Extracting information

Paul Christiano Follow
Oct 3, 2016 · 12 min read

Inspired by a discussion at MIRI. Related: how to buy a truth from a liar;

(*Note: the original formalization of the problem in this post was solved by Harrison Klaperman and Nisan Stiennon. I've included their solution, and a formalization of a more challenging problem, but I'm not sure whether the more challenging problem is actually very much more challenging*.)

Suppose that I am interested in estimating a quantity X (e.g. I want to estimate how happy I would be if a particular policy were adopted). For concreteness, suppose that I want to minimize the squared error of my estimate.

I have access to a large set of experts. If I give the experts some resources, they can spend them to uncover facts that are related to X. For example, they can decide to look up a relevant fact or search for a relevant argument.

Let's suppose further that **I can verify any facts uncovered by the experts** (motivated by an application to informed oversight). I don't understand the landscape of possible facts, and I can't find them myself. But when an expert points out an argument, or shows me the result of a calculation, I know enough about the subject that I can update my beliefs about X. In fact, the experts are no better-informed about X than I am—their only advantage is that they have the ability to gather possible facts, and understand which facts are available to be discovered.

If the experts were motivated to be helpful, then the experts would follow some optimal (adaptive) fact-gathering strategy, given the available resources, and report every fact they discovered. This is our **benchmark scenario.**

In fact the experts aren't motivated to be helpful. Instead, I am free to choose their reward, and their goal is to maximize their expected reward.

My question: is there any way that we can incentivize the experts to be almost as helpful as in the benchmark scenario? Can we at least get close?

Throughout, I am going to assume that I know how to **update naively** on the facts presented by the experts, but I cannot update accurately on which facts they choose to present, because I don't understand what the landscape of possible facts looks like or what tradeoffs the experts face.

(I give one formalization of this problem in an appendix at the end of the post.)

Motivation

Normally in machine learning we train a learning system by comparing its behavior to some ground truth. But eventually we will want to use machine learning to expand the frontiers of our understanding, and in these cases we don't have any external ground truth.

In these settings, we could try and train reinforcement learning systems to identify and report relevant considerations. But it seems like we don't yet know how to find an objective that would actually lead to efficient information-gathering, or even that would result in us arriving at a "correct" estimate.

I think this problem is quite natural. And though today we are mostly happy with building machine learning systems that imitate human performance or that make short-term predictions, I think that in the long run it will be extremely important to figure out how to use machine learning to push beyond human abilities and to answer questions about what we *ought to do*, questions that don't correspond to short-term predictions.

A satisfactory solution to this problem would also play a direct role in an act-based approach to AI control. In some sense it is a warm-up for the informed oversight problem, though it seems like it is the more fundamental problem and might well represent the main difficulty.

Some inadequate solutions

Argument

We could pick two experts, designate one of them the up expert and the other the down expert. The up expert's reward is my final estimate of X, and the down expert's reward is my 1 – my final estimate of X.

This scheme works perfectly when it is possible for the experts to gather all of the relevant facts (though there are many schemes that work perfectly in this case). In this case there is a "correct" estimate that I would have after updating on all of the relevant facts. To see that I reach the correct estimate, suppose that my estimate is wrong by being too low. Then the up expert could gather and report all of the relevant facts, causing my estimate to become correct, and earn a larger reward. On the other side, if my estimate is too high, the low expert can earn a larger reward by correcting it.

When it isn't possible to gather all of the information, this incentive system doesn't cause the experts to use their limited resources effectively.

For example, suppose that there are a large number of noisy indicators of X. After looking into an indicator, an expert will know whether it is favorable or unfavorable to them, and then report it only if it is favorable.

Assume that each expert can either get 10,000 estimates each of which will shift up or down my estimate by 0.0001, or they can get a single estimate which will shift up or down my estimate by 0.1. From the perspective of the experts, getting the 10,000 estimates is way better: they can report the 5,000 that point in their preferred direction, shifting my beliefs by 0.5 in their preferred direction. But from the perspective of efficiently gathering information, the "big" estimate is much more informative (since its variance is $0.1^2 = 0.01$, vs. $10{,}000 * 0.0001^2 = 0.0001$).

In general, I don't see any reason that this argument scheme should cause me to arrive at correct beliefs, and it seems clear that it incentivizes inefficient information-gathering. It would be nice to have something more satisfying.

Buying truth from a liar

Katja Grace suggests the following scheme:

- Allow the expert to tell you facts one at a time.
- Every time your beliefs change, give the expert a reward of 1.

