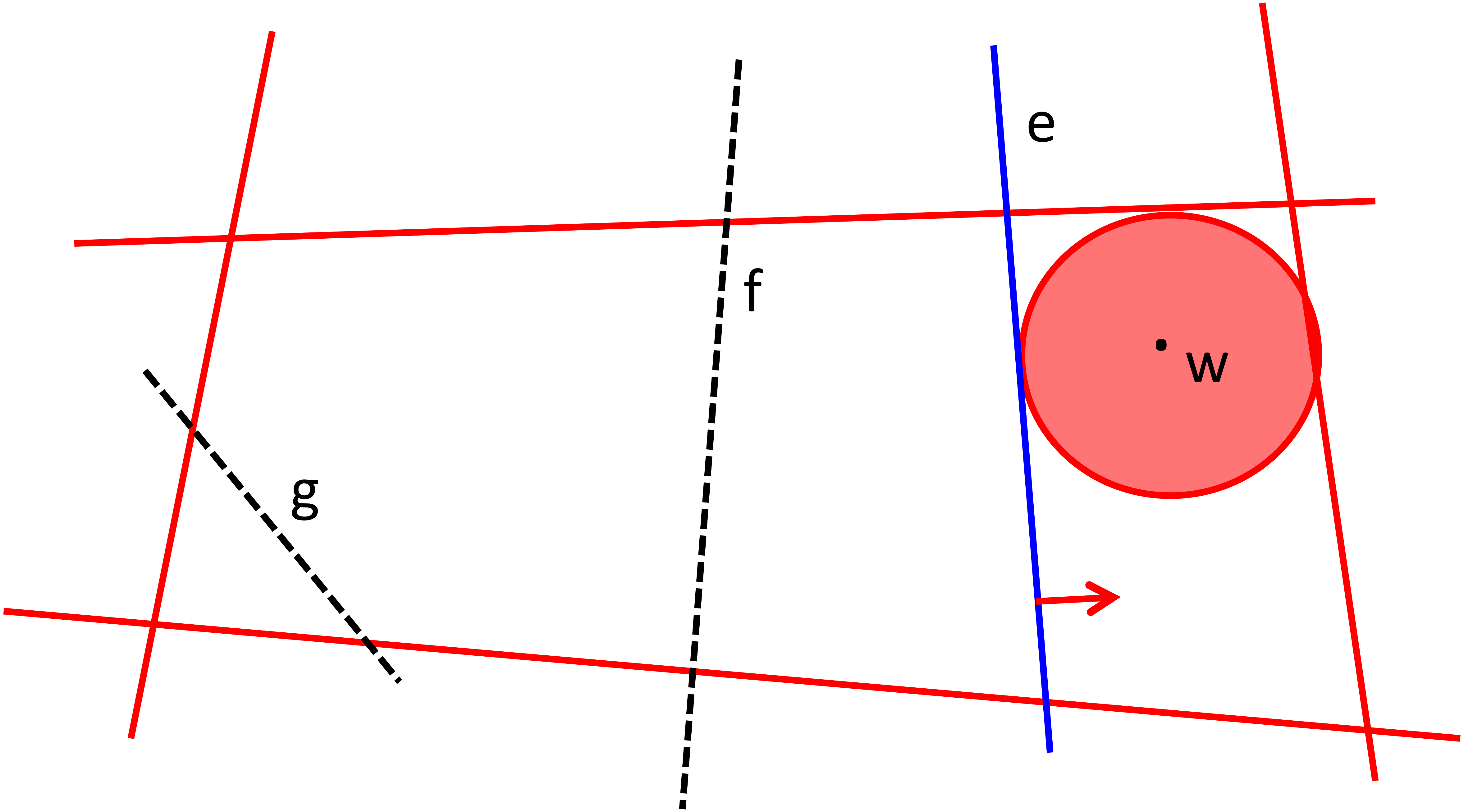
- Need approximation!

