note: this is still very much a WIP so read if you're interested but don't expect well-formed questions

In a previous post I outlined a question, trying to understand how much information could be leaked by allowing for certain kinds of computation (and feedback from the computation) on an encrypted transaction. I recommend reading the first post before you read this one.

Adding Economics

In part I, we asked how feedback from computation on private information can reduce an adversary's uncertainty around what the private information given that the information was a realisation of a distribution. The problem formulation, however, does not consider that there is some structure to the support of the distribution. To see this, let's look at two "equivalent" examples. Let's say an adversary has a belief that:

- the private information is a trade of 10 units between asset A and B with a 50% chance of being a buy and 50% chance of being a sell.
- the private information is a trade buying asset A with asset B and there is a 50% probability that the quantity being traded is 10 units and 50% probability that it is 1000 units.

Strictly from a probabilistic standpoint (depending on the priors), these two posterior beliefs could constitute equal amounts of information. However, from the perspective of the adversary these cases are not equivalent since we expect the adversary to act on this information. Uncertainty in the adversary's beliefs does not always translate to uncertainty in the adversary's outcomes. For example, it may be the case that the adversary's goal is to always try to replicate the trade, they seem worse off in the first case because an erroneous guess constitutes the "opposite" of the desired outcome wheres an error in the second case would still be directionally correct.

The Refined Setting:

Assume the same setup as before:

An adversary is given a finite set of functions \mathcal{F}

where every element f_i\in\mathcal{F}

is a mapping from a set X

to a set L

. The adversary is also given a prior distribution, \mathcal{D}

, over F

.

The game

A function f^*

is drawn according to \mathcal{D}

. The adversary does not know f^*

.

Now there are n

rounds. In each round i

, the adversary selects any x_i\in X

and is told f^*(x_i)

.

After the n

'th round, the adversary attempts to guess f^*

.

On top of this, we now introduce:

- a payoff function, p

, for the adversary so that their objective is to guess f'

in order to maximise p(f',f^*)

.

- a payoff function, q

, for the “sender” of the information. Thus we can measure the outcome of the game by q(f',f^*)

Now we have p

which effectively allows us to classify posterior distributions (each distribution maps on to some f'

assuming the adversary can find the optimal f'

). Additionally, q

provides a way for us to measure how much we actually care about the information being exposed. We can now phrase our question of how impactful the information leakage from optimal querying is in terms of the impact on q

. We can even define q

in such a way that some information leakage may actually be beneficial (corresponding to something like luring liquidity to trade into).