Katja argues that you must end up with the same beliefs as the expert: otherwise, the expert could earn an additional reward by revealing some additional facts.

I like this scheme a lot. But especially if we want to incentivize the expert to gather information strategically, it seems like there are some problems.

First we need to clarify the scheme: what exactly is the payoff as a function of your beliefs?

- The absolute value of the change: |estimate(t) – estimate(t+1)|
- You could pay every time your estimate crosses some threshold, say 0.5. This isn't really appropriate for the case where we want an estimate, but is closely analogous to the absolute value.
- The squared value of the change: $(\text{estimate(t)} - \text{estimate(t+1)})^2$

A big problem with using absolute value is that you can get arbitrarily large payoffs by breaking a piece of info into very tiny pieces.

For example, suppose that I am interested in the value of a Brownian motion at time 1, B(1). Given the values B(*t*) for *t* ≤ T < 1, the expectation of B(1) is precisely B(T). I can break the interval into a bunch of pieces 0 =t1 < t2 < … < tN = 1, and reveal B(t1), B(t2), B(t3), …, B(tN). Then the total absolute movement is:

- |B(t2) – B(t1)| + |B(t3) – B(t2)| + … + |B(tN) – B(tN-1)|

Unfortunately, this quantity approaches infinity as we use smaller and smaller subdivisions. So the expert can earn arbitrarily large reward. Of course in practice there will be limits on the data that they can present and so they maybe can't earn arbitrarily large rewards, but their incentive is still to find some information that can be effectively broken up into very tiny pieces. In fact they can earn arbitrarily large rewards without revealing almost any total evidence and so it seems unlikely that they can be induced to reveal everything.

With squared rewards, we can do a little bit more work to arrive at a different problem. Suppose that again we have a Brownian motion B and we are interested in B(1). Then a clever expert can find a large number of subintervals [t1,t2], [t3, t4], …, [tN-1, tN], over which the Brownian motion trended upwards, and report the sum:

- S = B(t2) – B(t1) + B(t4) – B(t3) + … + B(tN) – B(tN-1)

Conditioned on this fact, the expected value of B(1) is S. But in the limit of arbitrarily fine divisions, we can take S to be arbitrarily large. (And if the motion were limited to an interval, the expert could cause our view to bounce from 0 to 1 and back again arbitrarily many times.)

Once again, it's not exactly clear what the expert's incentives are, but they certainly cannot induce optimal information-gathering and honest reporting.

Minimizing uncertainty

After hearing all of the experts' arguments, I can have some subjective probability distribution over the quantity X. Given that my goal is to have minimal squared error, I could just compensate the expert directly based on the variance of my subjective distribution.

To see that this fails, suppose that there are two possible worlds. In one, X is zero. In the other, it is normally distributed with variance 1. The expert can acquire evidence about which world we are in. The expert's main motive is to convince the judge that they are probably in the zero-variance world, and they can selectively present evidence to prove their point.

We can try to improve the situation by adding a second expert whose goal is to increase the judge's uncertainty, but this just puts us back in the "argument" setting described above.

The sophisticated listener

So far I've assumed that the judge does not update on the observation that the expert chosen to present a certain fact.

If the judge has accurate beliefs about the landscape of possible facts, and the tradeoffs faced by the experts, then in some sense our problem has a trivial solution: the judge can calculate the optimal information-gathering strategy, and ask the expert to follow that strategy and then report everything they learn. The judge can then check for themselves whether the expert followed the optimal strategy.

I want to avoid these solutions because they seem unworkable in most cases. For example, there is not much reason to apply machine learning to a problem if you already know exactly what strategy is optimal (except to more quickly execute the human-level information-gathering policy, which is explicitly not the case we are interested in). I think that this restriction is captured well by the assumption that the user doesn't understand the landscape of possible facts or the expert's options.

Conclusion

I think this is one of the cleanest open questions with a strong claim to relevance for AI control.

I have no idea whether this problem is possible. If it's impossible, finding a proof would be great. We would then have to evaluate whether the intuitive desideratum is unsatisfiable or whether there might be a stronger model in which we can find a solution.

Appendix A: a precise formal model

We could give several possible complete formal models. Here is one.

We are given a set of boxes and a joint distribution P over the contents of all of the boxes and some latent parameter $X \in [-1, 1]$.

There are a large number of independent experts, who have the ability to open boxes and observe their contents. There is a family $\mathscr{F}$ of "feasible" subsets that controls which boxes the expert can open.