Approximation for query selection

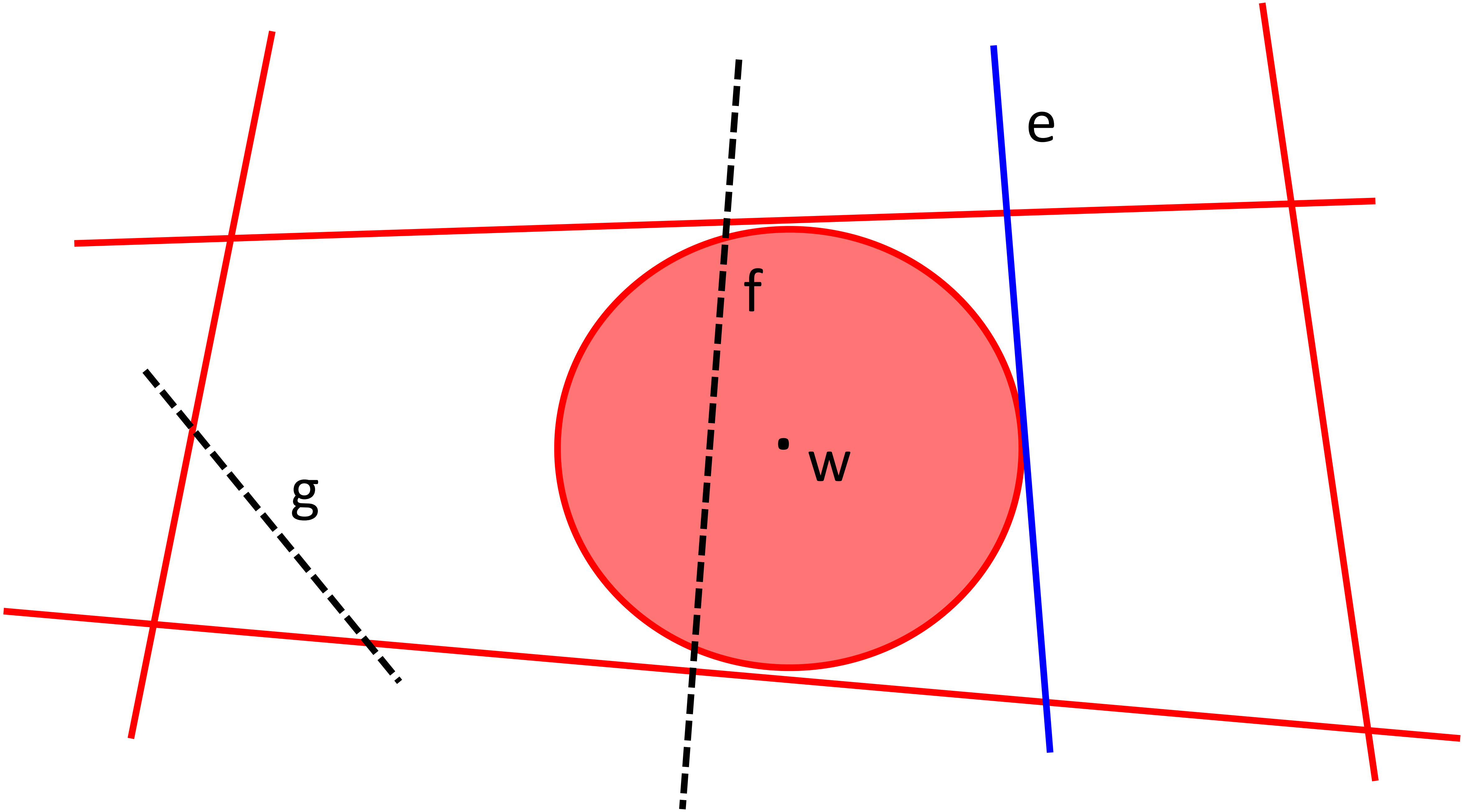


Uncertainty sampling picks data point closest to current solution

Approximation for query selection: (x,+)



Approximation for query selection: (x,-)



Suggests looking at the margins of the resulting SVMs

Query selection criteria

Suppose we're considering data point i.

