CS 599: Formal Methods in Security and Privacy

Quantitative Information Flow

Marco Gaboardi
gaboardi@bu.edu

Alley Stoughton
stough@bu.edu

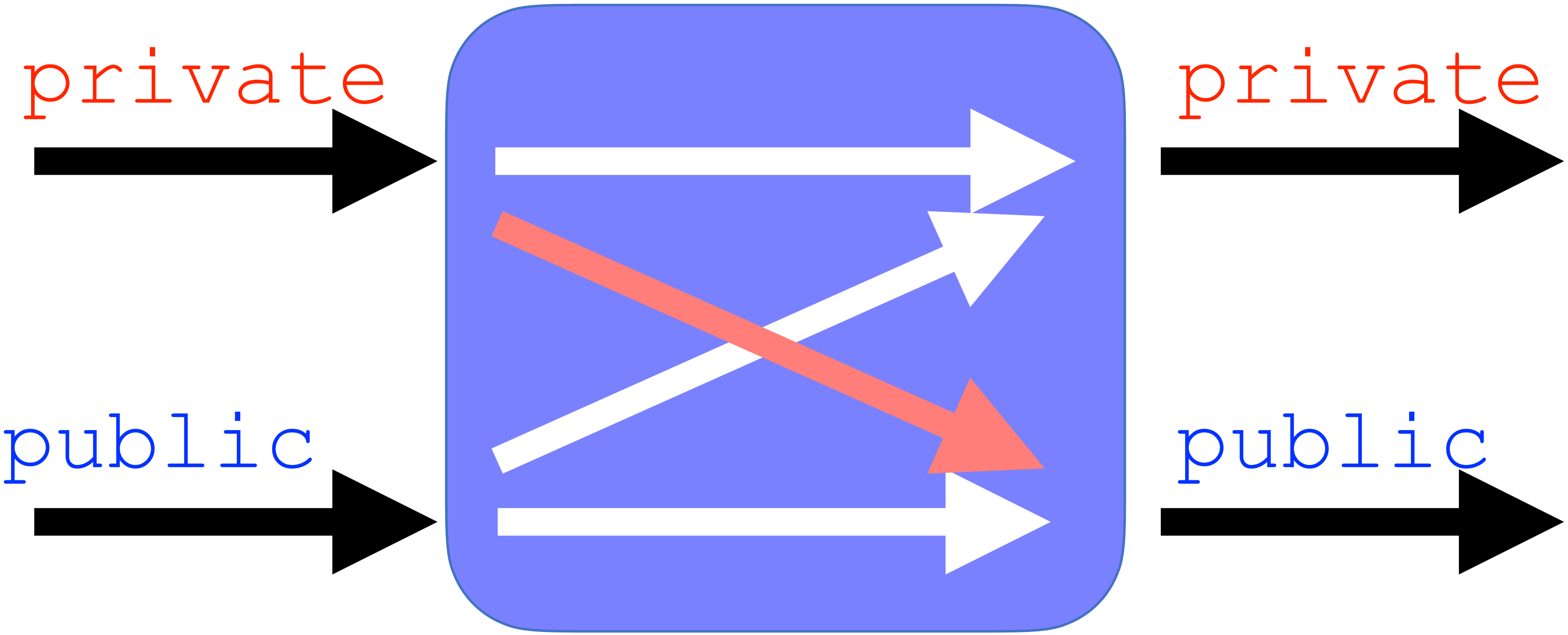
Wigderson Named Turing Awardee for Decisive Work on Randomness

By Neil Savage

Posted Apr 10 2024

Quantitative Information Flow Control

We want to quantify the confidential information that leaks in what is considered nonconfidential.



Quantitative Information Flow Control

Quantitative information flow has been used for:

- Analyzing distributed protocols and scheme,
- Analyzing side-channel vulnerabilities and preventions.
- Analyzing crypto protocols,
- Analyze election protocols
- Analyze differential privacy mechanisms
- …

Guessing Game

- The adversary has some prior π_R on R and it updates it after seeing U.

