Evaluation of Classifiers (IDSs with a Watermarking Application): Practical Considerations in Security Applications

Alvaro A. Cárdenas
University of Maryland

TRUST seminar
May 2006

Introduction: how good is a classifier? (no easy answer)

- Recent use of machine learning techniques in security-related applications: Intrusion detection systems (IDSs), Biometrics, Spam filters, data hiding (watermarking) in multimedia data, fraud detection etc.

- Measure Detection: (how can we deal with unseen attacks?)

- Large numbers of false alarms make IDSs difficult to maintain.

  - Unit of analysis problem: false alarm rate depends on what you measure, the more normal instances, the smaller the false alarm rate, however the same number of alarms! vs. pseudo false alarms

  - Base rate fallacy: the practical number of false alarms also depends on the likelihood of an attack, which can be very small

  - What is a small false alarm rate? 0.01? 0.001?

Issues with measuring the performance of classifiers on data sets

- Problems with empirical evaluations (performance of IDS in a data set)
  - No standard benchmark (comparison among IDS difficult)
  - There will always be a difference between a data set and the real scenario
    - Dynamic changing environments: hard to establish “normal” profiles
    - Attack data will miss all possible attack variations, or new attacks
  - Evaluation or even training data might have hidden attacks
- How to deal with an attacker? After all, the UCI machine learning repository never tried to attack your classifier

Outline of the Talk

- Unified framework for the study of evaluation metrics
  - Problem: comparison between metrics is difficult since each metric is proposed in a different framework (information theory, decision theory, cryptography, statistics etc.)
  - Our approach: all proposed metrics are instances of the multi criteria optimization problem where the Pareto surface are the ROC curves. Therefore we can compare several metrics in a unified manner. We also introduce new metric: B-ROC curves (a.k.a. IDOC curves).
- Towards secure evaluation
  - Need to include the resistance against attacks as part of an empirical evaluation of an IDS.

Notation and Definitions

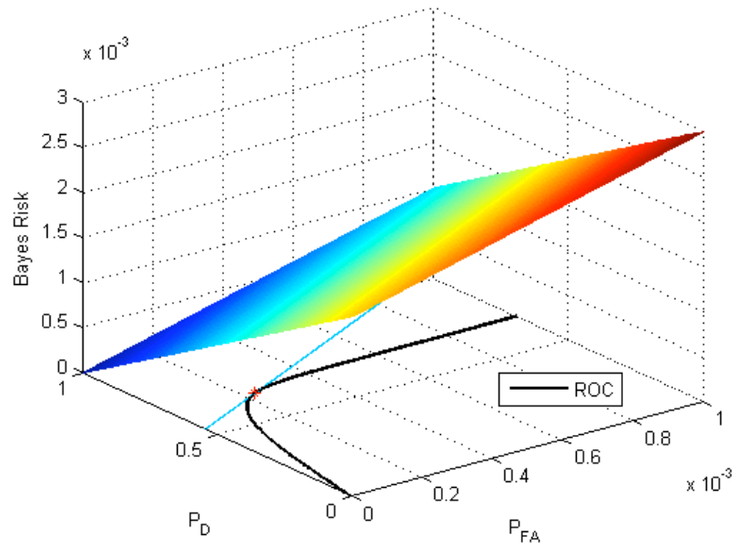
- Input to classifier **x**
- If **x** is generated by an intrusion *I*=1, otherwise *I*=0
- Given **x**, the output of a classifier is A=1 (alarm), otherwise A=0 (no alarm)
- The most basic metrics are: $P_{FA}=Pr[A=1|I=0]$ and $P_D=[A=1|I=1]$. The ROC curve shows points (P_{FA},P_D)
- The Base-rate Fallacy: even with "traditionally good" points in the ROC such as (0.01,1), if the likelihood of attack is very small, e.g., $p=10^{-5}$ then the positive predictive value: $PPV=Pr[I=1|A=1]=0.000999$

Previously Proposed Metrics

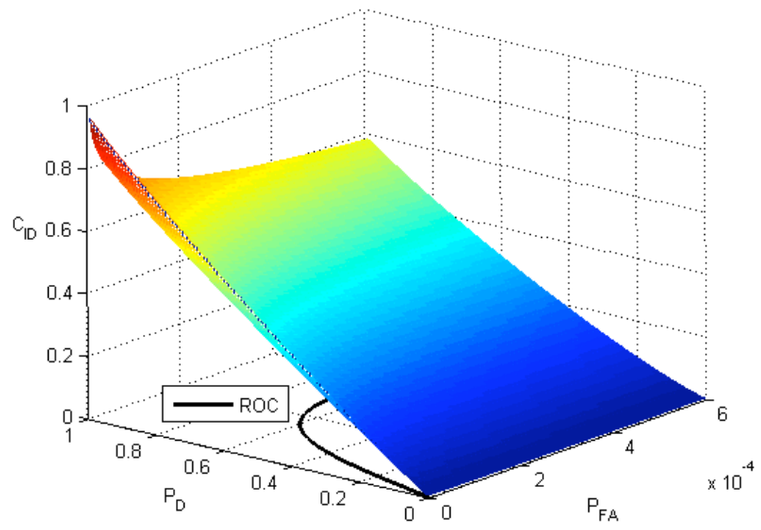
Metric	Field	Advantages	Disadvantages
ROC	Signal Processing	No base rate p or costs assumed: least # of assumptions	Evaluation depends on more factors than those considered in ROC
Cost sensitive eval. (Bayes risk) $\mathbf{E}[C(I,A)]$	Decision Theory	Single Metric Flexible	Need to know misclassification costs C(I,A) and the base rate
$C_{ID} = \frac{\mathbf{I}(I;A)}{\mathbf{H}(I)}$	Information Theory	No costs assumed a priori	No practical intuition Needs to know the base rate
Bayesian Detection Rate Pr[I|A]=PPV	Statistics	Good metric for evaluating the practical number of false alarms	Maximized when detection rate is zero.
Sensitivity Distinguishability	Cryptography	No base rate or costs assumed	Does not work well for very small values of the base rate.

Unified Framework: Multi-criteria Optimization (Pareto front=ROC Curves)