- For each possible label $\{+,-\}$ calculate resulting SVMs, with margins m^+, m^-
- Define informativeness score of i depending on how balanced the resulting margins are
  - Max-min margin:

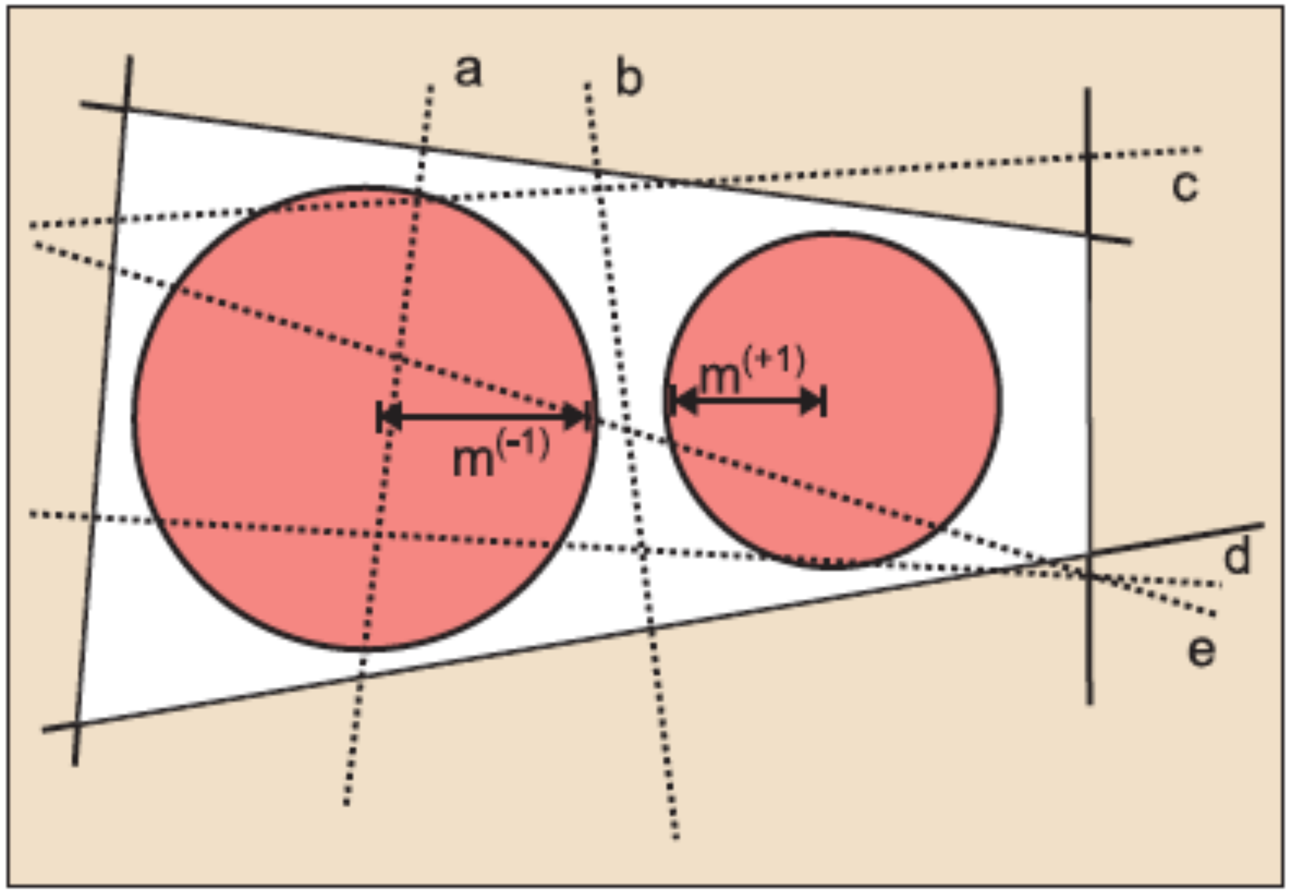
    $$\min\left(m^+, m^-\right)$$

  - Ratio margin:

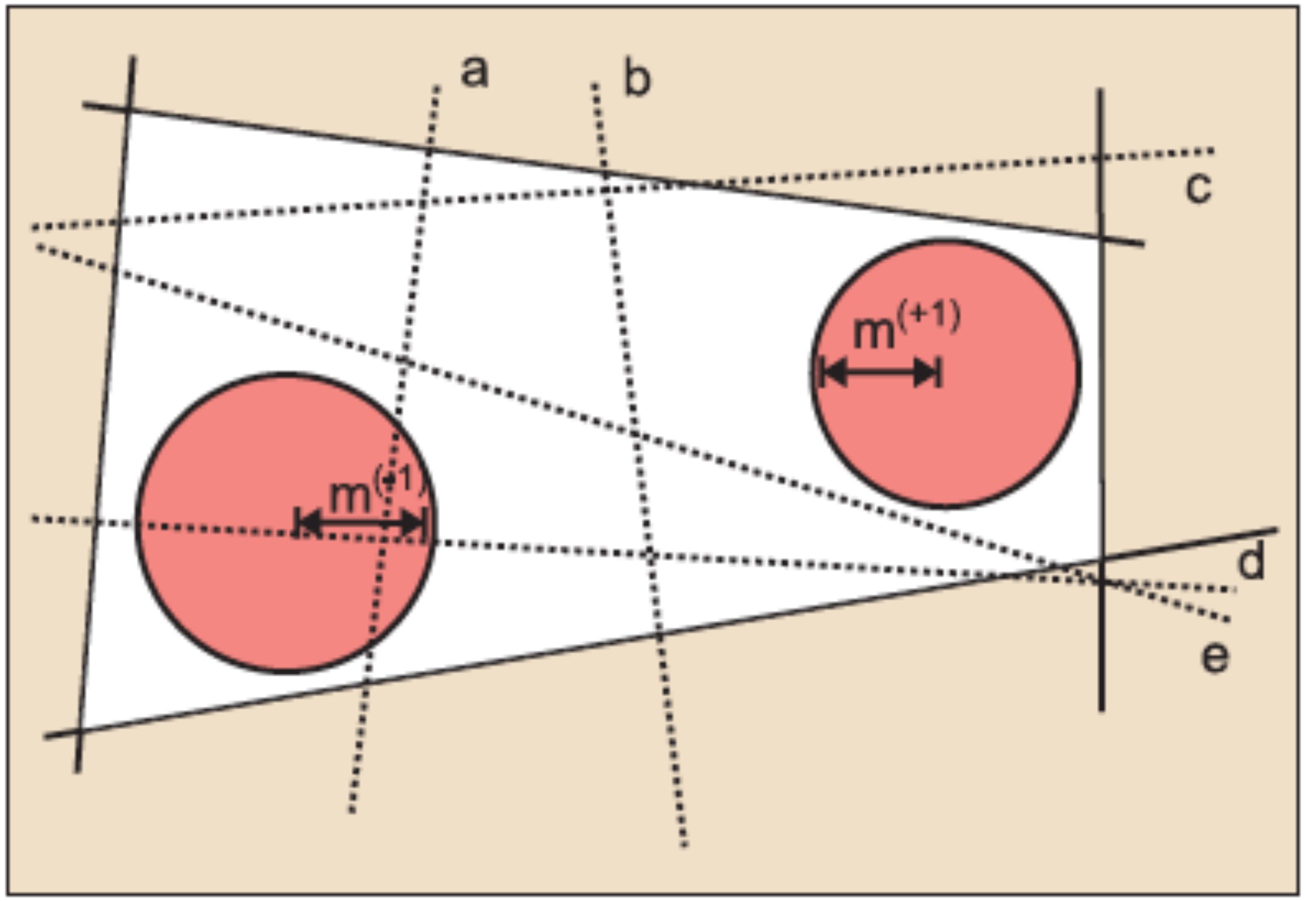
    $$\min\left(\frac{m^+}{m^-}, \frac{m^-}{m^+}\right)$$