Shannon Entropy

$$H(X) = \sum_{x \in \mathcal{X}} \Pr[X = x] \log\left(\frac{1}{\Pr[X = x]}\right) = \mathbb{E}\left[\log\left(\frac{1}{\Pr[X = x]}\right)\right]$$

- uncertainty about X
- expected amount of information gain by observing the value of the random variable,
- average number of bits required to transmit X optimally

Conditional Entropy

$$H(X \mid Y) = \sum_{y} \Pr[Y = y] \cdot \sum_{x \in \mathcal{X}} \Pr[X = x \mid Y = y] \log\left(\frac{1}{\Pr[X = x \mid Y = y]}\right)$$

- If C is constant H(R|U)=H(R).
- If C is non constant and deterministic H(R|U)=0, so:

$$\texttt{Leakage}(U) = H(R)$$

Information leakage

Information leaked =
initial uncertainty - remaining uncertainty

- Which could be

$$\texttt{Leakage}(U) = H(R) - H(R \mid U)$$

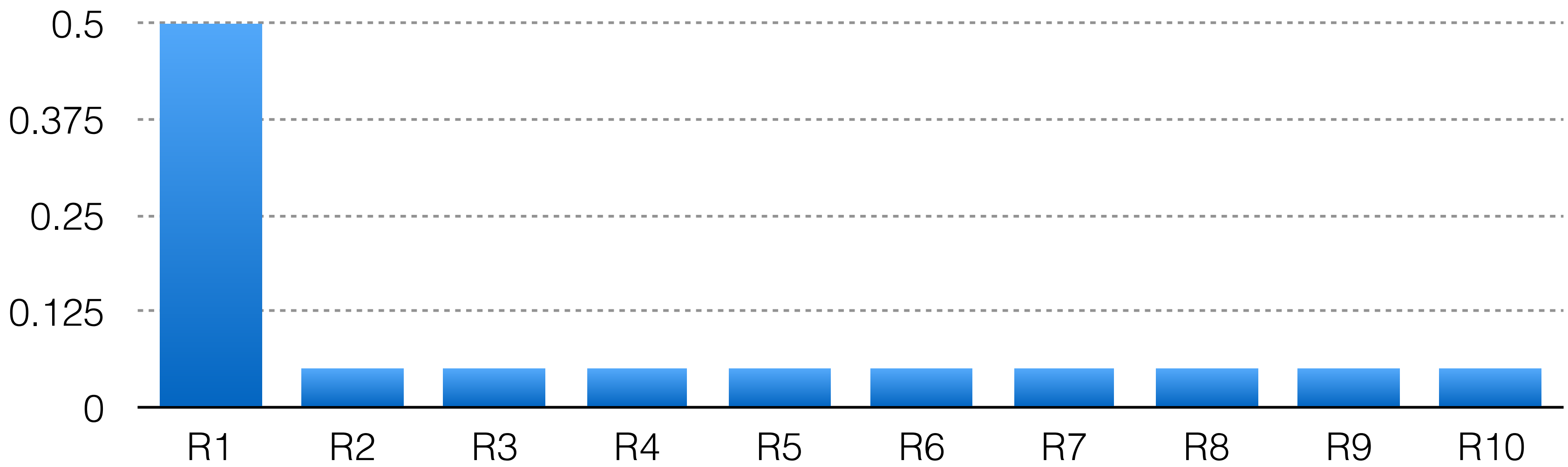
- This is the mutual information between R and U

Shannon Entropy

We could think that:

"If a secret X has distribution π, then an adversary's probability of guessing the value of X correctly in one try is at most $2^{-H(\pi)}$"

- This is false. E.g. for this distribution $H(\pi) \sim 2.44$, and $2^{-H(\pi)} \sim 0.18$



Same issue on conditional entropy

- Assume that R is a uniformly distributed 8k-bit integer with range $0 \leq R < 2^{8k}$, where $k \geq 2$. Hence $H(R) = 8k$.

- Consider these two programs:

```
if R mod 8 = 0 then U:= R else U := 1
```

And

$$U := R \;\&\&\; 0^{7k-1}1^{k+1}$$

- In both cases $H(R|U) \sim 7k-1$ suggesting that the number of guesses needed to guess R is $2^{-(7k-1)}$

