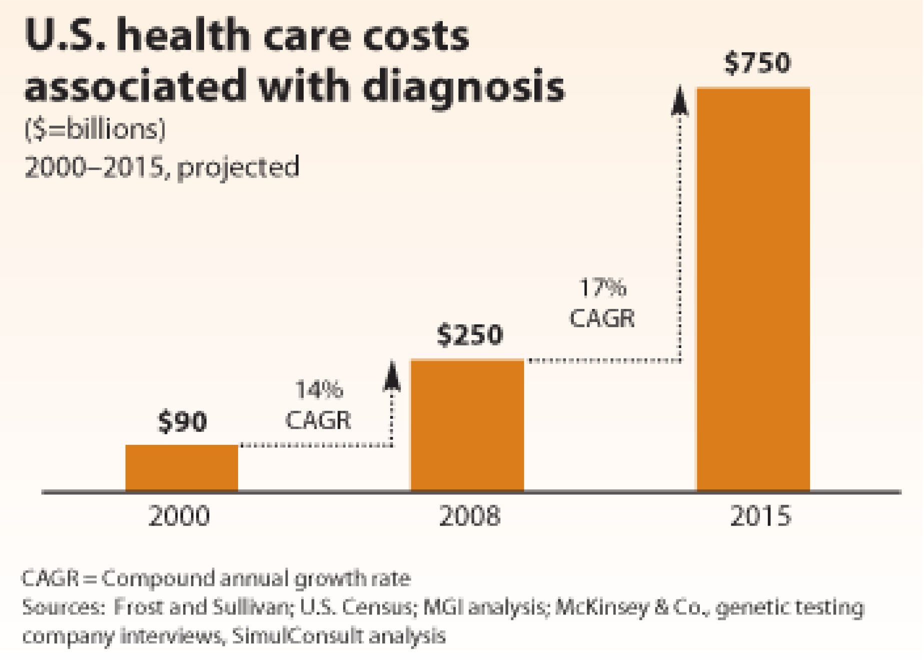
Optimal Decision Tree with Noisy Outcomes

Su Jia, Fatemeh Navidi, Viswanath Nagarajan, R. Ravi

Medical Diagnosis

- Diseases with probability of happening.
- Tests with costs.
- Each test has +/- outcome for each disease.
- We want to identify the disease of the patient by taking tests one by one, and observing each test outcome before choosing the next one.

Goal: minimizing expected cost of diagnosis.



https://www.managedcaremag.com/archives/2009/5/managing-cost-diagnosis

Optimal Decision Tree

- Hypotheses with a distribution, based on which one of them (i*) has happened.
- Decisions with costs and +/- outcomes on hypotheses.
- We make decisions one by one, and observe the feedback, before the next one.

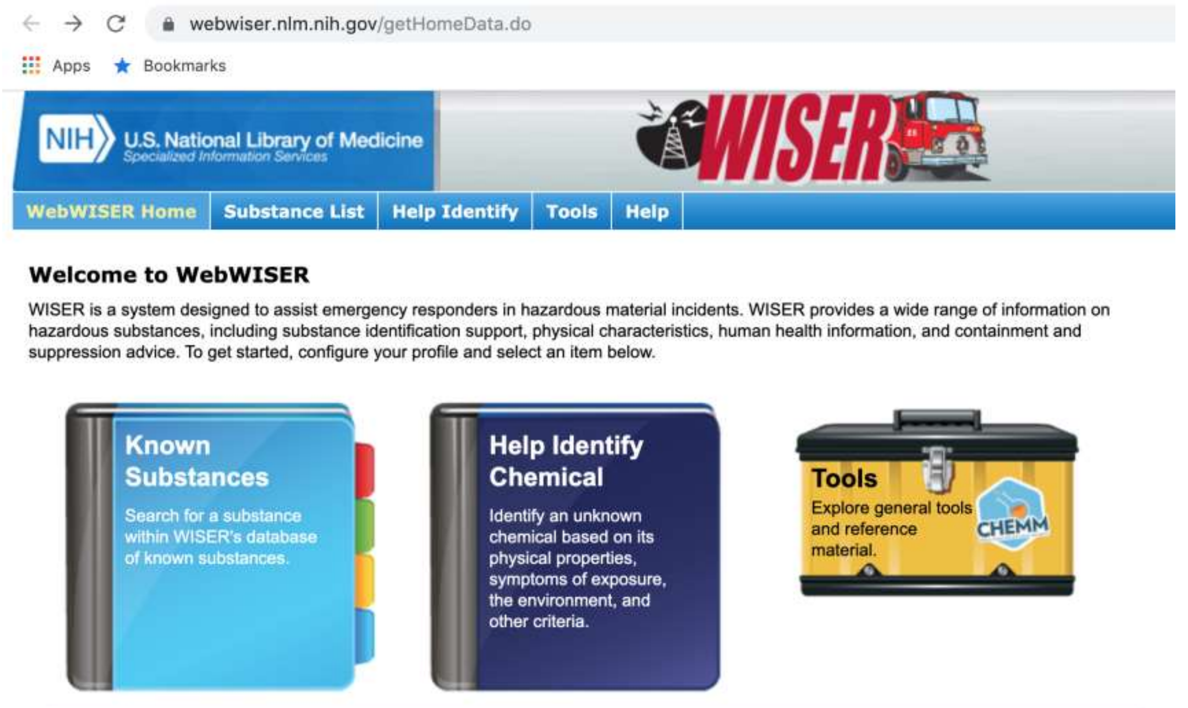
Goal: minimizing expected cost of identifying the hypothesis that has happened.