Each expert will choose a set of boxes $A \in \mathscr{F}$ and observe the contents of those boxes. They will then pick a subset $B \subseteq A$. We observe each of these sets B, as well as the contents of the corresponding boxes.

The non-adaptive version of the problem has been solved (see the next appendix); to make things harder we can also consider an adaptive version.

- **Adaptive version**: Experts sequentially choose boxes and observe their contents; this may be interleaved in any way with their interaction with the mechanism and with each other. Experts may reveal the contents of any box they have opened to the mechanism at any time. An expert cannot open a box if it would mean that they have opened a set of boxes $A \notin \mathscr{F}$.

A mechanism is a function which takes as input:

- A joint distribution P over box contents and X,
- one set S for each expert, and
- the contents of each box in any of the sets S.

And outputs a reward for each expert together with a guess for X. Note that a mechanism does not know what sets of boxes are feasible; the identity of $\mathscr{F}$ is hidden. And of course the mechanism does not get to observe the contents of the unopened boxes.

- **Adaptive version**: we also allow extensive form mechanisms, which have boxes revealed to them one at a time, may send messages to experts after any box is revealed (for example with info about the boxes revealed by other experts) and only output rewards+guesses at the end of the process.

Given a mechanism, the experts will choose a policy that maximizes their expected reward, i.e. they play a Nash equilibrium of the resulting game.

Our loss is equal to the expected squared error of the mechanism's guess. That is, we sample X and the contents of the boxes from P and run the mechanism and compute its output; our loss is the expected squared error of that output. This depends both on the mechanism and on the choice of equilibrium.

Note that in contrast with traditional mechanism design, we don't care at all about the rewards that we assign to the experts, we only care about the accuracy of our estimate.

We compete with the loss of the following benchmark strategy, where we are allowed to just open the boxes ourselves:

- We open the set of boxes A∈$\mathscr{F}$ such that the expected variance of X conditioned on the contents of A is minimal.
- We output the expectation of X conditioned on the contents of the boxes in a.

(Note that in the benchmark we are only able to open a single set of boxes, but in the actual scenario there are many experts each of whom can open a set of boxes. So in some sense the benchmark isn't "the best we could hope for," but I think it is still a good first target to shoot for.)

- **Adaptive version**: A *policy* for an expert is a function which describes which box to next open, given the set of boxes that have been opened so far + the contents of those boxes. We can run such a policy in a natural way, starting from "no boxes have been opened" and ending when the expert tries to open a box that would be infeasible. The quality of a policy is the expected variance of X conditioned on the contents of all boxes that have been opened at the end. For a fixed family $\mathscr{F}$ and distribution P, we can evaluate the quality of each policy and pick the highest-quality policy. This is our benchmark.

The question is: can we find a mechanism such that, for all P and all $\mathscr{F}$ and all Nash equilibria of the resulting game, our loss is at most the loss of the benchmark policy? If not, how small can we make the worst-case gap between the benchmark and the mechanism's output?

Appendix B: a mechanism for the non-adaptive case

Nisan Stiennon and Harrison Klaperman proposed the following solution in the non-adaptive case:

- Given a set of boxes A, pay the expert the *expected* variance of X conditioned on the contents of those boxes. Ignore the actual contents of the boxes.

This obviously works, and moreover is clearly the natural+correct solution to the problem as posed.

I think that the general idea of ignoring the actual data given, and just reasoning about how useful the information is expected to be, is probably a generally useful idea. It can probably be applied to some extent in the original setting of informed oversight, though there are some practical difficulties that were covered up by the simple formalization of the problem.

The adaptive version is a small step towards the full setting that we actually care about. I don't know whether there is a solution in the adaptive case that is analogous to the "ignore the boxes" solution in the non-adaptive case. It is clear that any solution will need to look at the actual contents of the boxes, since the "correct" behavior depends on the contents of the boxes and so it can't be incentivized while ignoring the contents of the boxes. But there may be some way to ignore the contents of each box when deciding how much to reward for revealing that box, while taking it into account when deciding how much to reward for future boxes.

One problem with both the adaptive and non-adaptive formalization is that they don’t handle “absence of evidence” well. For example, if a story may be plagiarized, we can imagine a box representing the “is it plagiarized” fact. This is a very important box to open if we want to estimate the originality of the story. And if the expert opens the box and finds that it is plagiarized, we should obviously reward them. But if they open the box and aren’t able to detect plagiarism, then it may be hard for them to prove that they opened the box and found it empty.

I’m not sure how to expand the model to account for this kind of asymmetry; there are also many other realistic complications that seem quite hard to fit into the model without making it a lot less clean.