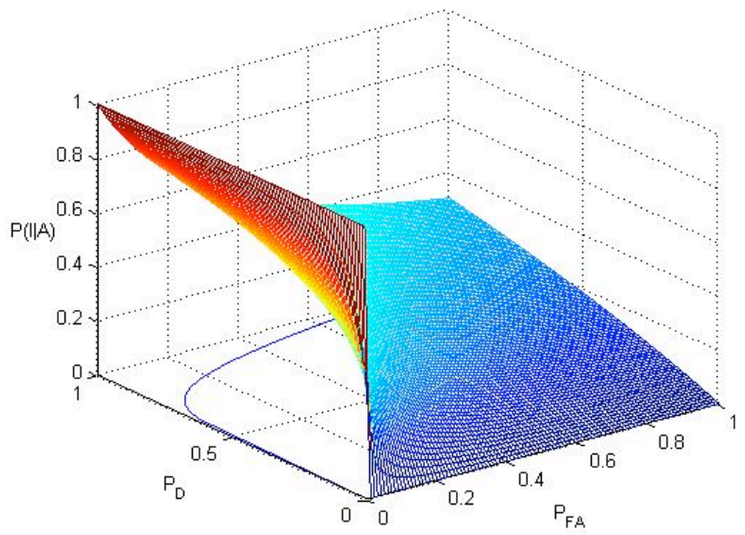
$\mathbf{E}[C(I,A)]$



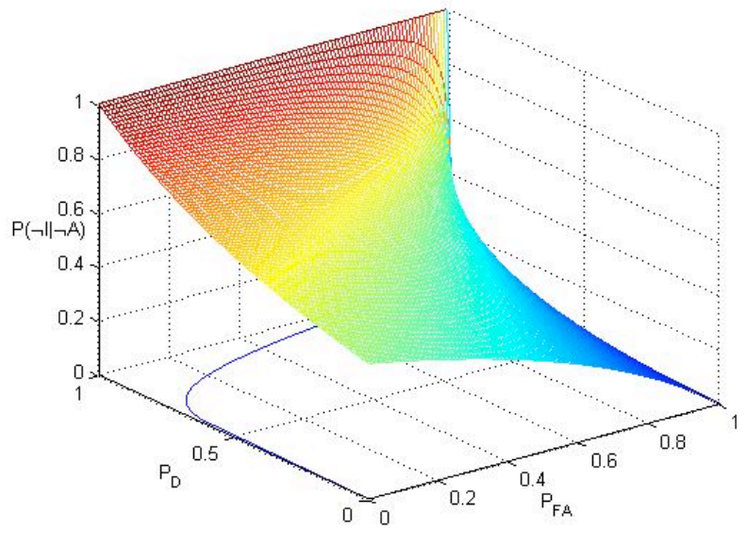
C_{ID}



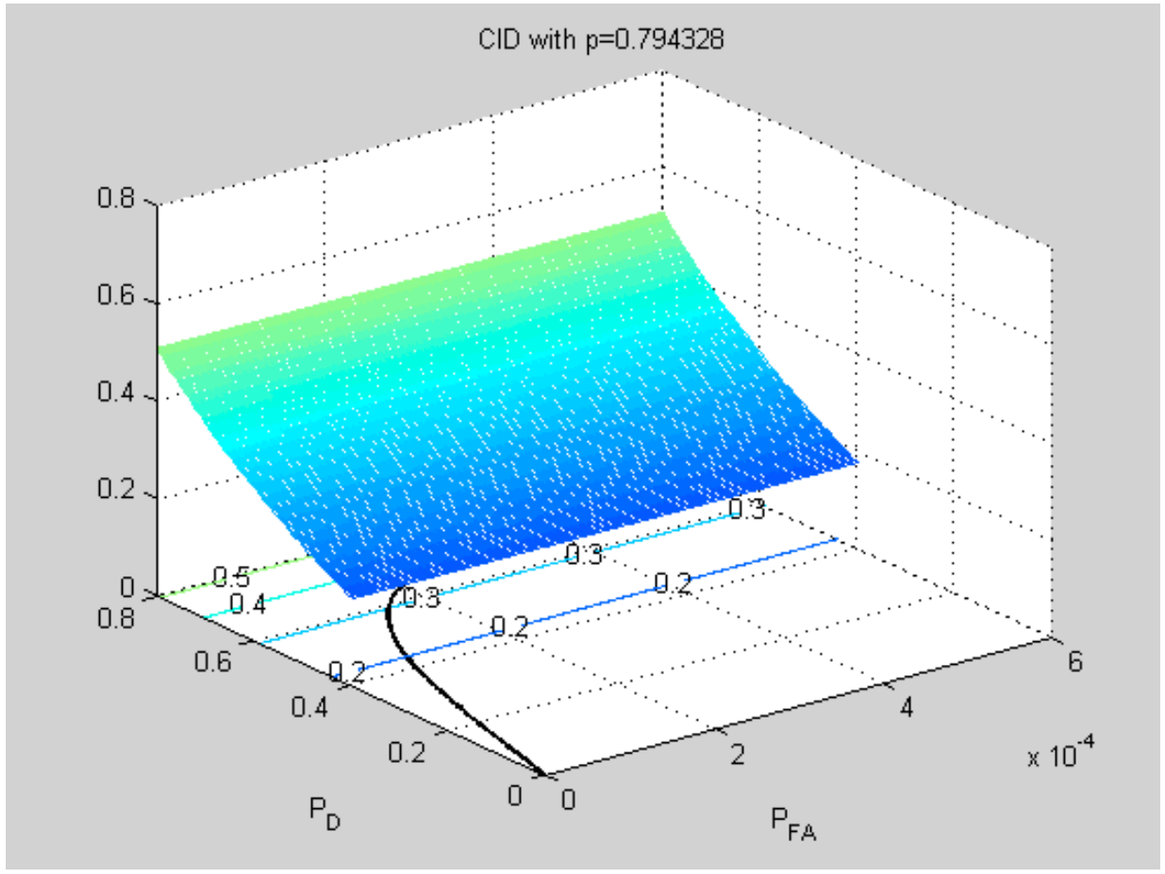
PPV=$\Pr[I|A]$



NPV=$\Pr[\neg I|\neg A]$



C_{ID}

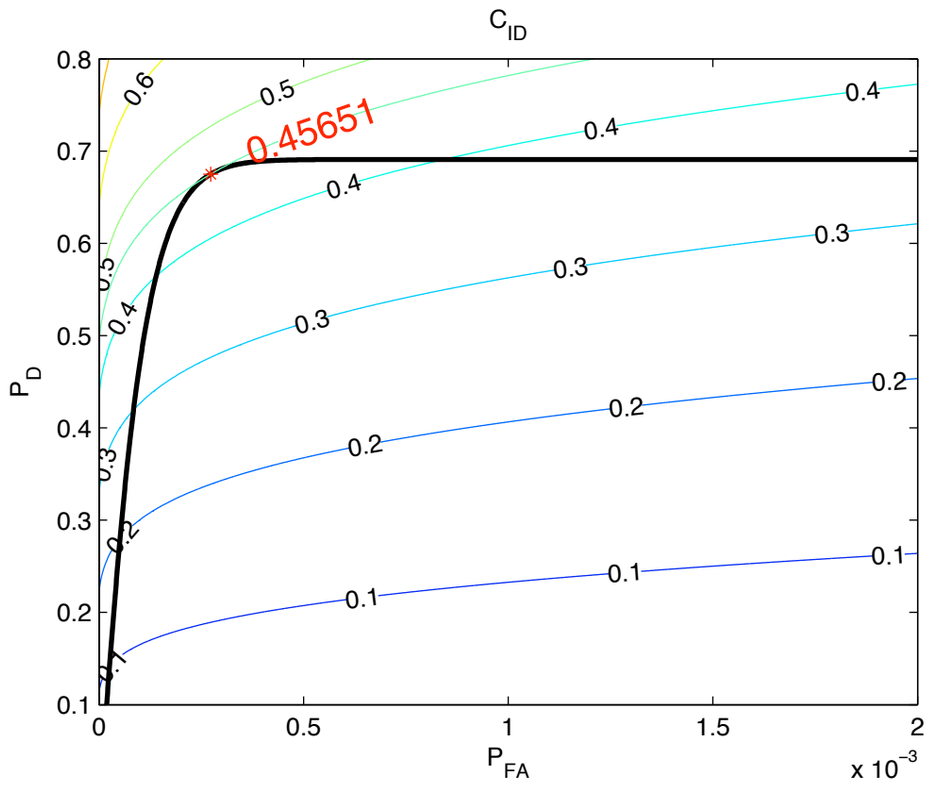


CID can be seen as an expected cost metric:

$$
\begin{aligned}
(P_{FA}^*, P_D^*) &= \arg \max_{(P_{FA},P_D)\in ROC} \frac{\mathbf{I}(I;A)}{\mathbf{H}(I)} \\
&= \arg \max_{(P_{FA},P_D)\in ROC} \mathbf{I}(I;A) \\
&= \arg \min_{(P_{FA},P_D)\in ROC} \mathbf{H}(I|A) \\
&= \arg \min_{(P_{FA},P_D)\in ROC} \mathbf{E}[-\log \Pr[I|A]]
\end{aligned}
$$

Optimal C_{ID}=0.4565
Associated Costs:
C(0,0)=3x10^{-5}
C(0,1)=0.2156
C(1,0)=15.52
C(1,1)=2.849

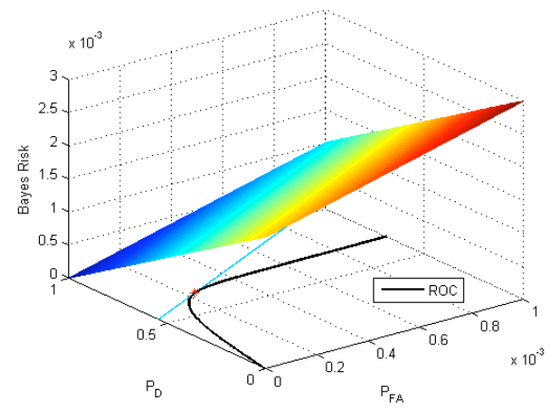
Isoline projections of C_{ID} onto the ROC curve.



Critical Slope: Cost Interpretation of the Base-Rate Fallacy

For costs independent of the base rate, the false alarm and detection rates (constant costs), the expected cost metric is characterized by the following slope:

$$m \equiv \frac{P_{D2} - P_{D1}}{P_{FA1} - P_{FA2}} = \frac{1-p}{p} \frac{C(0,1) - C(0,0)}{C(1,0) - C(1,1)}$$



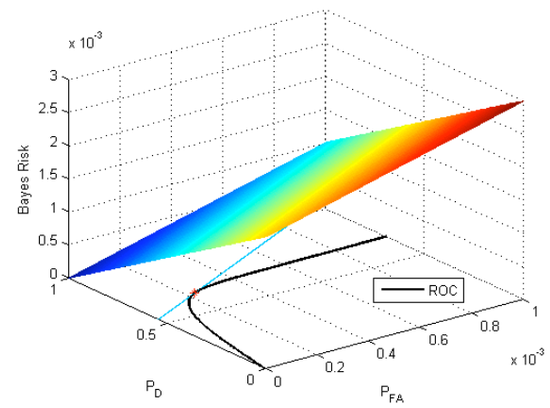
As p decreases, we tend to decide on not using the IDS

Unless C(1,0)>>C(0,1)

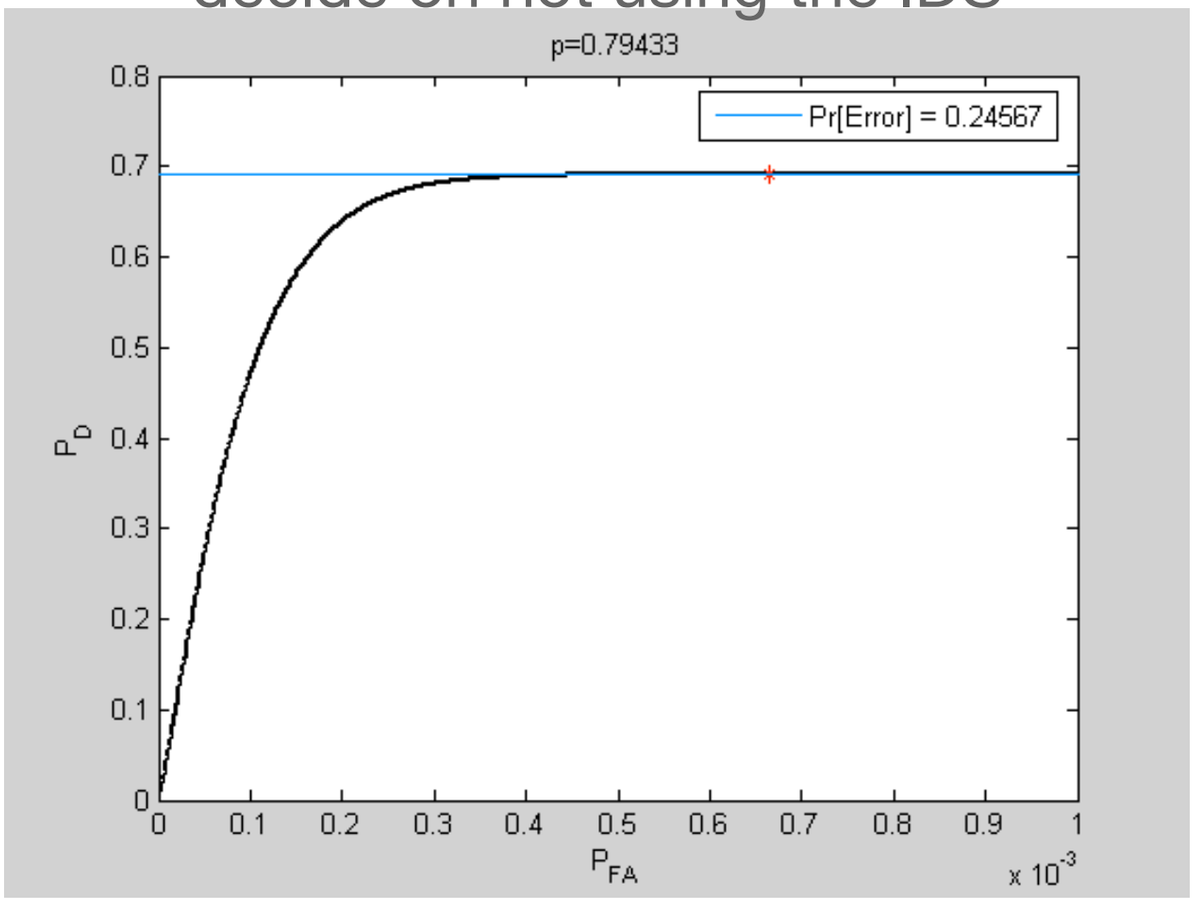
Critical Slope: Cost Interpretation of the Base-Rate Fallacy

For costs independent of the base rate, the false alarm and detection rates (constant costs), the expected cost metric is characterized by the following slope:

$$m \equiv \frac{P_{D2} - P_{D1}}{P_{FA1} - P_{FA2}} = \frac{1-p}{p} \frac{C(0,1) - C(0,0)}{C(1,0) - C(1,1)}$$



As p decreases, we tend to decide on not using the IDS

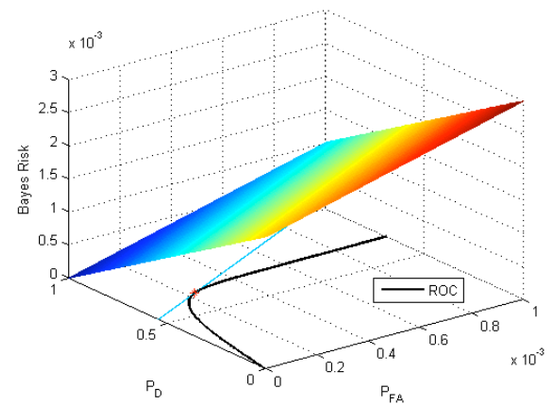


Unless C(1,0)>>C(0,1)

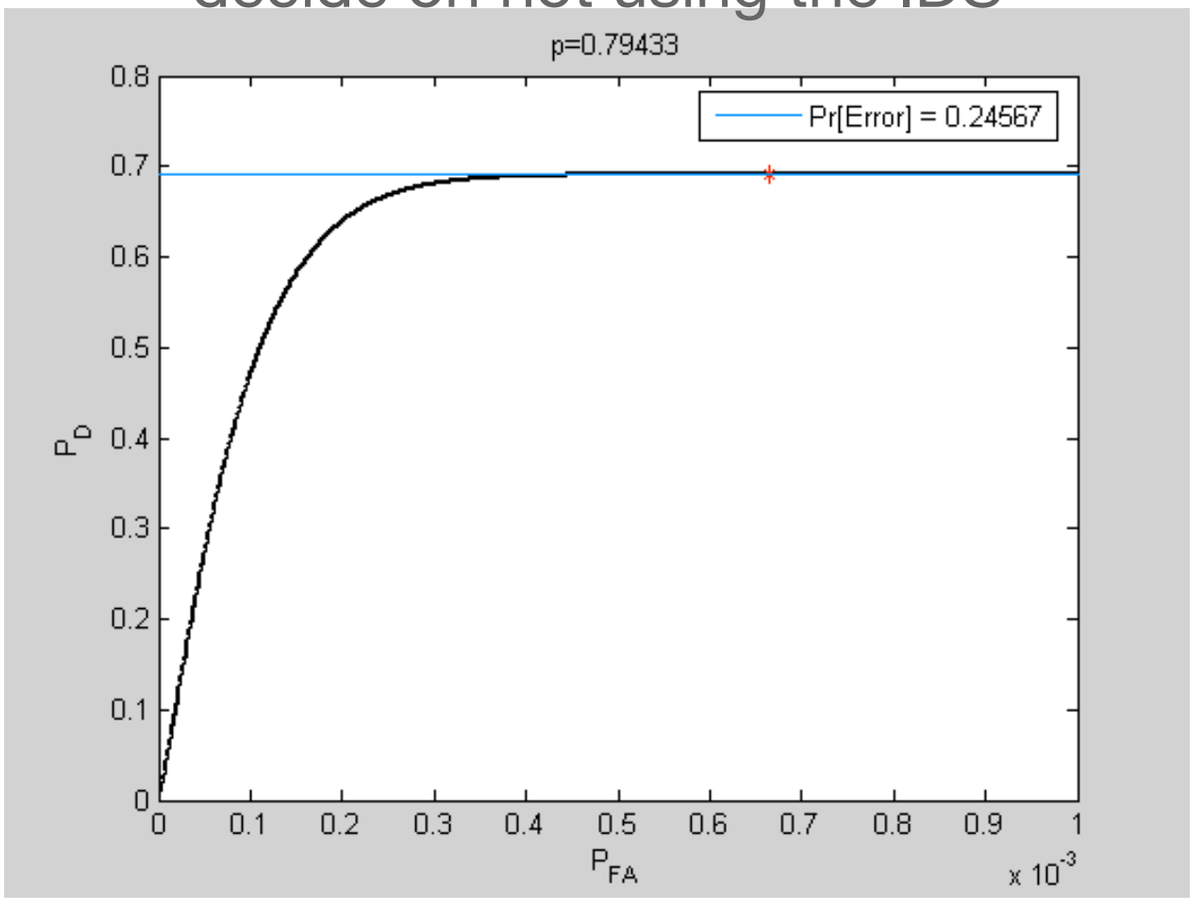
Critical Slope: Cost Interpretation of the Base-Rate Fallacy