Bayes Vulnerability

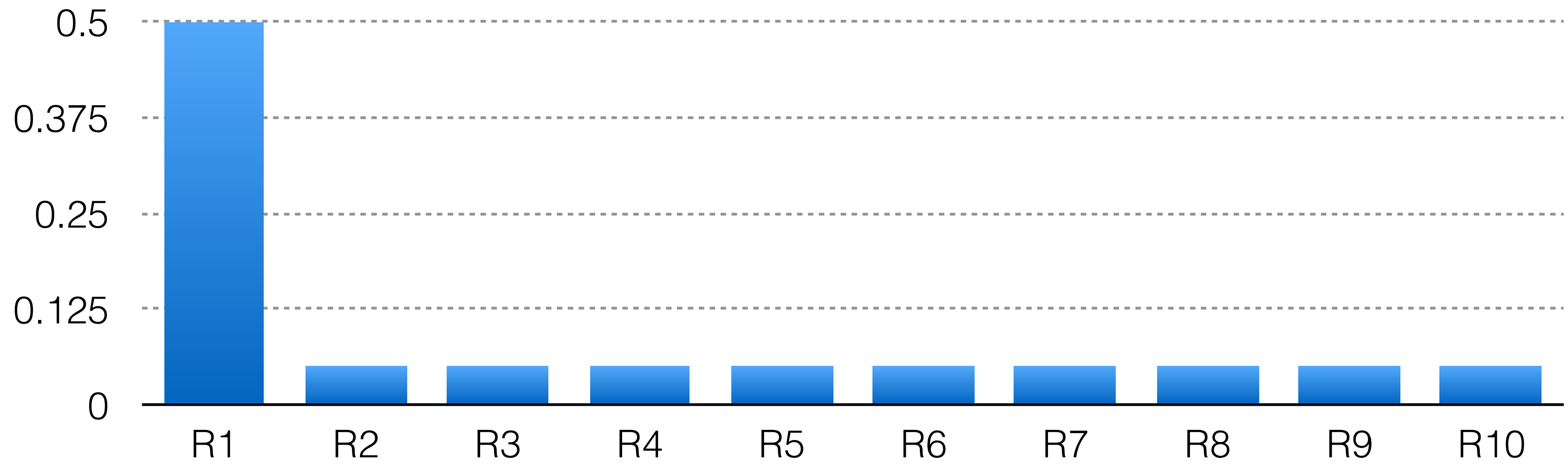
$$V(X) = \max_{x \in \mathcal{X}} \Pr[X = x]$$

- In our case it is the max probability assigned by the prior π_R.
- Best choice for a rational adversary to guess the secret in one try.

Bayes Vulnerability examples

$$V(X) = \max_{x \in \mathcal{X}} \Pr[X = x]$$

- Consider π_R to be a uniform distribution over n outcomes. Then, $V(\pi_R)=1/n$
- Consider π_R to be the following distribution again, we have $V(\pi_R)=.5$



Min Entropy

- We can use Bayes vulnerability to define a notion of entropy.

$$H_{\min}(X) = \log \frac{1}{V(X)}$$

- This is actually known as min entropy, and it can be seen as the greatest lower bound of the information content in bits of observations of X.

Conditional Min Entropy

- We can have a conditional version of the previous notions

$$H_{\min}(X \mid Y) = \log \frac{1}{V(X \mid Y)}$$

- Where

$$V(X \mid Y) = \sum_{y \in \mathcal{Y}} \Pr[Y = y] \max_{x \in \mathcal{X}} \Pr[X = x \mid Y = y]$$

Information leakage v2

Information leaked =
initial uncertainty - remaining uncertainty

- Which could be

$$\texttt{Leakage}(U) = H_{\min}(R) - H_{\min}(R \mid U)$$

Bayes vulnerability and min entropy

We have:

$$V(R \mid U) = 2^{H_{\min}(R|U)}$$

- The expected probability that the adversary could guess R given U decreases exponentially with $H_{min}(R|U)$.

Conditional Min Entropy

- Assume that R is a uniformly distributed 8k-bit integer with range $0 \leq R < 2^{8k}$, where $k \geq 2$. Hence $H(R) = 8k$.

- Consider these two programs:

```
if R mod 8 = 0 then U:= R else U := 1
```

And

$$U := R \;\&\&\; 0^{7k-1}1^{k+1}$$

- For the first we have $H_{min}(R|U) \sim 3$ while for the second is still $H_{min}(R|U) \sim 7k-1$.

Conditional Min Entropy

- Assume that R is a uniformly distributed 8k-bit integer with range $0 \leq R < 2^{8k}$, where $k \geq 2$. Hence $H(R) = 8k$.
- Consider these two programs:

```
if R mod 8 = 0 then U:= R else U := 1
```

And

$$U := R \,||\, 0^{8k-3}1^3$$

- For both of them we have $H_{min}(R|U) \sim 3$.

