Screening p-Hackers: Dissemination Noise as Bait*

Federico Echenique† Kevin He‡

First version: March 16, 2021
This version: February 10, 2022

Abstract

We show that adding noise to data before making data public is effective at screening p-hacked findings: spurious explanations of the outcome variable produced by attempting multiple econometric specifications. Noise creates "baits" that affect two types of researchers differently. Uninformed p-hackers who engage in data mining with no prior information about the true causal mechanism often fall for baits and report verifiably wrong results when evaluated with the original data. But informed researchers who start with an ex-ante hypothesis about the causal mechanism before seeing any data are minimally affected by noise. We characterize the optimal level of dissemination noise and highlight the relevant trade-offs in a simple theoretical model. Dissemination noise is a tool that statistical agencies (e.g., the US Census Bureau) currently use to protect privacy, and we show this existing practice can be repurposed to improve research credibility.

Keywords: p-hacking, dissemination noise, screening.

1 Introduction

In the past 15 years, academics have become increasingly concerned with the harms of p-*hacking*: researchers' degrees of freedom that lead to spurious empirical findings. For the observational studies that are common in economics, and other social sciences, p-hacking often takes the form of multiple testing: attempting many regression specifications on the same data with different explanatory variables, without an ex-ante hypothesis, and then selectively reporting the results that appear statistically significant. Such p-hacked results can lead to misguided and harmful policies, based on a mistaken understanding of the causal relationships between different variables. Recent developments in data and technology have also made p-hacking easier: today's rich datasets often contain a large number of covariates

*This research was made possible through the support of the Linde Institute. Echenique also thanks the NSF's support through the grants SES-1558757 and CNS-1518941. We are grateful for comments from Sylvain Chassang, Eva Jin, Albert Ma, Pascal Michaillat, Marco Ottaviani, Nathan Yoder, and the audiences at the University of Pennsylvania, Caltech, Universidad de la República, Boston University, and ASSA 2022.

†California Institute of Technology. Email: fede@hss.caltech.edu

‡University of Pennsylvania. Email: hesichao@gmail.com

that can be potentially correlated with a given outcome of interest, while powerful computers enable faster and easier specification-searching.

In this paper, we propose to use *dissemination noise* to address and mitigate the negative effects of p-hacking. Dissemination noise is purely statistical noise that is intentionally added to raw data before the dataset is made public. Statistical agencies, such as the US Census Bureau, already use dissemination noise to protect respondents' privacy. Our paper suggests that dissemination noise may be repurposed to screen out p-hackers. Noise can limit the ability of p-hackers to present spurious, but statistically significant results, as genuine causal mechanisms, and thus "game" standards of evidence. We show that the right amount of noise can serve as an impediment to p-hacking, while minimally impacting honest researchers who have an ex-ante hypothesis that they wish to test in the data.

1.1 p-Hacking

Spurious results in many areas of science have been ascribed to the ability of researchers to, consciously or not, vary procedures and models to achieve statistically significant results. The reproducibility crisis in Psychology has been blamed to a large extent on p-hacking.[1] For example, one representative study in Psychology (Open Science Collaboration, 2015) tried to replicate 100 papers, 97 of which reported statistically significant results. Only 36 of the results remained significant in the replication. As a response to p-hacking, one Psychology journal has taken the extraordinary step of banning the use of statistical inference altogether (Woolston, 2015). In a follow-up to the reproducibility crises in Psychology, Camerer et al. (2016) evaluate economic experiments. They too find a significant number of experiments that do not replicate.[2]

Most empirical work in economics and other social sciences are observational studies that use existing field data, and not experiments that produce new data. Observational studies lead to a different sort of challenge for research credibility. In the social sciences, p-hacking stems mostly from discretion in econometric specification and in choosing explanatory variables, rather than in experimental design and data generation. In experimental work, a remedy for p-hacking is available in the form of pre-registration: researchers must describe their methods and procedures before data is collected. Pre-registration limits researchers' degrees of freedom to generate a positive result. But pre-registration is problematic for observational studies because of an obvious credibility problem, as the data in question have already been made publicly accessible by statistical agencies (e.g. the US Census Bureau or the Bureau of Labor Statistics). Even if an economist pre-registers a methodological procedure and explanatory variables that are to be applied to existing data, one cannot rule out that they tried other procedures and variables on the same data before formulating the proposal.[3]