- Select example that splits the version space as equally as possible

Selecting balanced splits



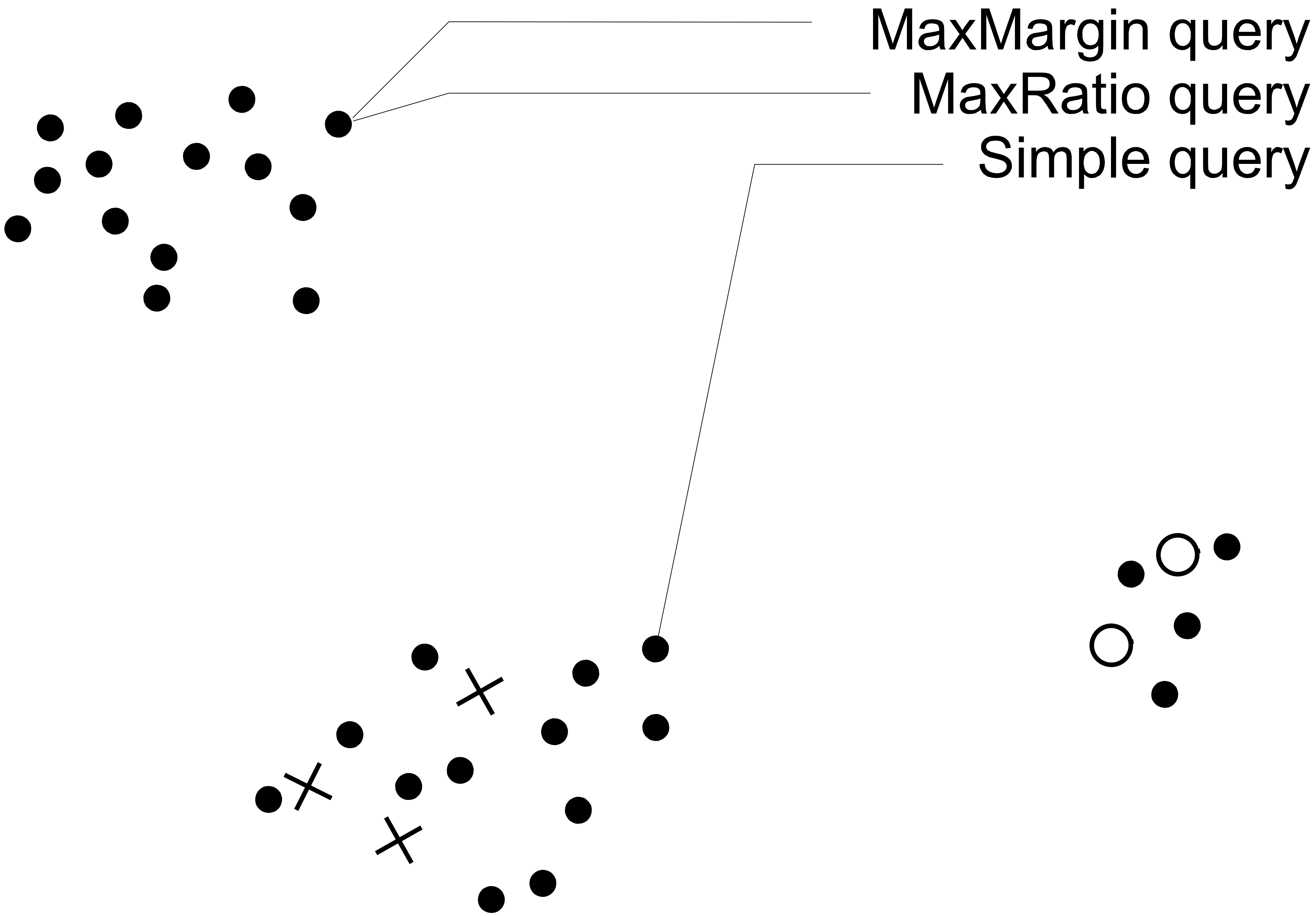
Max-min margin



Ratio margin

Example: selecting informative examples

[Tong & Koller]