For costs independent of the base rate, the false alarm and detection rates (constant costs), the expected cost metric is characterized by the following slope:

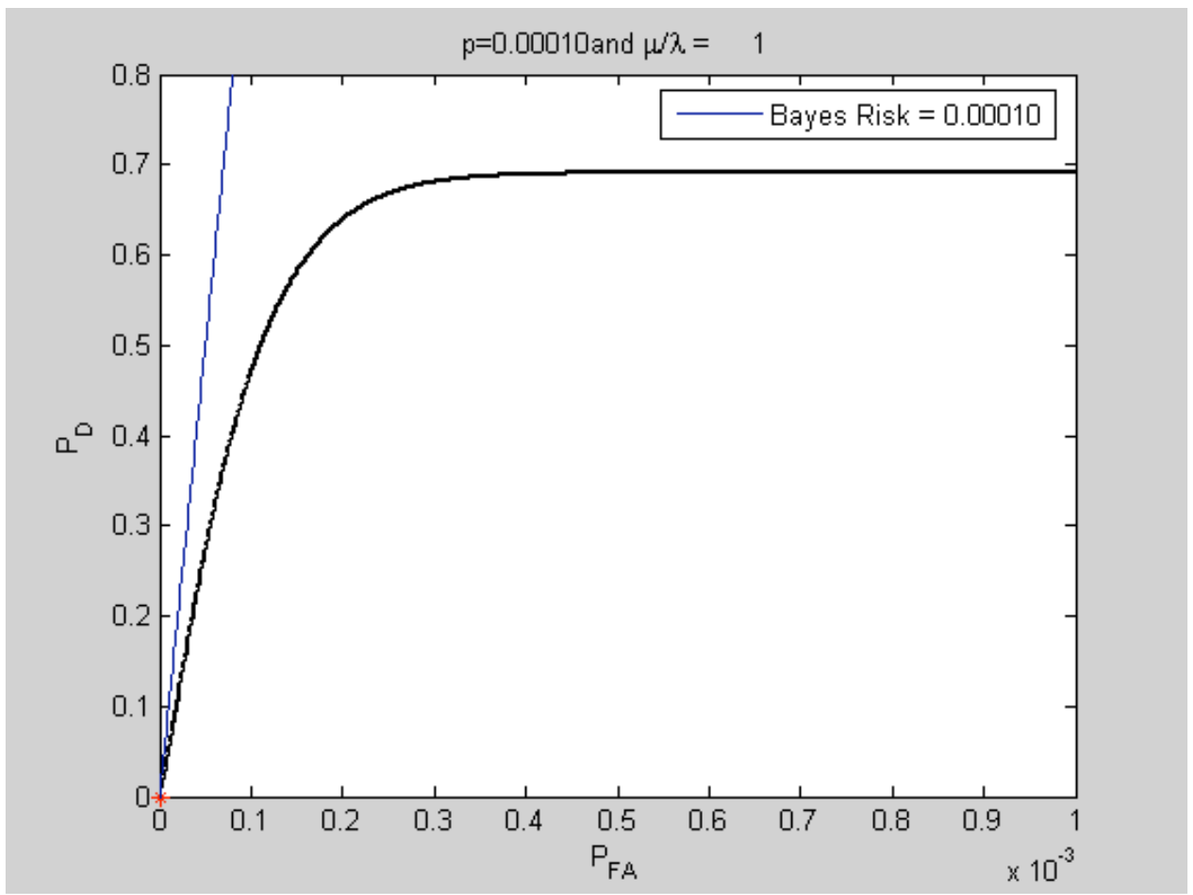
$$m \equiv \frac{P_{D2} - P_{D1}}{P_{FA1} - P_{FA2}} = \frac{1-p}{p} \frac{C(0,1) - C(0,0)}{C(1,0) - C(1,1)}$$



As p decreases, we tend to decide on not using the IDS



Unless C(1,0)>>C(0,1)

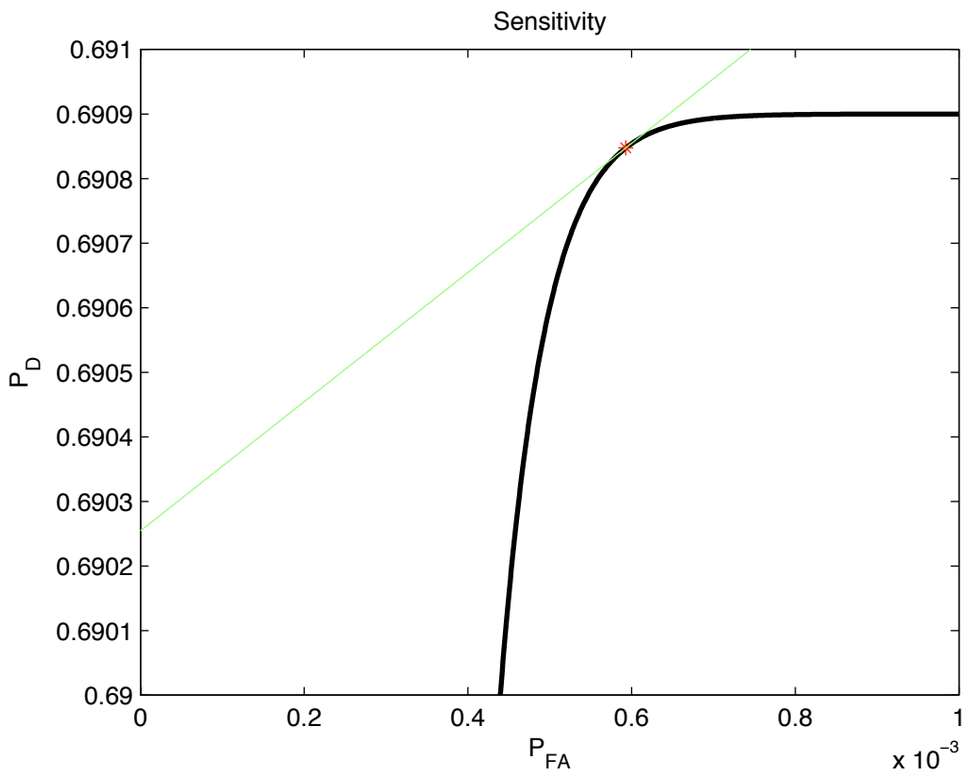


Sensitivity

- An input space **X** is *v-sensitive* if it exists an efficient algorithm **E** such that $\left|\Pr[\mathbf{E}(\mathbf{X})=1|\text{Intrusion}]-\Pr[\mathbf{E}(\mathbf{X})=1|\text{No Intrusion}\right|>v$

- For a given IDS, this optimal point can be found as $\max_{(P_{FA},P_D)\in ROC} P_D - P_{FA}$

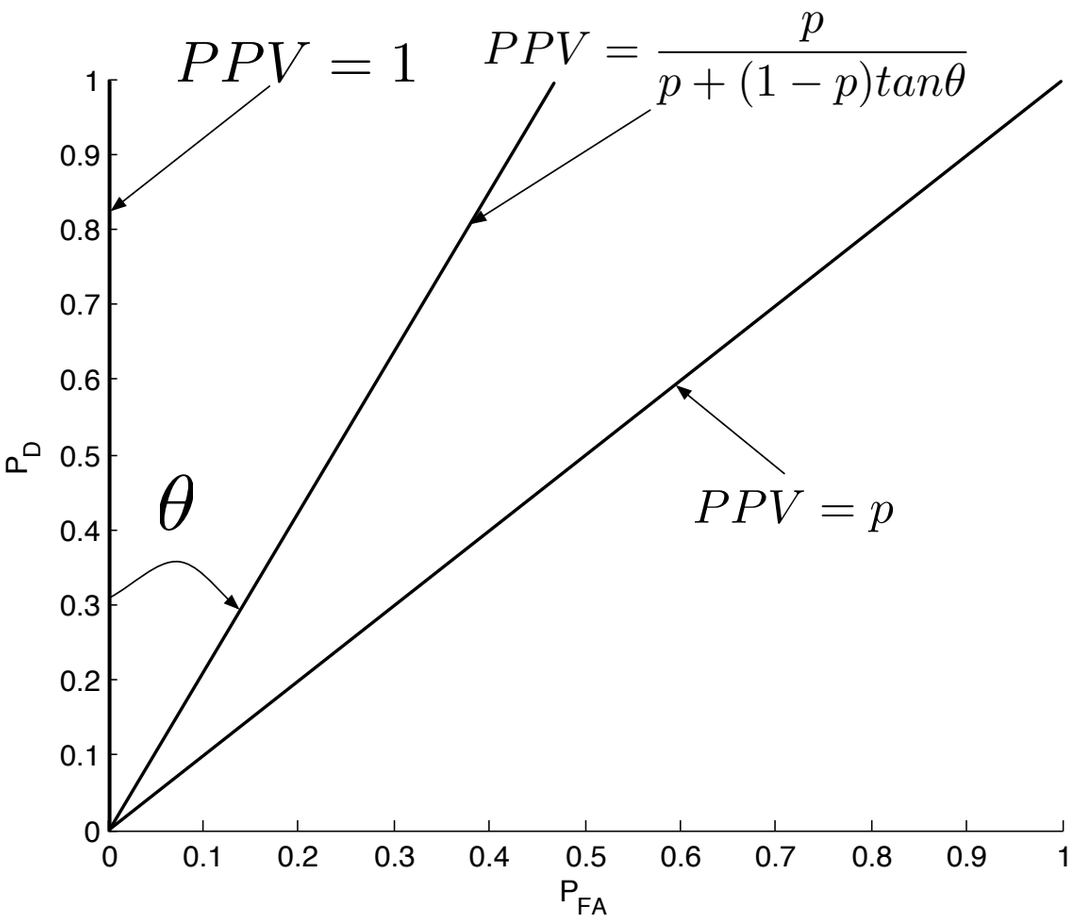
- This corresponds to isolines of the expected cost with slope m=1:

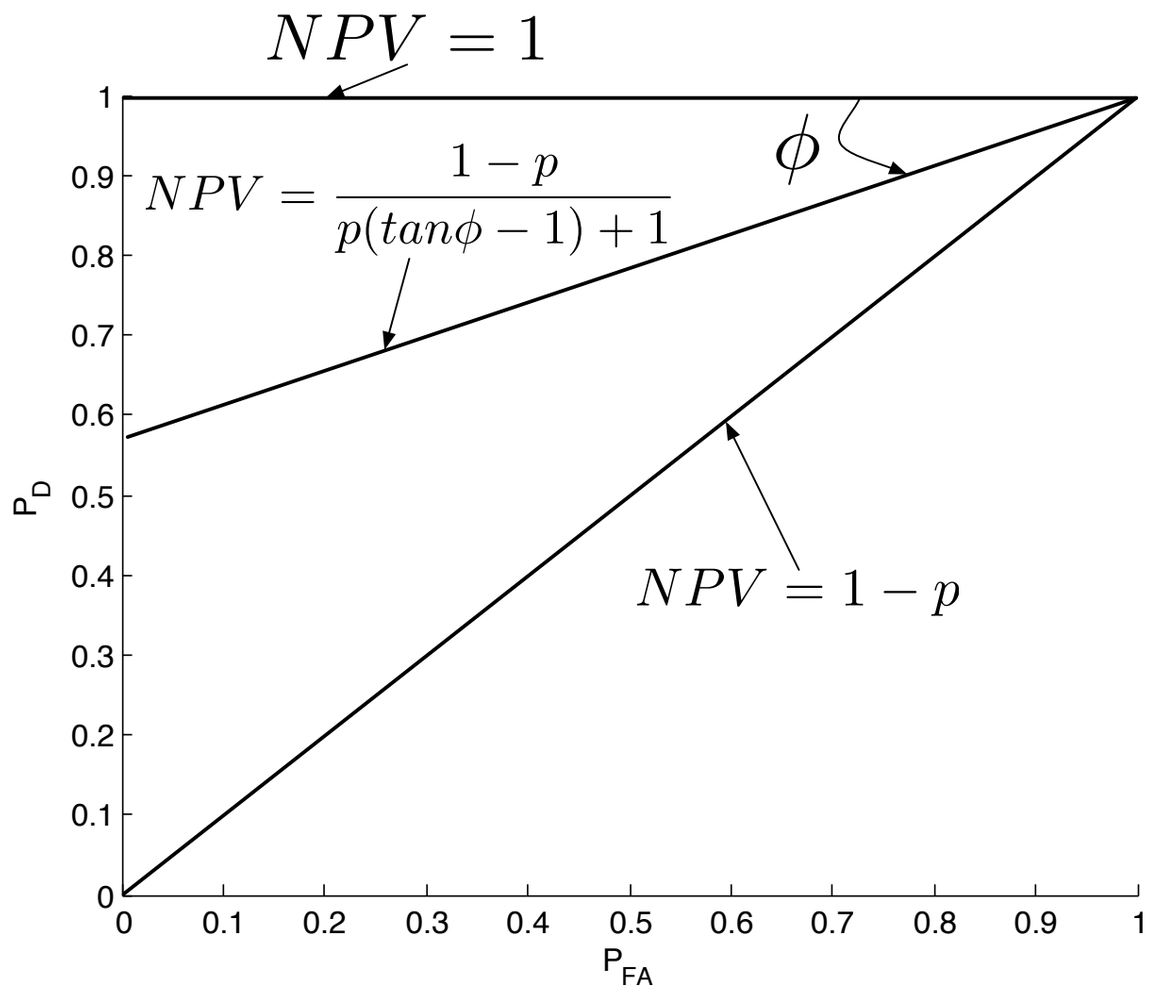


B-ROC Curves: PPV and NPV Isolines

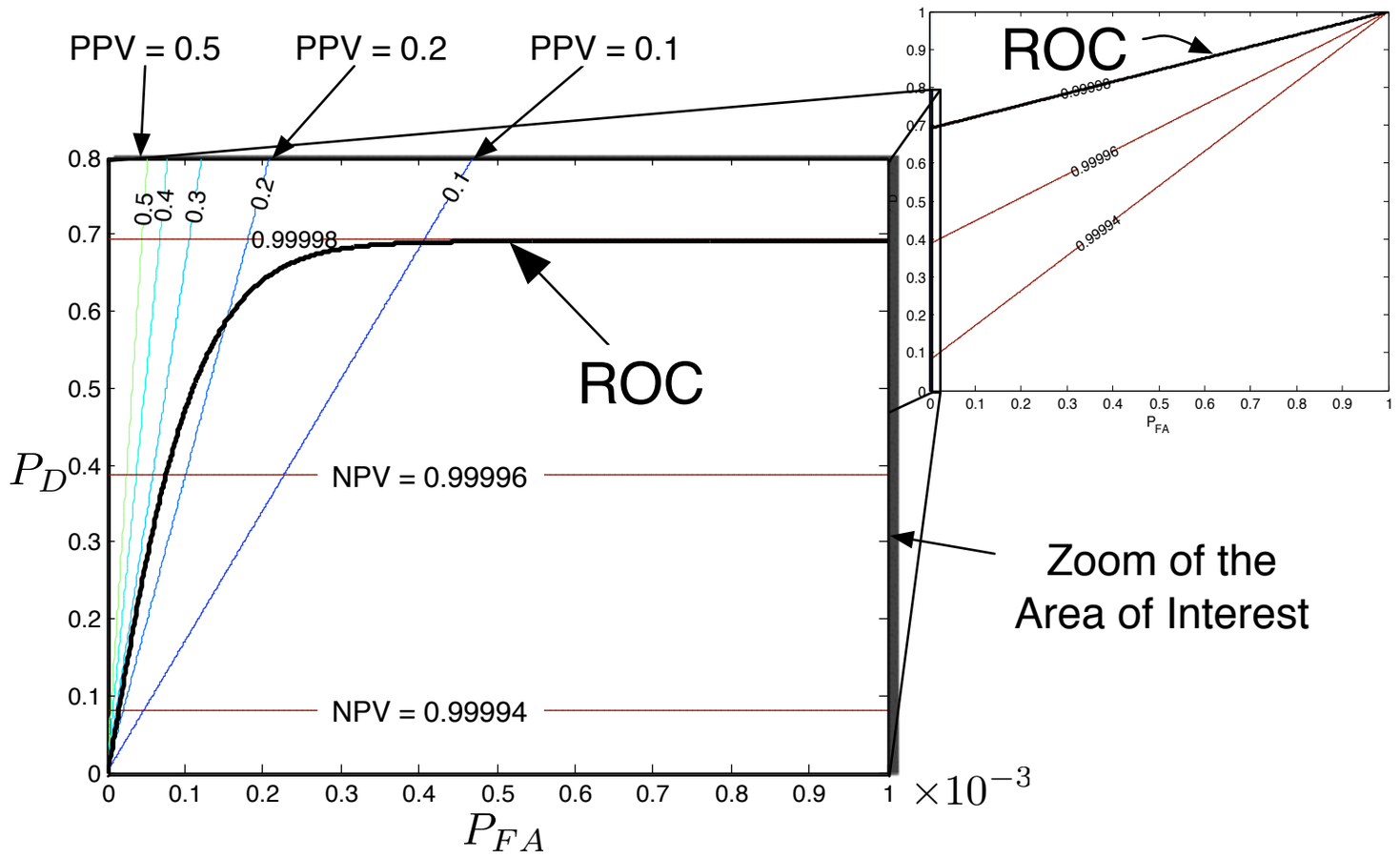
$$\frac{P_{FA2}}{P_{D2}} = \frac{P_{FA1}}{P_{D1}} = \tan\theta$$

$$\frac{1-P_{D1}}{1-P_{FA1}} = \frac{1-P_{D2}}{1-P_{FA2}} = \tan\phi$$





B-ROC Curves: PPV and NPV Isolines in Practice



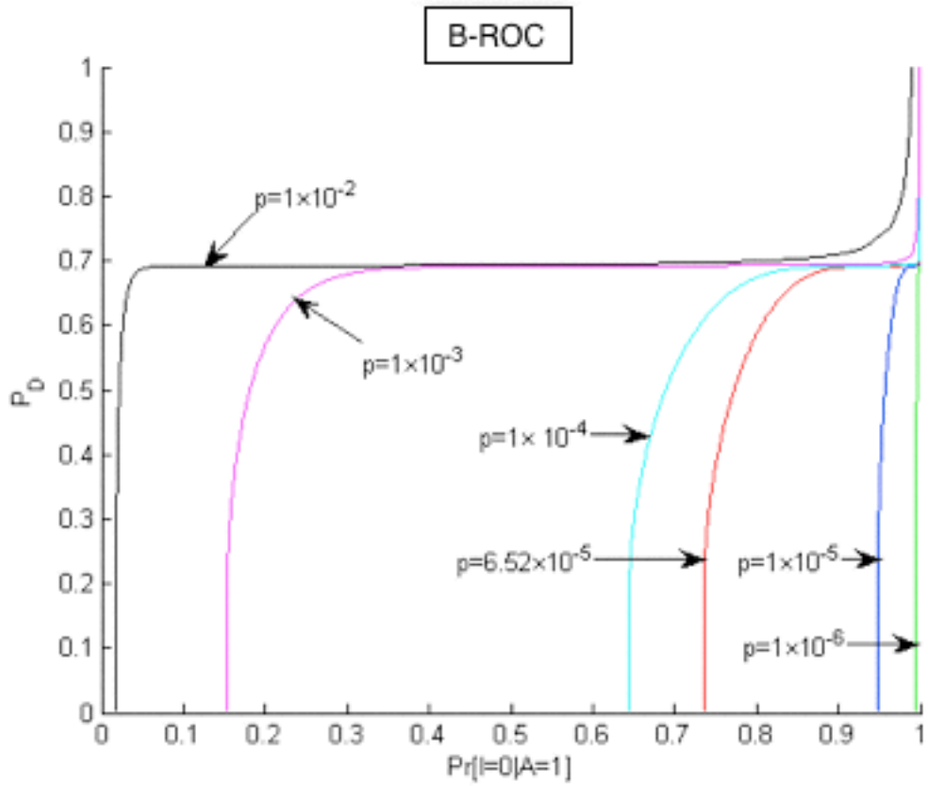
- With PPV and NPV isolines we can see the tradeoff between four variables of interest: P_{FA}, P_D, PPV and NPV.
- For practical considerations however, NPV and P_{FA} are sort of fixed.
- No way to deal with the uncertainty of the base rate *p*

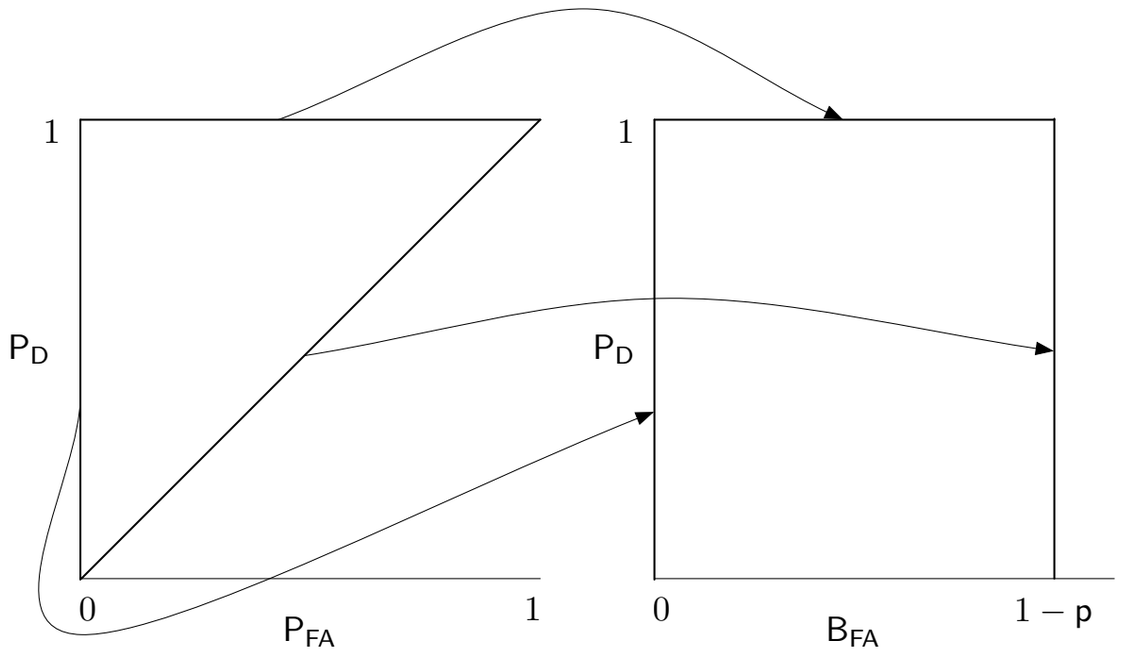
B-ROCs

- P_{FA} is the percentage of normal events that fire an alarm
- B_{FA} is the percentage of alarms that turn out to be false (i.e. B_{FA} =1-PPV)
- B-ROCs show the tradeoff between P_D and B_{FA} for different values of the uncertain parameter *p*
- There is a 1-1 mapping between ROC and B-ROCs. Point (0,0) maps to:

$$ROC'(0^+) = \lim_{P_{FA} \to 0^+} ROC'(P_{FA})$$

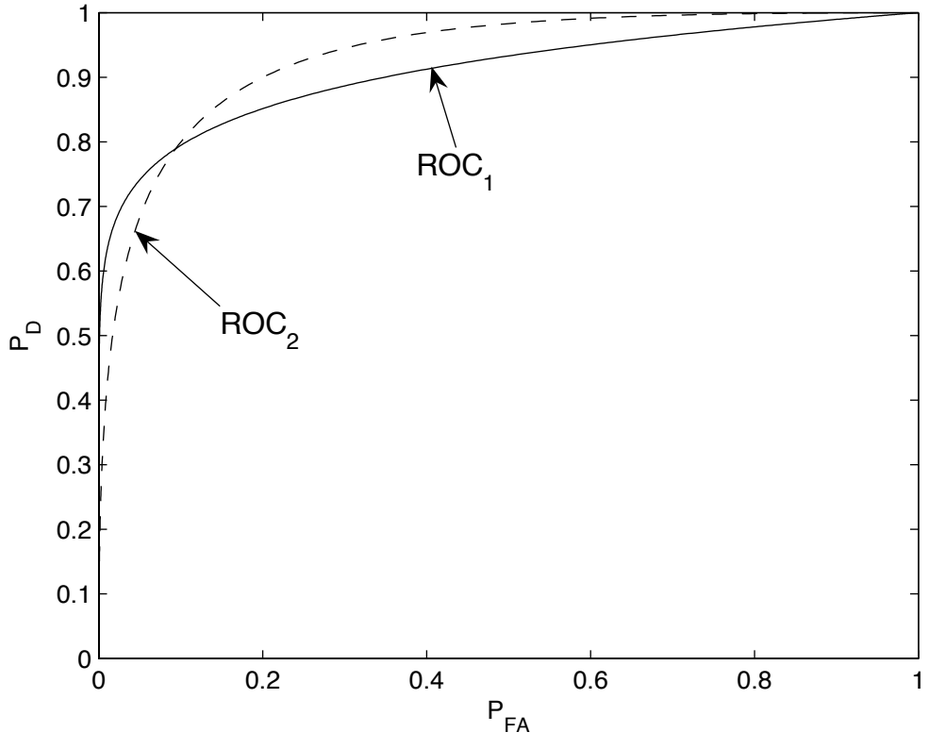
$$\lim_{P_{FA} \to 0^+} B_{FA} = \frac{1-p}{p(ROC'(0^+) - 1) + 1}$$