Is this reasonable?

Can we have a more general approach?

Gain function

- Suppose we have a set of secrets **X** and a set of actions **W**, then a gain function g is a function of type:

$$g : \mathbf{X} \times \mathbf{W} \rightarrow \mathbb{R}$$

- We can think about g as a scoring function for actions on a secret

Gain function

- Suppose we have a set of secrets **X** and a set of actions **W**, then a gain function g is a function of type:

$$g : \mathbf{X} \times \mathbf{W} \rightarrow \mathbb{R}$$

- We can think about g as a scoring function for actions on a secret

We could have a similar definition based on losses.

g-Vulnerability

$$V_g(X) = \max_{w \in \mathcal{W}} \sum_{x \in \mathcal{X}} \Pr[X = x] \cdot g(w, x)$$

- The best action for a rational adversary is the one that maximizes the expected gain.

g-Vulnerability example

Example 3.3 With $\mathcal{X} = \{x_1, x_2\}$ and $\mathcal{W} = \{w_1, w_2, w_3, w_4, w_5\}$, let gain function g have the (rather arbitrarily chosen) values shown in the following matrix:

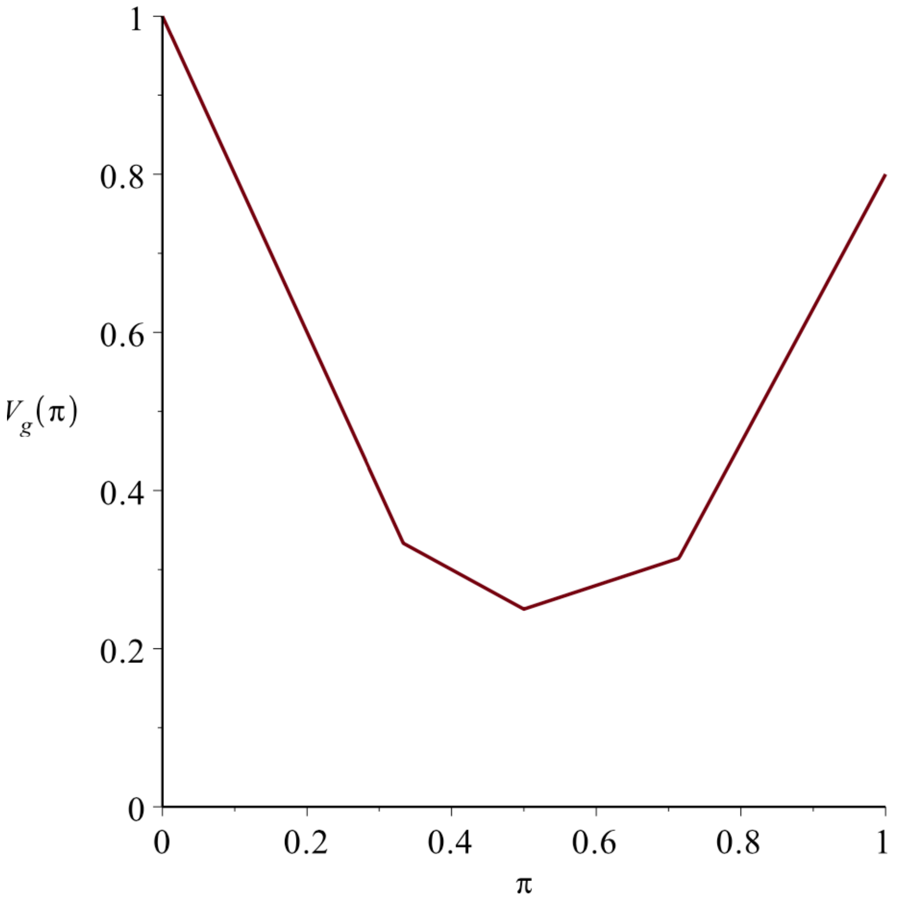
G	x_1	x_2
w_1	-1.0	1.0
w_2	0.0	0.5
w_3	0.4	0.1
w_4	0.8	-0.9
w_5	0.1	0.2