Toxic Chemicals Identification

- Missing Data
- Device Errors
- Inconsistent behaviors

hyp. \ test	1	2	3	4
1	+	+	*	+
2	-	-	*	*
3	*	+	-	-

* specifies an unknown outcome.



Noise Model

- We can model unknown outcomes to be + or - with probability ½ each.
- Extension to other probabilities
- Persistent Noise

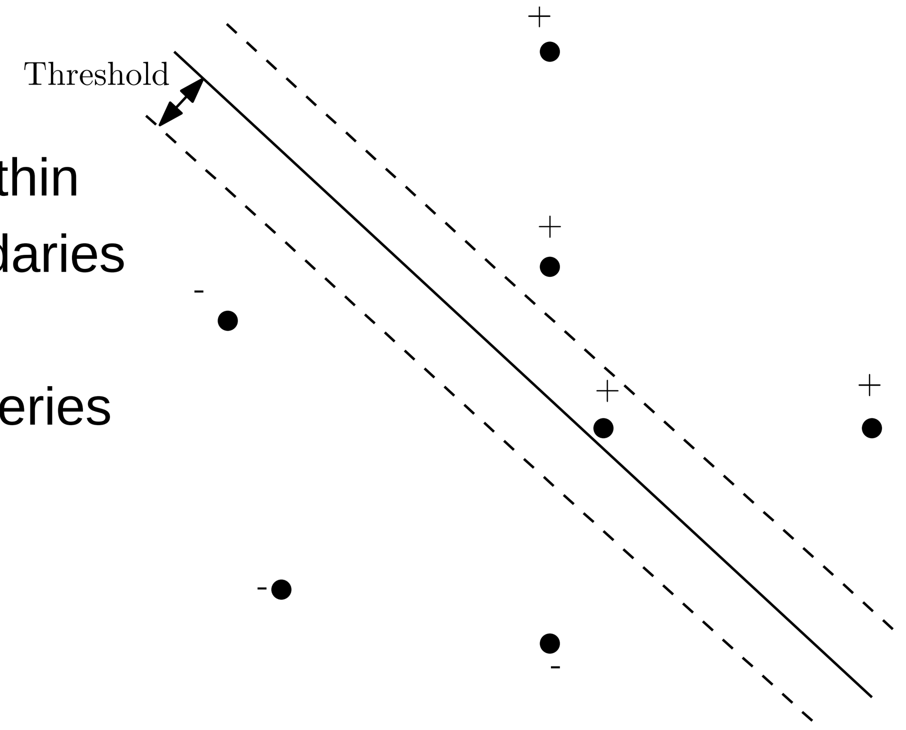
Example: In table above if i* = 2 and we run test 3, then we observe + w.p. ½ or - w.p. ½. While we always observe - if we run test 2.

Adaptive vs Non-Adaptive

- In adaptive model we observe the feedback after each test.
- In non-adaptive model:
  - ➢ No observed feedback
  - ➢ The same sequence for every chosen hypothesis
  - ➢ Can be used for batch-mode testing
  - ➢ No real-time processing time

Bayesian Active Learning

- A set of data points, each has an unknown label
- A set of linear classifiers, under each the data points have specific labels
- One classifier has happened based on a distribution
- We want to query labels of data points one by one until we identify the classifiers
- Noisy labels when data is within a threshold of classifiers boundaries
- Minimizing the number of queries



Our Results

- Adaptive: $O(\log m + \min(r, h))$-approximation algorithm
  - ➢ m: number of hypothesis
  - ➢ r: maximum number of unknowns for each test
  - ➢ h: maximum number of unknowns for each hypothesis
- Adaptive for sparse case: $O(\log m)$-approximation algorithm
- Non-adaptive: $O(\log m)$-approximation algorithm.
- First result that handles any number of unknown outcomes.
- Tight result for adaptive case if either r or h are $O(\log m)$, and for non-adaptive case with any number of unknowns.

Adaptive Algorithms

- Simple greedy style algorithms
- Repeatedly selecting a test that maximizes a combination of:
  - ➢ The expected number of eliminated hypotheses
  - ➢ The minimum probability of eliminated hypotheses
- Updating the set of compatible hypotheses based on observed feedback

Non-Adaptive Algorithms

The non-adaptive algorithm comes in two phases:

- ➢ In phase 1, using sampling we run an algorithm by [Azar, Gamzu'11] for Submodular Function Ranking problem on our instance, to estimate a score for each element.
- ➢ In phase 2 , we choose the test with maximum score. If it is smaller than a threshold, the approximation fails and we need to run all tests.

Experiments

- Information Theoretic Lower Bound (Entropy)
- Low Adaptive
- Our Algorithms: Non-Adaptive, ODTN-r and ODTN-h

WISER:
- ➢ 415 chemicals
- ➢ 78 tests/symptoms

Linear Classifiers:
- ➢ 234 classifiers
- ➢ 100 points