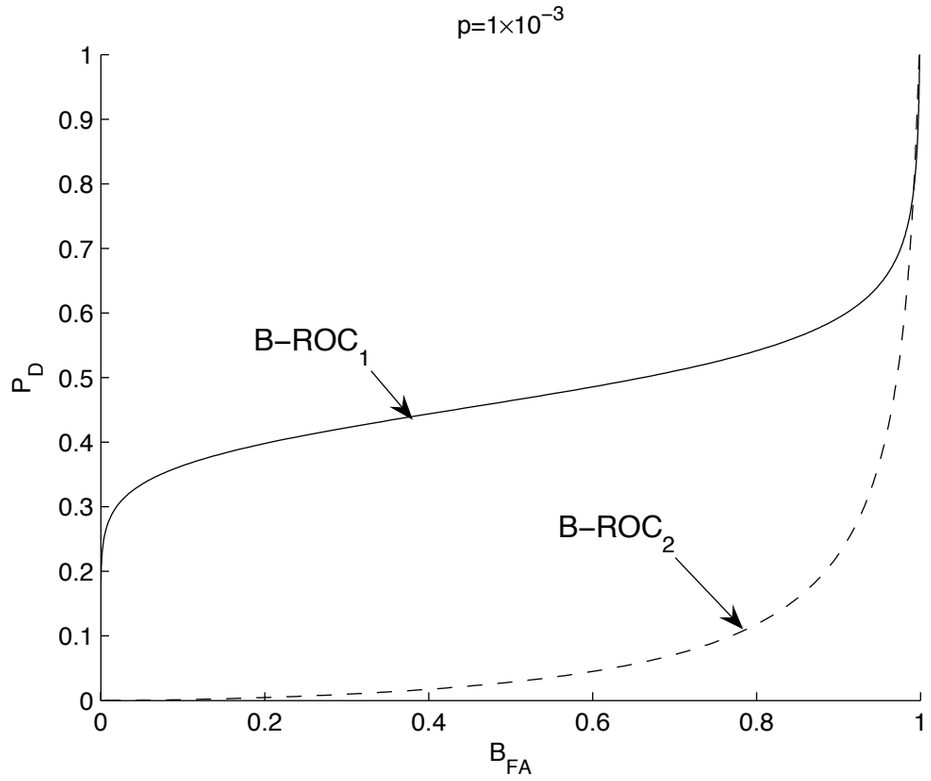




Comparison of two IDSs with B-ROCs

- Comparison between IDSs can also be done with B-ROC curves without assuming misclassification costs:





Comparison of Metrics

Metric	Type	Comparison
ROC	Tradeoff	Can be considered the base of more elaborate metrics
Expected Cost $\mathbf{E}[C(I,A)]$	Single value	Flexible isoline analysis
$C_{ID} = \frac{\mathbf{I}(I;A)}{\mathbf{H}(I)}$	Single value	Instance of expected cost metric (with nonlinear costs)
Bayesian Detection Rate Pr[I|A]	Tradeoff/ Single Value	Average PPV and NPV is an expected cost problem. NPV looses its relevance with small p
Sensitivity Distinguishability	Single value	Expected cost problem with isoline slope = 1
B-ROC	Tradeoff	Same # of assumptions of ROC but more info Better intuition than ROC curves Uncertain p Comparison of classifiers without knowledge of cost

Outline of the Talk

- Unified framework for the study of evaluation metrics
  - Problem: comparison between metrics is difficult since each metric is proposed in a different framework (information theory, decision theory, cryptography, statistics etc.)
  - Our approach: all proposed metrics are instances of the multi criteria optimization problem where the Pareto surface are the ROC curves. Therefore we can compare several metrics in a unified manner. We also introduce new metric: B-ROC curves (a.k.a. IDOC curves).
- Towards secure evaluation
  - Need to include the resistance against attacks as part of an empirical evaluation of an IDS.

Evaluation Guidelines

- **Feasible Design Space**: The design space $\mathcal{S}$ for the IDS.
- **Information Available to the Adversary**: Detection rules? Normal model? training data base? operating point?
- **Capabilities of the Adversary**: Define a feasible class of attackers $\mathcal{F}$.
- **Evaluation Metric**: Measure of how well the IDS meets our desired properties. We call an evaluation **M robust** if its metric outcome **M** is satisfied for any attacker in $\mathcal{F}$
- **Goal of the Adversary**: The intruder can use its capabilities and information to perform two main classes of attacks. Evaluation Attack and Base System Attack
- **Model Assumptions**:Limit the number of assumptions and evaluate the resiliency of the remaining ones. Security depends above all on the assumptions made!

Motivation for Guidelines: Secret Key Cryptography Example

- **Feasible Design Space**: $\mathcal{S}$ is the set of PPT algorithms that satisfy correctness: for any *sk* and *m* $D_{sk}(E_{sk}(m))=m$.

- **Information Available to the Adversary**: The only information originally not available to the adversary is *sk*.

- **Capabilities of the Adversary**: $\mathcal{F}$ is the set of PPT algorithms with extra capabilities modeled with oracles (e.g. can get extra information: chosen-plaintext attacks).

- **Evaluation Metric**: $\left|\Pr[\mathcal{A}=1|\text{World 1}]-\Pr[\mathcal{A}=1|\text{World 2}]\right|$. Algorithm proposed is **secure** (robust) if the above is negligible for all $\mathcal{A}$ in $\mathcal{F}$.

- **Goal of the Adversary**: Evaluation attack.

- **Model Assumptions**: Cryptographic primitives such as one way functions.

How to model an adaptive intruder?

- Implicit assumption of evaluating in labeled data: Stationarity (i.e., the assumption is that of a non-adaptive intruder).

- Simplest approach: model the average intruder with respect to the parameters we have already used: p, P_{FA}, P_D (PPV and other metrics depend on these values).

- Evasion attacks: Some intruders (or intrusions) might find ways to avoid the IDS, while others will still get caught. Result: inferior P_D. Parameter: β

- Base-Rate attacks: Uncertainty of p already discussed. Now how to find the least favorable p? Parameter: δ

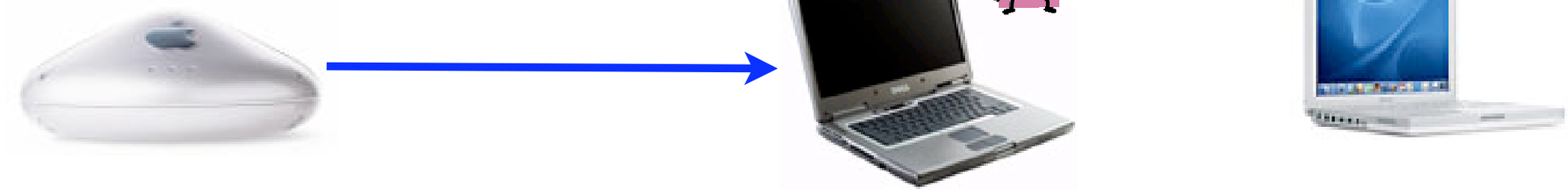
- DoS attacks: How feasible is it to create false alarms? False alarms can also increase without the intervention of a real attacker. Parameter: α

Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks

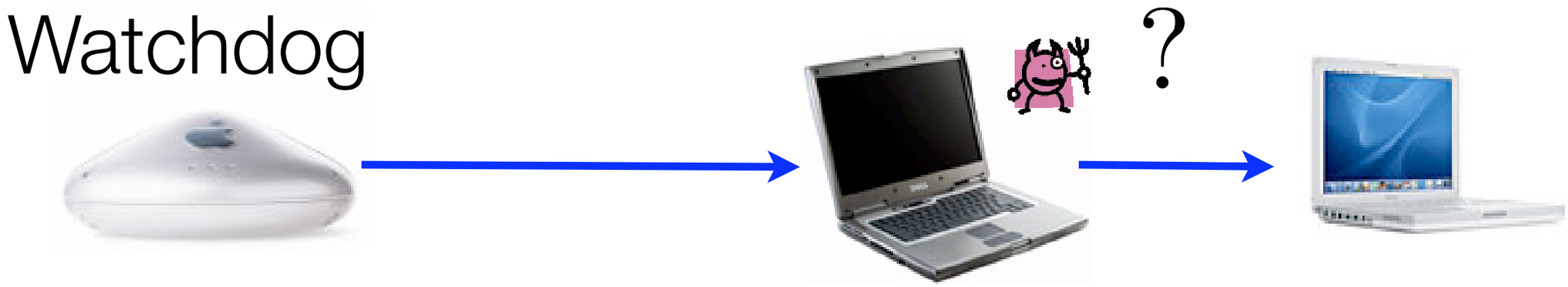
Watchdog

Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks

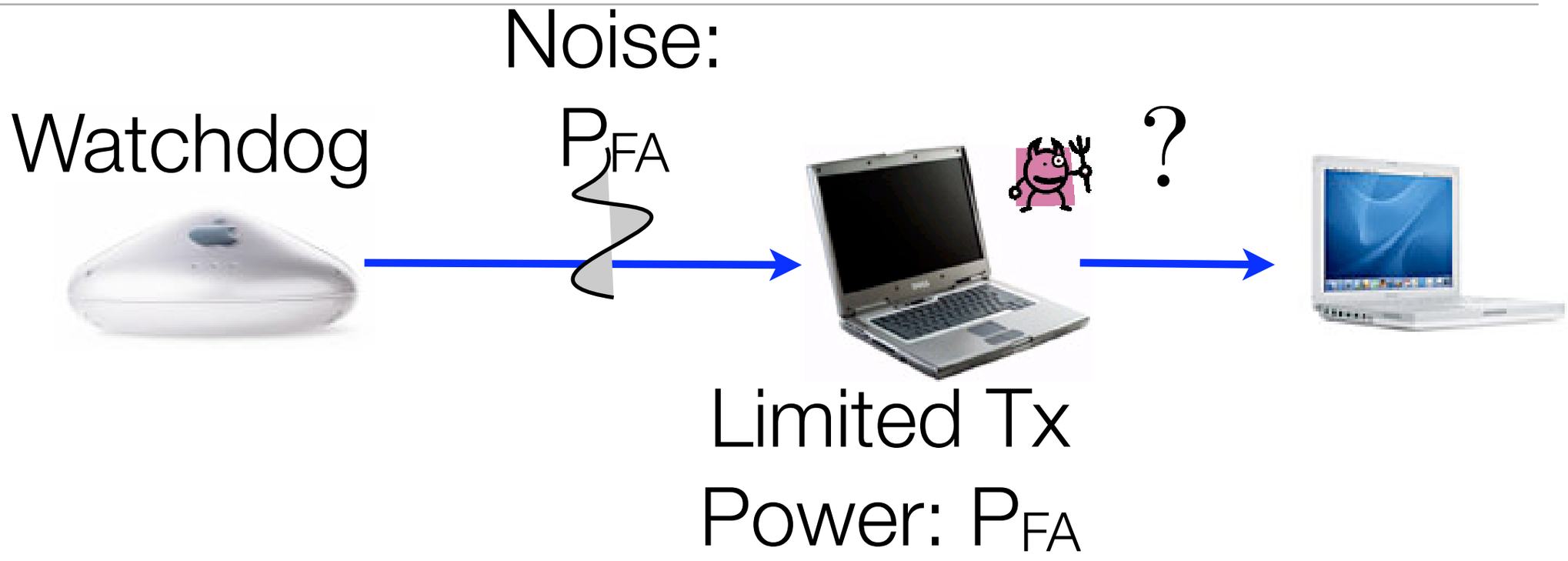
Watchdog



Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks



Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks



Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks

Channel Busy: $1-P_D$

Watchdog

Noise: P_{FA}

?