Computing challenges

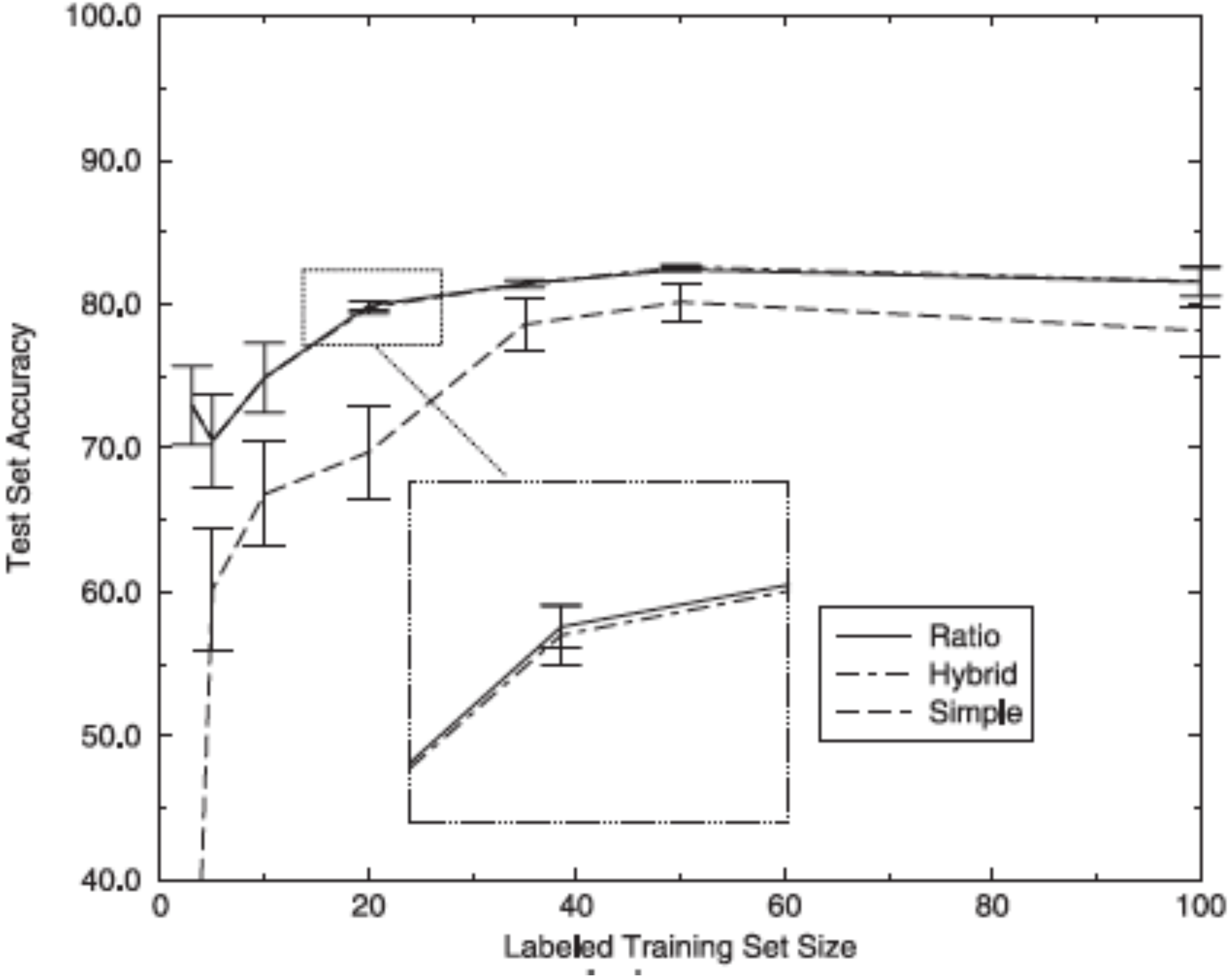
Max-min margin and ratio margin more expensive

- Need to train an SVM for each data point, for each label

Practical tricks

- Only score and pick from small random subsample of data
- Only use informativeness criterion for the first few examples, then switch to uncertainty sampling
- Occasionally pick points uniformly at random

Results (text classification)



Summary

Active learning

- Data arrives in either as a stream, or a pool
- Labels are not available by default; must decide which data to query
- Goal is to actively find training examples that are "most useful"

Pool-based active learning

- Uncertainty samppling: efficient, but can fail
- Informative sampling: expensive but effective (reduce version space)
- Computational tricks to speed up uncertainty/informative sampling

Reading materials & acknowledgment

- B Settles, Active Learning, Springer, 2009 (tech report: `https://research.cs.wisc.edu/techreports/2009/TR1648.pdf`)
- Informative sampling based on materials from Andreas Krause.