Active Learning in the Drug Discovery Process

Presented by Liina Kamm
Machine Learning Course
University of Tartu, 2008
Active Learning in the Drug Discovery Process

Manfred K. Warmuth, Gunnar Rätsch, Michael Mathieson, Jun Liao, Christian Lemmen

Advances in Neural Information Processing Systems 14, 2001
Overview

- Problem statement
- Active learning
- Selecting batches
- VoPerc, SVM and VolEst
- Performance and evaluation
The Problem Statement

- Large databases
- Active compounds
- Find these compounds quickly
How to achieve this?

- Iterative approach
- Batches of unlabeled compounds
- Biological assays
- Active hits in the batch
Using Active Learning

- Iterative approach
- Learner selects a batch
- All compounds labeled positive or negative
Specific Properties

- More negative examples
- More interest in positive examples
- Each compound described by a large vector of binary shape features (e.g. 139351)
- Vectors are sparse
Long-Term Goal

- New unlabeled examples added
- New labels added, when tests completed
- Program suggests a batch for a new test
- => Mine larger data sets more quickly
Machine Learning Aspect

- Fixed set of points in $\mathbb{R}^{139351}$
- Data is unlabeled, active or inactive
- Data is linearly separable
Selection Algorithm (1)

- Batch size
- Initialization
- Selection strategy
Selection Algorithm (2)

- Batch size - 5%
- Initialization - first batch chosen randomly
- Selection strategy - based on linear classifiers of the data labeled so far
Finding the Weight Vector

- Perceptron (Perc)
- Maximum margin hyperplane (SVM)
- "Voting"
- Bouncing a billiard ball (Monte Carlo method)
Voting: VoPerc

- Multiple passes until the vector is consistent
- Store the vector
- 100 random permutations of the labeled examples
- Each vector gets one vote: + or -
- Prediction on an example positive, if total vote > 0
- Select the unlabeled examples whose total vote closest to 0
Figure 3: The number of examples selected by the random selection strategy grows linearly with the number of batches (in each random batch, we act on 1 example). The performance of Perc, VoPerc, SVM, and geometric distance is evaluated on the topology of the data set. The fraction of examples selected for a particular batch size was: 10, 50, and 100.

- Perc true pos
- Perc false pos
- VoPerc true pos
- VoPerc false pos
- SVM false pos
- SVM true pos

The graph shows the number of examples selected versus the fraction of examples selected.
SVM

- Shrink the geometric margin of the support vectors
- Pick unlabeled examples that cause the margin to shrink the most
- Pick examples closest to the hyperplane
A Bit of Geometry

When $wx > 0$, then $x$ lies on the positive side of $w$

In a dual view, $w$ lies on the positive side of $x$

A weight vector $w$, consistent with all labeled examples $(x_n, y_n)$ must lie on the $y_n$ side of the plane $x_n$
Bouncing a billiard: VolEst

Version space - set of all consistent weight vectors

A billiard is bounced 1000 times in the version space

Fraction $f_n$ of bounce points on the positive side of $x_n$ is computed

Positive if $f_n$ is larger than half

Select unlabeled points whose fraction is closest to half
Figure 3: The number of examples in the best three strategies as a function of the fraction of examples selected. The number of examples after the first round is about 50. This number grows linearly with the number of batches (in each random batch the best three strategies are selected).
Performance

- All three strategies perform much better than those that select randomly.
- Proposed methods particularly suitable when few positive examples.
SVM vs SVM-rand
Evaluation Criterion

- Total number of positives among all examples tested
- Linear in the random selection strategy
Total Hit Performance

The figure shows the total number of hits for different methods as a function of the fraction of examples selected. The methods compared are VoPerc, SVM, and VolEst. The graphs illustrate how the methods perform in terms of hit count when selecting examples from a dataset, with the highest hits achieved by VoPerc, followed by SVM and VolEst. The performance metrics are important for evaluating the effectiveness of these methods in real-world applications.
Changing the Strategy

- Positive hits are more valuable
- Select unlabeled data with largest
  - Positive distance to the maximum margin hyperplane \((\text{SVM}^+)\)
  - Vote \((\text{VoPerc}^+)\)
  - Largest fraction \(f_n\) \((\text{VolEst}^+)\)
As the finding of we to reasonable option Vmum time if the prod

Figure 7: fraction of examples selected vs. total number of hits VoPerc, SVM, VolEst

Figure 6: fraction of examples selected vs. total number of hits VoPerc, SVM, VolEst

Strategy+
Finding a reasonable estimation of the time we spend to pick examples closest to the maximum margin hyperplane since these examples are expected to change the maximum margin.

**Total Hit Performance**

![Graph showing Total Number of Hits vs. Fraction of Examples Selected]

- **VoPerc**
- **SVM**
- **VolEst**
Effect of Batch Size

Batch size of 5% not much worse than batch size of 1 example

If the latter gave better results, more sophisticated selection strategies would be worth exploring
The described selection strategies do much better than choosing random batches.

VoPerc seems to be the simplest and the most adaptable.

No rigorous justification for the \(^*\) version of the algorithms.

Active learning ideally fits the drug design cycle.
Thank You!