Limited Tx Power: P_{FA}

Evaluation 1: Selfish Behavior in Wireless Ad Hoc Networks

Channel Busy: $1-P_D$

Noise: P_{FA}

Watchdog

?

Limited Tx Power: P_{FA}

Watchdog verifies if packet was forwarded or not. It then has four possible options:

$$h_1(V) = 0 \quad h_1(\neg V) = 0$$
$$h_2(V) = 1 \quad h_2(\neg V) = 0$$
$$h_3(V) = 0 \quad h_3(\neg V) = 1$$
$$h_4(V) = 1 \quad h_4(\neg V) = 1$$

Evaluation 1: Formulation

- **Feasible Design Space**: Select h_i with probability π_i . Then $\mathcal{S} = \left\{ \pi_i : \sum_{i=1}^{4} \pi_i = 1 \right\}$
- **Information Available to the Adversary**: Intelligent Adversary (i.e., omniscient).
- **Capabilities of the Adversary**: A selfish node can arbitrarily select either to forward or drop a packet, therefore $\mathcal{F} = \{p \ : \ p \in \delta = [0, 1]\}$
- **Goal of the Adversary**: Evaluation attack.
- **Evaluation Metric**: Minimize the probability of misclassifying a node. Let

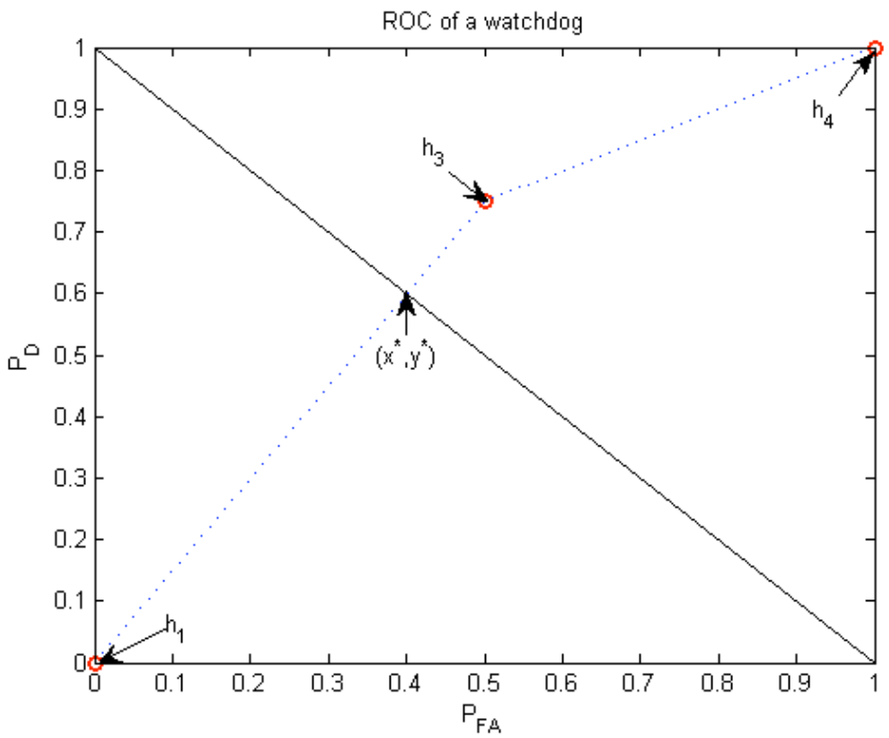
$$r = \min_{\pi \in \mathcal{S}} \ \max_{p \in \mathcal{F}} \ P_{Error}[\pi, p]$$

  - Then π^* is an optimal detection strategy if

$$\forall p \in \mathcal{F}, \quad P_{Error}[\pi^*, p] \leq r$$

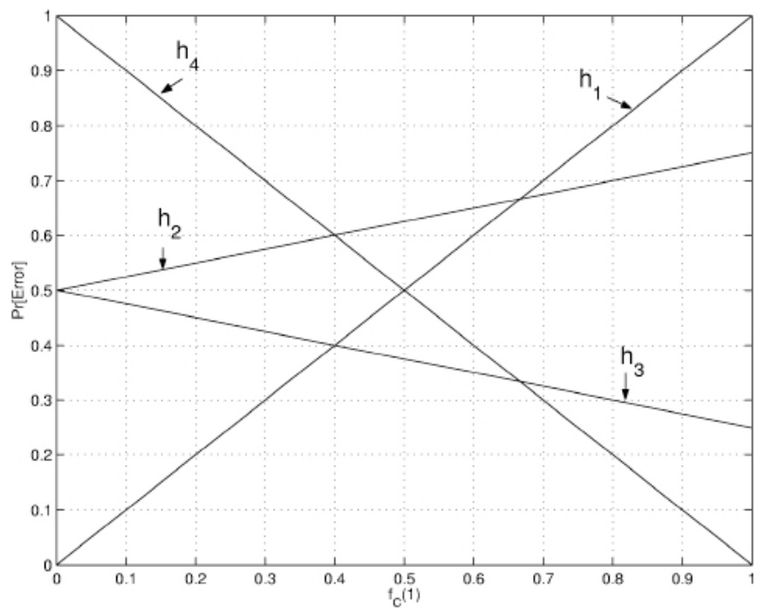
- **Model Assumptions**: An accurate upper bound on the false alarm and detection rates for the verification of each packet!

Evaluation 1: Solution



$$p^* = \frac{P_{FA}}{P_{FA} + P_D} = 2/5$$

$$\pi_1^* = \frac{P_{FA} - (1 - P_D)}{P_{FA} - (1 - P_D) + 1} = 1/5$$



$$\pi_3^* = \frac{1}{P_{FA} + P_D} = 4/5$$

Evaluation 2: Buffer Overflow Detection

- **Feasible Design Space**: Compare buffer length of each program execution with a buffer threshold, therefore $\mathcal{S} = \{t \ : \ t \text{ is a threshold } \}$

- **Information Available to the Adversary**: Intelligent Adversary (i.e., omniscient).

- **Capabilities of the Adversary**: $\mathcal{F} = \{(p, p_2, p_3) \ : \ p \in \delta, \ p_2 \in [0, \alpha] \ p_3 \in [0, \beta]\}$

- **Evaluation Metric**: Expected cost $r(t, \mathcal{I}) = \mathbf{E}[C(I, A)]$ (B-ROC in paper)
  - We want to find $t^* \in \mathcal{S} \ s.t. \ \forall \mathcal{I} \in \mathcal{F} \ r(t^*, \mathcal{I}) < r(t, \mathcal{I})$

- **Goal of the Adversary**: Evaluation attack, $\mathcal{I}^* \in \mathcal{F} \ s.t. \ \forall t \in \mathcal{S} \ r(t, \mathcal{I}) < r(t, \mathcal{I}^*)$

- **Model Assumptions:** We assume our estimate of the parameters is accurate!

$$p \in \delta = [1.5 \times 10^{-4}, 1.6 \times 10^{-4}] \quad \alpha = 1 \times 10^{-5} \quad \beta = 0.1$$

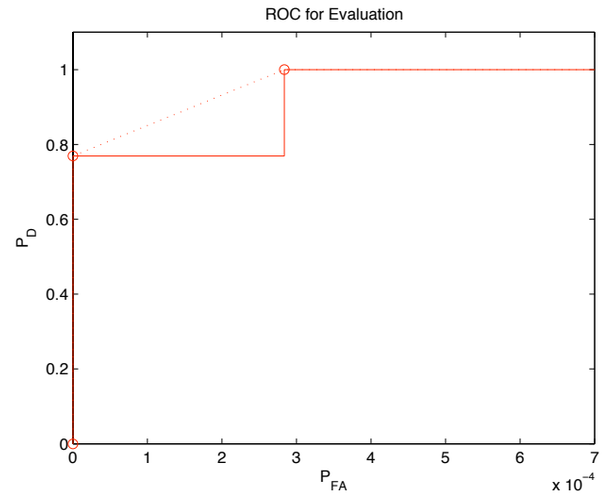
$$C(0,0) = C(1,1) = 0 \quad C(0,1) = 10000 \ C(1,0) = 85000$$

Evaluation 2: Buffer Overflow Detection

Evaluation in data set $\mathcal{E}$

t^d=399,

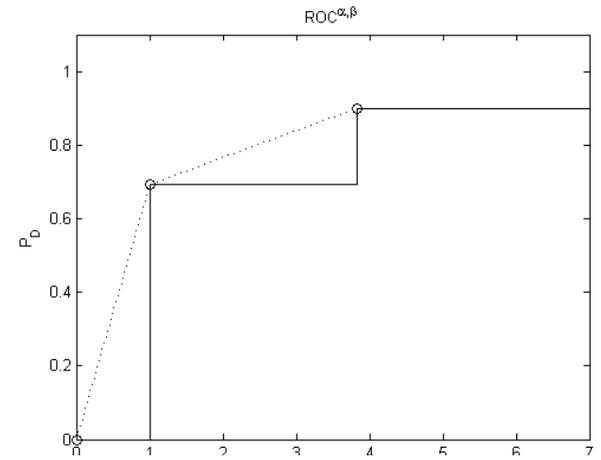
$r(t^d,\mathcal{E})$=2.83



Evaluation with Intruder model

t^*=799,

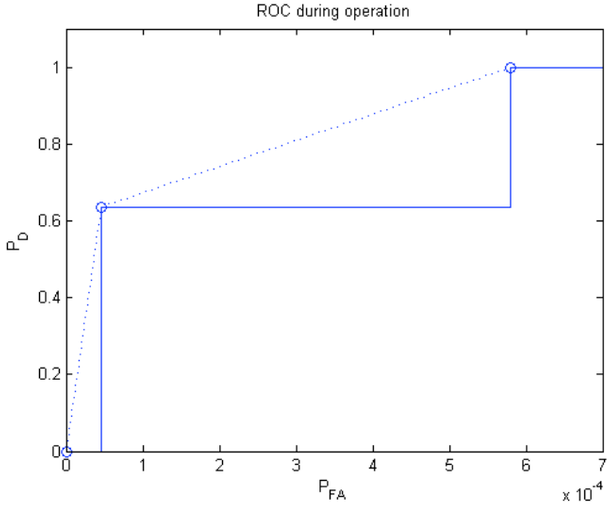
$r(t^*,\vartheta^*)$=5.19



Performance of both solutions in the "real" environment $\mathcal{R}$:

t^d=399, $r(t^d,\mathcal{R})$=6.934

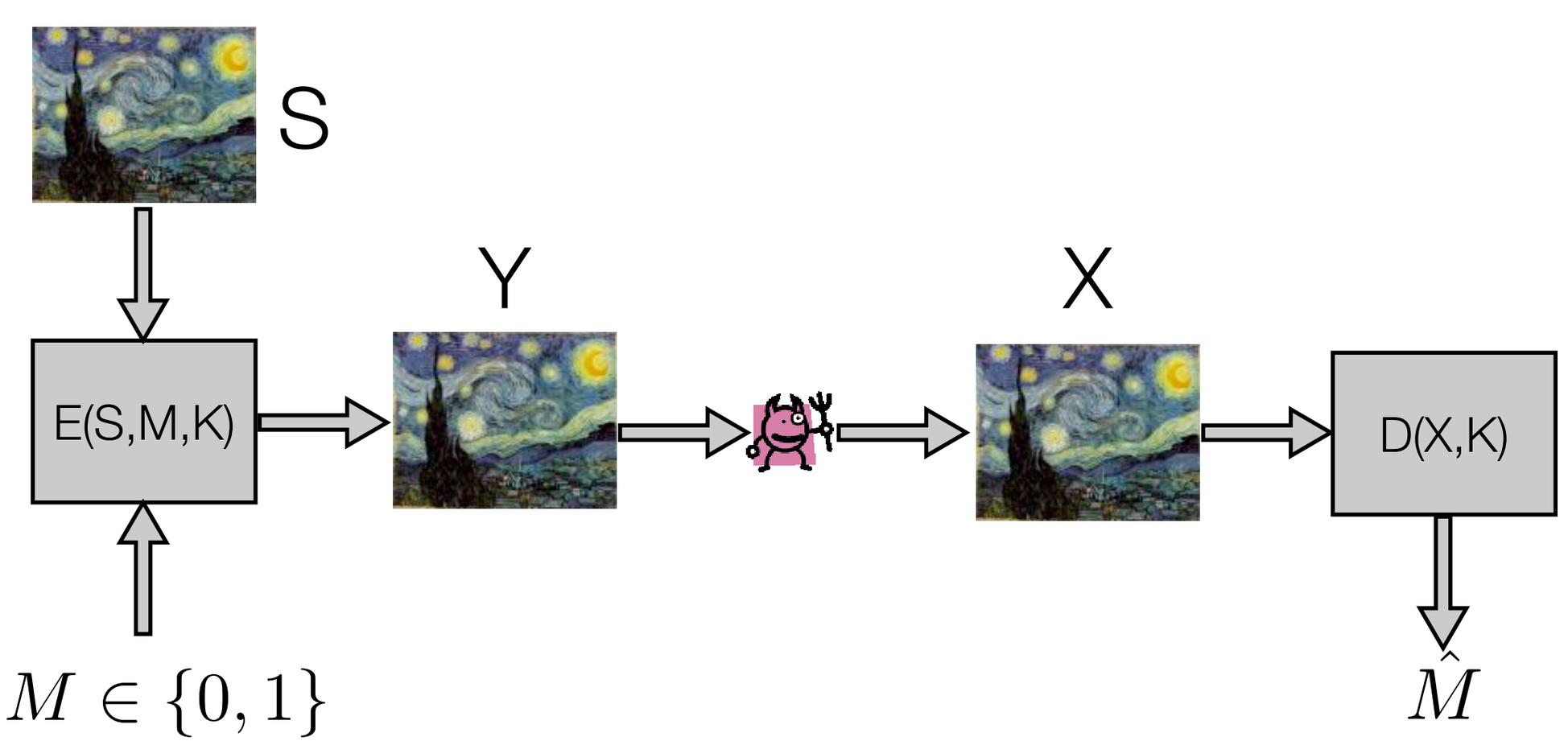
t^*=799, $r(t^*,\mathcal{R})$=2.73



Evaluation 3: Watermark Verification Problem

- Applications of watermarking and data hiding in multimedia include copyright protection, document authentication, broadcast monitoring, etc.

- Watermark verification problem: encoder E and decoder D share K used to generate a watermark W←$f_w(w)$:



Evaluation 3: Formulation

- **Feasible Design Space**: $\mathcal{S} = \{f_w : \mathbf{E}[d(S, Y(W))] \leq D_w\}$

- **Information Available to the Adversary**: Given ***y***, the attacker does not know the contributions of ***s*** and ***w***. The attacker knows however f_w and f_s and the detection rule (MAP)

- **Capabilities of the Adversary**: $\mathcal{F} = \{f_{x|y} : \mathbf{E}[d(X, S)] \leq D_a\}$

- **Evaluation Metric**: $r^* = \min_{f_w \in \mathcal{S}} \max_{f_{x|y} \in \mathcal{F}} r(f_w, f_{x|y})$

  - f_w^* is secure if $\forall f_{x|y} \in \mathcal{F} : \quad r(f_w^*, f_{x|y}) \leq r^*$

- **Goal of the Adversary**: Evaluation attack

- **Model Assumptions:** Assumed knowledge (or existence) of f_s, and realistic distortion metrics *d* for the average case (expectation) with upper bounds D_w and D_a

Evaluation 3: Previous Work (Moulin and Ivanovic)

- Distortion: per-sample squared-error metric: $d(s,x) = ||x-s||^2 N^{-1}$
- Distribution of the source signal: $f_s = \mathcal{N}(0, R_s)$ and watermark: $f_w = \mathcal{N}(0, R_w)$
- Spread Spectrum Watermarking (with scaling factor) $y = \Phi(s+w)$
- Gaussian attack: $x = \Gamma y + e$ where $f_e = \mathcal{N}(0, R_e)$
- Evaluation Metric: $r(f_w^*, f_{x|y}) = \Pr[Error]$
- Extra assumptions: received process can be “whitened” (this depends on the attacker!) and approximation to the probability of error (without any bounds!)

Evaluation 3: Our Contributions, Optimal watermark distribution

- The exact probability of error can be easily computed as:

$$\Pr[Error] = \mathbf{E}\left[\mathcal{Q}\left(\sqrt{w^t \Omega w}\right)\right] = \int \mathcal{Q}\left(\sqrt{w^t \Omega w}\right) f_w(w) dw$$

where

$$\Omega = \Phi^t \Gamma^t R_y^{-1} \Gamma \Phi$$

- However Q($x^{1/2}$) is a convex function and thus we can use Jensen's inequality:

$$\Pr[Error] = \mathbf{E}\left[\mathcal{Q}\left(\sqrt{w^t \Omega w}\right)\right] \geq \mathcal{Q}\left(\sqrt{\mathbf{E}[w^t \Omega w]}\right) = \mathcal{Q}\left(\sqrt{tr\{\Omega R_w\}}\right)$$

- Assuming Rw is fixed, the lower bound on the error is achieved if with prob. 1

$$w^t \Omega w = tr\{\Omega R_w\}$$

therefore if we let the SVD of the matrix be $R_w^{1/2} \Omega R_w^{1/2} = U \Sigma U^t$

we can use $w = R_w^{1/2} U A$

where the elements of A are +1 or -1 with equal probability. It is easy to check that the above distribution satisfies the two constraints

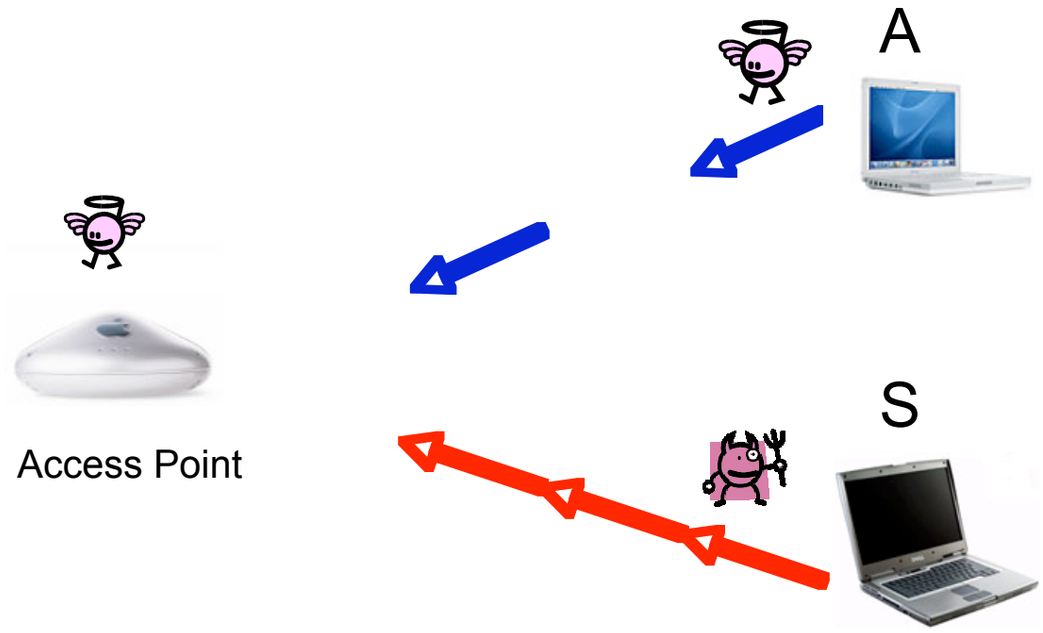
Evaluation 3: Our Contributions, Solution without "whitening" assumption

- Overall Objective: $\max_{\Phi, R_w} \min_{\Gamma, R_e} tr\left\{\Phi^t \Gamma^t (\Gamma \Phi R_s \Phi^t \Gamma^t + R_e)^{-1} \Gamma \Phi R_w\right\}$
- Subject to: $tr\{(\Phi - I) R_s (\Phi - I)^t + \Phi R_w \Phi^t\} \leq N D_w$

$$tr\{(\Gamma\Phi - I) R_s (\Gamma\Phi - I)^t + \Gamma\Phi R_w \Phi^t \Gamma^t + R_e\} \leq N D_a$$

- Tools: Tr{$A^t B$} is an inner product: $(tr\{A^t B\})^2 \leq tr\{A^t A\} tr\{B^t B\}$
  - equality iff A=kB, where k is a scalar. Another tool: variational methods.
- Problems: We still rely on Gaussian Attacks. New research focuses on non-linear attacks.
- We have a toy version where we give the attacker complete control of the attack distribution subject to different distortion constraints.

Evaluation 4: Previous work in MAC layer Misbehavior



- **DOMINO** (Raya et. al.)
  - If pre_alarm for S_i
    - Cheat_count(S_i) = Cheat_count(S_i) + 1
    - If Cheat_count(S_i) > K then Alarm & "Punish"
    - Else if Cheat_count(S_i) > 0
      - Cheat_count((S_i) = Cheat_count(S_i) - 1
  - Pre_alarm:
    - $X_{av} < \gamma Bnom = thresh$
    - Adversary is rational but not intelligent:
  - adversary chooses its backoff as *(1 - m)CWmin*

Evaluation 4: Analysis of Previous Work

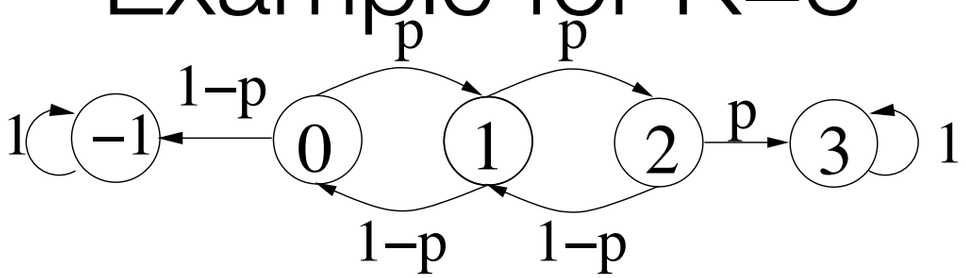
Characterized by three parameters. p: the probability of pre-alarm, and u_0, u_k: the times to absorption

$$
\begin{aligned}
p &= \Pr\left[\sum_{i=1}^{n} X_i \le n\gamma B\right] \\
&= \sum_{k=0}^{\lfloor n\gamma B\rfloor} \Pr\left[\sum_{i=1}^{n} X_i = k\right] \\
&= \sum_{k=0}^{\lfloor n\gamma B\rfloor} \sum_{\{(x_1,\ldots,x_n):\sum_{i=1}^{n} x_i = k\}} \frac{1}{CW^n}
\end{aligned}
$$

$$
\begin{bmatrix}
-p & p & 0 & 0 & \cdots & 0 \\
1-p & -1 & p & 0 & \cdots & 0 \\
0 & 1-p & -1 & p & 0 & 0 \\
0 & 0 & 1-p & -1 & p & 0 \\
 & & \vdots & & & \\
0 & 0 & \cdots & 0 & 1-p & -1
\end{bmatrix}
\begin{bmatrix} \mu_0 \\ \mu_1 \\ \mu_2 \\ \vdots \\ \mu_K \end{bmatrix}
=
\begin{bmatrix} -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{bmatrix}
$$

New dimension: Time!

Example for K=3



$$\mathbf{E}[\text{time to false alarm}] = \mu_0 = \frac{1-p+2p^2+2p^3}{p^4}$$

Conclusions

- We introduced a framework to compare and analyze several previously proposed metrics: expected cost, ID capability, PPV, NPV and sensitivity

- B-ROC curves are good as a metric for any classification problem with class imbalances

- First steps toward analyzing the security of empirical evaluations of IDSs.