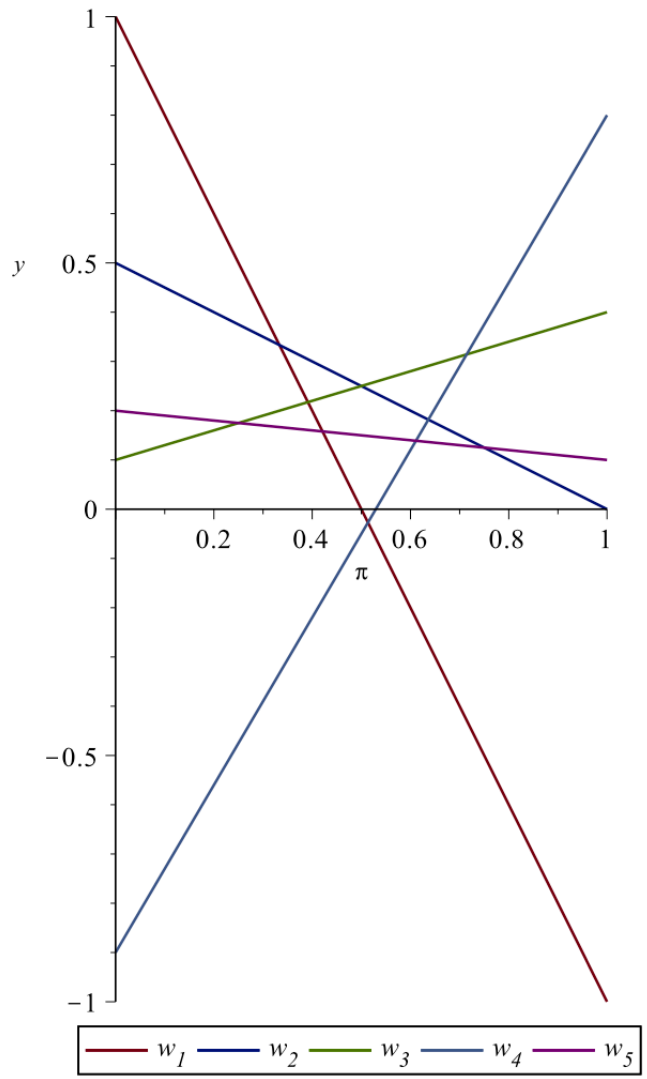
To compute the value of V_g on (say) $\pi = (0.3, 0.7)$, we must compute the expected gain for each possible action w in $\mathcal{W}$, given by the expression $\sum_{x \in \mathcal{X}} \pi_x \, g(w, x)$ for each one, to see which of them is best. The results are as follows.

$$
\begin{array}{lcccc}
\pi_{x_1} g(w_1, x_1) + \pi_{x_2} g(w_1, x_2) & = & 0.3 \cdot (-1.0) + 0.7 \cdot 1.0 & = & 0.40 \\
\pi_{x_1} g(w_2, x_1) + \pi_{x_2} g(w_2, x_2) & = & 0.3 \cdot 0.0 + 0.7 \cdot 0.5 & = & 0.35 \\
\pi_{x_1} g(w_3, x_1) + \pi_{x_2} g(w_3, x_2) & = & 0.3 \cdot 0.4 + 0.7 \cdot 0.1 & = & 0.19 \\
\pi_{x_1} g(w_4, x_1) + \pi_{x_2} g(w_4, x_2) & = & 0.3 \cdot 0.8 + 0.7 \cdot (-0.9) & = & -0.39 \\
\pi_{x_1} g(w_5, x_1) + \pi_{x_2} g(w_5, x_2) & = & 0.3 \cdot 0.1 + 0.7 \cdot 0.2 & = & 0.17
\end{array}
$$

Thus we find that w_1 is the best action and $V_g(\pi) = 0.4$. $\square$

g-Vulnerability example





Interesting gain functions

- Identity gain function: $g(w,x)=1$ if $x=w$ and 0 otherwise.
- Gain functions induced by a metric d: $g(w,x)=d(w,x)$
- Binary gain functions $g(w,x)=1$ if $x \in w$ and 0 otherwise.
- Penalty gain functions $g(w,x)=1$ if $x=w$, 0 if $w=\perp$, -1 otherwise.
- Loss functions $l(w,x)=-\log(w(x))$ where w is a distribution

Gain function properties

- We can show that for every gain function g, the g vulnerability V_g is a convex function.