- On the goal of the adversary:

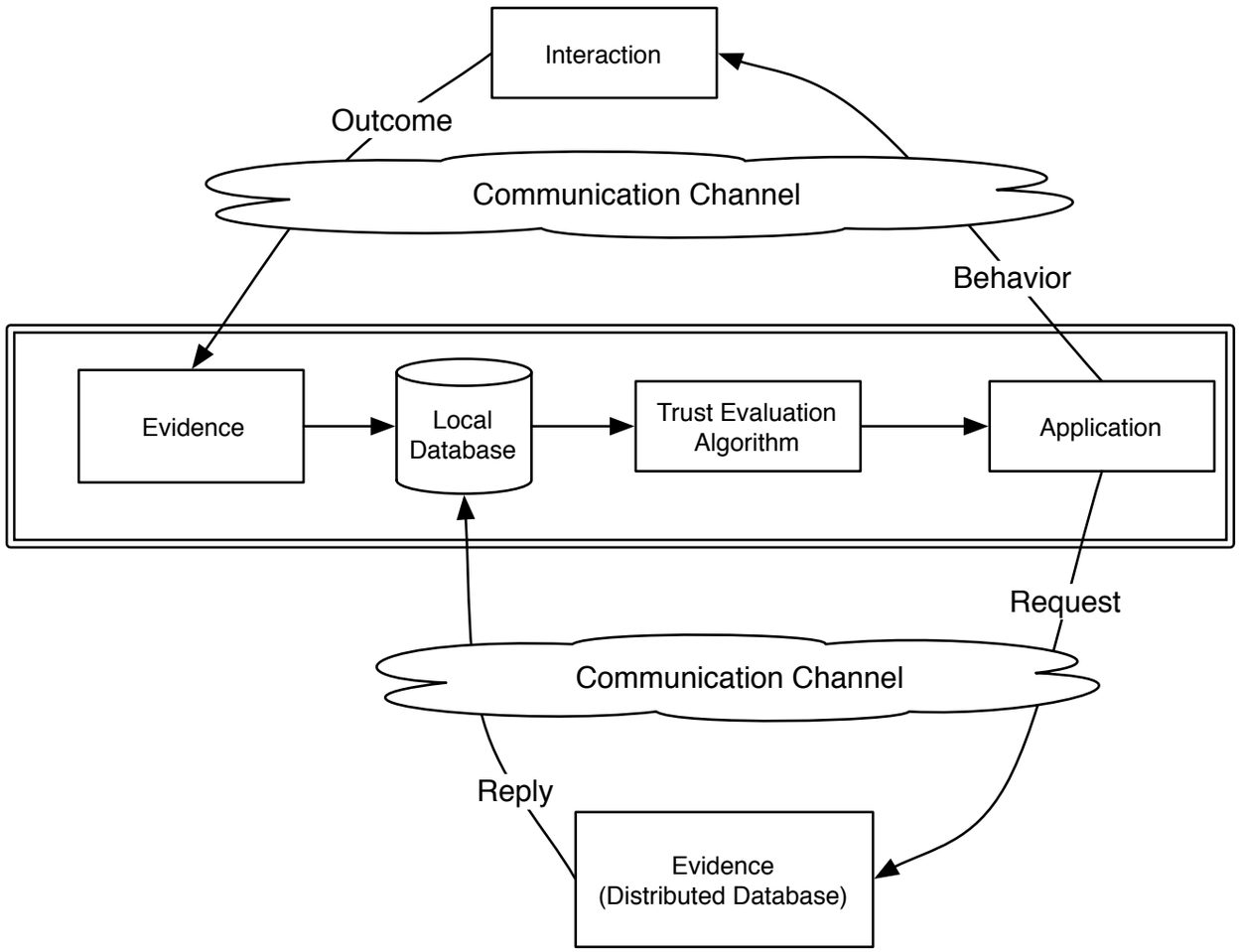
	Advantage	Disadvantage
Evaluation Attacks	More robust against modeling errors	Pessimistic evaluation: might be too restrictive
Base System Attacks	Can model more realistic attackers (Mimicry attacks)	Makes extra assumptions that might not hold in practice

Future Work

- Classification accuracy is only one of the metrics to consider
  - Need more general evaluation methodology, such as risk assessment
  - New tradeoff parameters. We are exploring the throughput impact of MAC layer misbehavior. We no longer consider missed misbehavior but misbehavior that affects our throughput.
- How to optimally combine scores from different sensors
  - When all sensors are trusted (alarm correlation for IDSs)
  - When sensors are not 100% trustworthy (trust/reputation systems)

Evaluation of Distributed Classification Or Reputation Based Systems

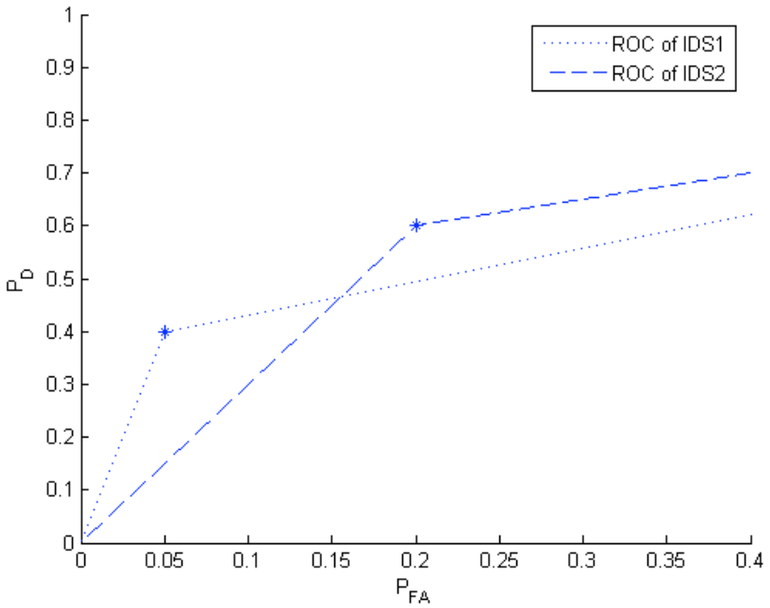
- Evidence=<Issuer, Target, Action, Statement, Confidence of Evidence, Time>

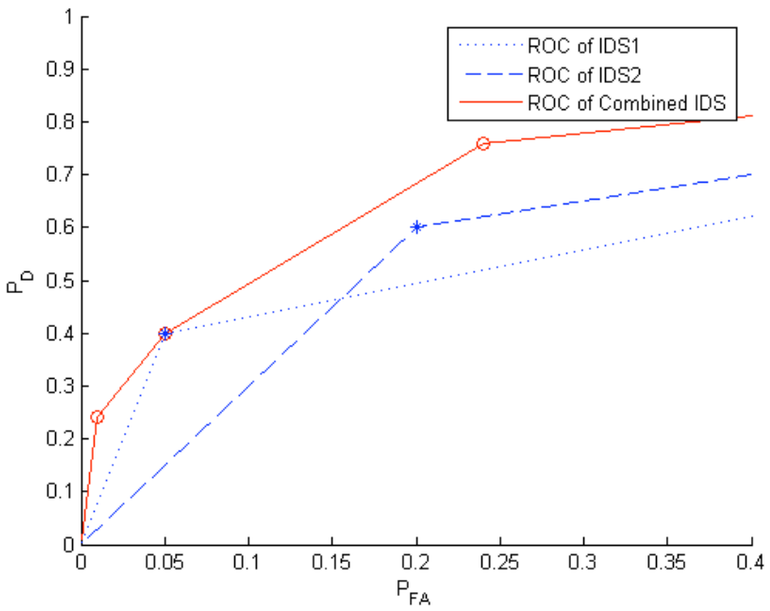


Finding the Feasible Design Space of TES (and its Adversary)

	Interactions	Evidence				Trust Evaluation	Application	Adversary		
		Application		Recommender			(Decision Maker)	Objective	Capabilities	Strategy
		Trust	Confidence	Trust	Confidence					
Maurer96	Public Key Certificates	Every User Adds to its certificate a value between 0 and 1	NA as Evidence, but implicit in Trust Metric Algorithm: more evidence (more paths) implies higher confidence	Value between 0 and 1	NA	Combine several paths in a probability framework	NA	NA	NA	NA
Reiter98	Public Key Certificates	0 or 1: a key is authentic or not	NA as Evidence. Implicit in metric: independent paths	NA: An authentic PK is considered trusted	NA	Find number K of disjoint paths or k-connective paths with bounded length	Compare K to a threshold	NA	NA	NA
Reiter99	Public Key Certificates	Amount of money the owner of K_1 insures attributes and behavior of K_2	NA	NA	NA	Find the Minimum capacity cut to determine minimum amount of money a user can expect to recover	NA	Malicious	Compromise keys	No intelligent strategy
Josang99	Certification Chains	$1 > Belief > 0$ and $1 > Disbelief > 0$	$Uncertainty = 1 - Belief - Disbelief$	Same as Application Trust	The Same as Application confidence	Subjective Algebra	NA	Malicious	?	?
Levien98	Certification Chains	1 or 0 trust	(implicit)	Delegation Trust 0 or 1	NA	Compares shortest path, network flow trust metric, disjoint paths (Reiter98) and best case (this paper)	threshold	Malicious	Node attack: generates arbitrary certificates (delegation or binding). Edge attack: trick honest node to certify untrustworthy keys as trustworthy. Attacker can chose Node and Edge Attacks Randomly or Targeted (to maximize its chances of success)	success of attack defined as the fraction of sources accepting a particular forgery or the probability that nodes accept an invalid binding
KeyChains	Certificate Chains	PGP style trust	NA	NA	NA	PGP	NA	Malicious	Man in the middle: interferes with P2P comm. but does not corrupt peers. Peer Compromising adversary. User Compromising adversary: able to change trust graph	What is the objective of the attacker?
EigenTrust	File downloads	Successful downloads - Unsuccessful downloads	NA	Assumed the same as appl. Trust	NA	Adjacency matrix iterations to achieve a global value	Isolate peers with low trust values. Give prizes to peers with high trust values. Nodes choose who to interact based on: Deterministic Algorithm: Everyone Chooses peer with highest trust value. Probabilistic algorithm depending on trust values	Malicious		Four threat models. The strategy that achieves the maximum number o finauthentic downloads given the same effort (providing authentic upoloads) is the fourth threat model: percentage of malicious peers acts normally and recommend bad nodes, bad nodes always defect.
Buchegger04	routing or file downloads in P2P	Parameters of a Beta distribution (α', β')	Implicity in (α', β'). However the confidence is not considered in Metric	Parameters of another Beta distribution: whether reported info is likely to be true	yes	Beta distribution parameters updated with each interaction an based on recommendations	Selfish?	set of malicious nodes in the protocol	no intelligent strategy	
Jiang06	Abstract neighbor interactions	$c_{ij} \in -1, 1$	NA	To avoid conflicting reports they use $\hat{c}_{ij} = c_{ij} + c_{ji}$	NA	Use properties of a Markov Network to compute the probability estimate of a node being good or bad	Given the estimated pdf, the final policy is just a realization of it	Malicious	set of nodes in the network	Non-colluders (falsely incriminate neighbors). Colluders: vote good for colluders, and vote -1 for everyone else. Random: vote randomly for neighbors Empirically, Random adversary degrades the estimate the most
Theodor06	Abstract interactions	Values in [0,1]	Confidence in [0,1]	NA	NA	Semiring algebra to compute trust in the graph	NA	Malicious	set of nodes in the network	give good reputation to colluders and bad reputation to honest nodes
Guha04		?	?	?	?	?	?	?	?	?
Blaze96	Is a user allowed to access certain resouces?	Assertions and certificates	?	?	?	Evaluates certificates received and the policy to determine whether user is allowed certain actions	?	NA	NA	NA
KeyNote	?	?	?	?	?	?	?	?	?	?

Future Work
Alarm Correlation





Can we use the same framework for alarm correlation?

$$lr(\neg A, \neg A) = \frac{(1-P_{D1})(1-P_{D2})}{(1-P_{FA1})(1-P_{FA2})}$$

$$lr(A, \neg A) = \frac{(P_{D1})(1-P_{D2})}{(P_{FA1})(1-P_{FA2})}$$

$$lr(\neg A, A) = \frac{(1-P_{D1})(P_{D2})}{(1-P_{FA1})(P_{FA2})}$$

$$lr(A, A) = \frac{(P_{D1})(P_{D2})}{(P_{FA1})(P_{FA2})}$$

- 5 vertex points example:
  - Always fire an alarm
  - Fire an alarm whenever IDS1 or IDS2 fire an alarm
  - Fire an alarm if IDS1 fires an alarm
    - Firing an alarm if IDS2 fires an alarm is suboptimal!
  - Fire an alarm only when both IDS fire an alarm
  - Never fire an alarm