- Algebraic structure on gain functions translate to algebraic structure on the associated g-vulnerability.

$$V_{g\times k}(X) = k \times V_g(X) \qquad \text{for } k \geq 0$$

$$V_{g+r}(X) = V_g(X) + r$$

Information leakage v2

Information leaked =
initial uncertainty - remaining uncertainty

Channel

- We can abstract programs over finite data types c to stochastic matrices.

C	y_1	y_2	y_3	y_4
x_1	$1/2$	$1/2$	0	0
x_2	0	$1/4$	$1/2$	$1/4$
x_3	$1/2$	$1/3$	$1/6$	0

- where C_{xy}=Pr[c(X)=y|X=x]

Bayes Theorem

$$\Pr[x \mid y] = \frac{\Pr(y \mid x)\Pr(x)}{\Pr(y)}$$

- We can use Bayes' theorem and a channel to compute the posterior given a prior.

Posteriors

Given $\pi = [\frac{1}{3}, \frac{1}{3}, 0, \frac{1}{3}]$ And

C	y_1	y_2	y_3	y_4
x_1	$1/2$	$1/6$	$1/3$	0
x_2	0	$1/3$	$2/3$	0
x_3	0	$1/2$	0	$1/2$
x_4	$1/4$	$1/4$	$1/2$	0

We can compute the joint channel:

J	y_1	y_2	y_3	y_4
x_1	$1/6$	$1/18$	$1/9$	0
x_2	0	$1/9$	$2/9$	0
x_3	0	0	0	0
x_4	$1/12$	$1/12$	$1/6$	0

And with this, renormalizing:

	$p_{X\|y_1}$	$p_{X\|y_2}$	$p_{X\|y_3}$
x_1	$2/3$	$2/9$	$2/9$
x_2	0	$4/9$	$4/9$
x_3	0	0	0
x_4	$1/3$	$1/3$	$1/3$

Hyper-distribution

Consider this set of posteriors

	$p_{X\|y_1}$	$p_{X\|y_2}$	$p_{X\|y_3}$
x_1	$^2/_3$	$^2/_9$	$^2/_9$
x_2	0	$^4/_9$	$^4/_9$
x_3	0	0	0
x_4	$^1/_3$	$^1/_3$	$^1/_3$

We could think about it as a distribution over posteriors

$[\pi \triangleright \mathsf{C}]$	$^1/_4$	$^3/_4$
x_1	$^2/_3$	$^2/_9$
x_2	0	$^4/_9$
x_3	0	0
x_4	$^1/_3$	$^1/_3$

This is what we call a hyper-distribution, read as π through C.

Hyper-distribution

$[\pi \triangleright C]$	$1/4$	$3/4$
x_1	$2/3$	$2/9$
x_2	0	$4/9$
x_3	0	0
x_4	$1/3$	$1/3$

We can write a hyper-distribution as:

Outer probabilities

$$[\pi \triangleright C] = \sum_i a_i [\delta^i]$$

Inner probabilities

Abstract channels

We can think about channels as essentially mapping priors to hyper-distributions.

The abstract channel **C** of a channel C is the mapping:

$$\pi \rightarrow [\pi \triangleright C]$$

We can think about this as the semantics of C

$$[[C]] = \lambda\pi\,.\,[\pi \triangleright C]$$

We can write a hyper-distribution as:

$$[\pi \triangleright C] = \sum a_i[\delta^i]$$

Properties

- C satisfies non-interference if its abstract channel is a lifting:

$$[[C]] = \lambda\pi\,.\,\texttt{unit}\,\pi$$

- We can identify canonical forms for abstract channels and characterize abstract channels properties through properties about their functions.

- We can also take convex combinations of abstract channel and compose them in other abstract channels.

Posterior g-Vulnerability

$$V_g[\pi \triangleright C] = \sum_i a_i V_g(\delta^i)$$

Assuming

$$[\pi \triangleright C] = \sum_i a_i \delta^i$$

- Expected value of g-vulnerabities.

g-Vulnerability example

Example 3.3 With $\mathcal{X} = \{x_1, x_2\}$ and $\mathcal{W} = \{w_1, w_2, w_3, w_4, w_5\}$, let gain function g have the (rather arbitrarily chosen) values shown in the following matrix:

G	x_1	x_2
w_1	-1.0	1.0
w_2	0.0	0.5
w_3	0.4	0.1
w_4	0.8	-0.9
w_5	0.1	0.2

To compute the value of V_g on (say) $\pi = (0.3, 0.7)$, we must compute the expected gain for each possible action w in $\mathcal{W}$, given by the expression $\sum_{x\in\mathcal{X}} \pi_x\, g(w,x)$ for each one, to see which of them is best. The results are as follows.

$$
\begin{array}{rclcl}
\pi_{x_1}g(w_1,x_1) + \pi_{x_2}g(w_1,x_2) & = & 0.3\cdot(-1.0) + 0.7\cdot 1.0 & = & 0.40 \\
\pi_{x_1}g(w_2,x_1) + \pi_{x_2}g(w_2,x_2) & = & 0.3\cdot 0.0 + 0.7\cdot 0.5 & = & 0.35 \\
\pi_{x_1}g(w_3,x_1) + \pi_{x_2}g(w_3,x_2) & = & 0.3\cdot 0.4 + 0.7\cdot 0.1 & = & 0.19 \\
\pi_{x_1}g(w_4,x_1) + \pi_{x_2}g(w_4,x_2) & = & 0.3\cdot 0.8 + 0.7\cdot(-0.9) & = & -0.39 \\
\pi_{x_1}g(w_5,x_1) + \pi_{x_2}g(w_5,x_2) & = & 0.3\cdot 0.1 + 0.7\cdot 0.2 & = & 0.17
\end{array}
$$

Thus we find that w_1 is the best action and $V_g(\pi) = 0.4$. □

Posterior g-Vulnerability example

Let's consider this channel

C	y_1	y_2
x_1	0.75	0.25
x_2	0.25	0.75

With prior (0.3,0.7) we get:

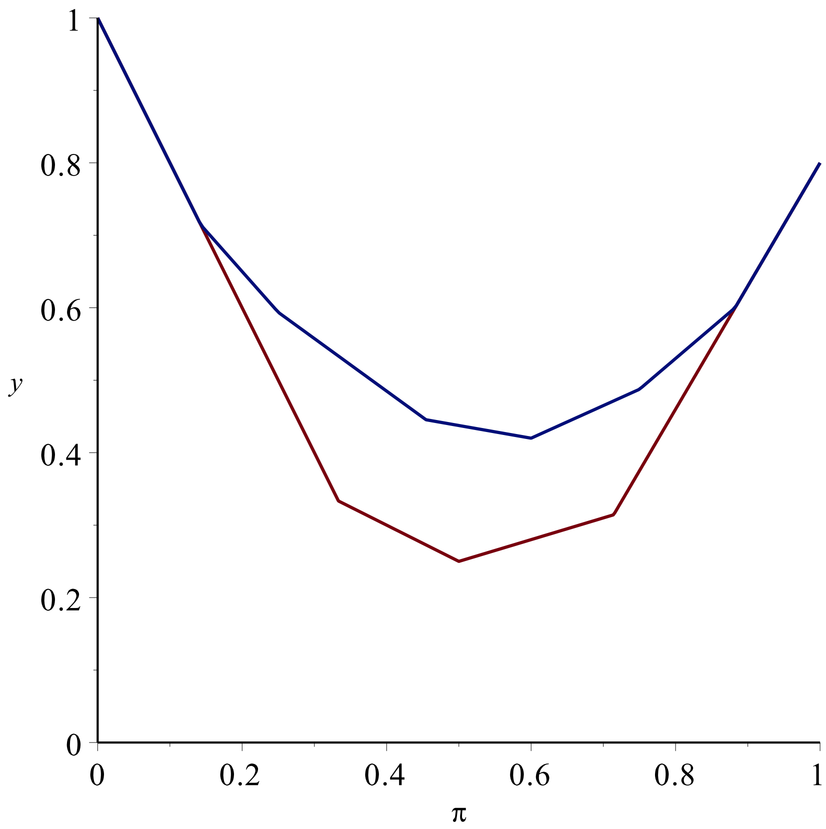
a_1=.4

a_2=.6

δ^1=(0.5625,0.4375)
δ^2=(0.5625,0.4375)

$$V_g[\pi \triangleright C] = 0.5575$$

Posterior g-Vulnerability example



Comparison of $V_g(\pi)$ (red) and $V_g[\pi \triangleright \mathsf{C}]$ (blue) for

Many other topics

- How to apply it in practical analyses
- How to use program logics to reason about this framework
- Geometric properties
- Stochastic properties
- …