[1] Simmons, Nelson, and Simonsohn (2011) is the classic study that started off the reproducibility crisis in psychology. See also the follow-up Simonsohn, Nelson, and Simmons (2014), which coins the term p-hacking. An earlier influential paper Ioannidis (2005) touches upon some of the same issues, with a focus on the medical literature and using the term "data dredging."

[2] See also Camerer et al. (2018) and Altmejd et al. (2019). Imai, Zemlianova, Kotecha, and Camerer (2017) find evidence against p-hacking in experimental economics.

[3] Pre-registration in Economics remains very rare, even for field experiments that generate their own data

Our paper focuses on researchers who use an existing dataset. The dataset contains an outcome variable that can be potentially explained, in some statistical sense, using a large set of possible explanatory variables: what we call a "wide" dataset. In a wide dataset, the number of econometric specifications is large relative to the number of observations. This allows p-hackers to find, and pass as legitimate, statistical models that are spurious in reality.

1.2 Dissemination Noise

Dissemination noise is currently used by major statistical agencies to preserve privacy. The US Census Bureau, for instance, will only disseminate a noisy version of the data from the 2020 Census. They use the notion of *differential privacy*, an algorithm proposed and promoted in computer science to protect the privacy of individual census records while preserving the value of aggregate statistics.[4] The 2020 Census is not the first data product that the Bureau has released with noise. Previously, the Bureau also released a geographic tool called "On the map" whose underlying data was infused with noise, following the differential privacy paradigm. Even earlier technologies for preserving respondent confidentiality like swapping data and imputing data can also be interpreted as noisy data releases. The contribution of our paper is to propose a new use for dissemination noise.

1.3 Setup and Key Results

We consider a society that wants to learn the true cause behind an outcome variable. Researchers differ in their expertise: they may be a *maven* whose domain knowledge and theoretical background allow them narrow down the true cause to a small set of candidate causes, or a *hacker* who has no prior information about the true cause. Researchers derive utility from proposing a cause that explains the data well, and influences policy decisions. So uninformed hackers have an incentive to "game" the system by trying out different covariates, after seeing the data, to fish for one that appears correlated with the outcome.

In an environment where uninformed researchers can p-hack to game statistical conventions for policy implementation, we show that dissemination noise can help screen the researcher's expertise by introducing spurious correlations that can be proven to be spurious. These noise-induced correlations act like *baits* for p-hackers. The idea is simple: adding noise to the data creates potential spurious correlations that can be checked against the non-noisy version of the data. Of course, this comes at a cost. It makes the data less useful for the informed mavens who wish to use the data to test a specific ex-ante hypothesis derived from theory.

We explore this trade-off in our model, and analytically characterize the optimal level of dissemination noise. Our model, which is very simple, is meant to isolate the trade-off between screening hackers and hurting mavens. The key intuition for why dissemination noise can help address p-hacking is that *a small amount of noise hurts hackers more than mavens* (Lemma 2). All researchers act strategically to maximize their expected payoffs, but their optimal behavior differ. Mavens entertain only a small number of hypotheses, so

(Abrams, Libgober, and List, 2021).

[4]See https://www.census.gov/about/policies/privacy/statistical_safeguards.html and https://www2.census.gov/about/policies/2019-11-paper-differential-privacy.pdf

a small amount of noise does not interfere too much with their chances of detecting the truth. Hackers, by contrast, rationally try out a very large number of model specifications because they have no private information about the true cause behind the outcome variable. The hackers' data mining amplifies the effect of even a small amount of noise, making them more likely to fall for a bait and get screened out. So, a strictly positive amount of noise is optimal, and hackers get screened out precisely because they (rationally) p-hack. Moreover, we derive comparative statics on how the optimal level of noise varies with the fraction of hackers and with the size of the dataset.

One caveat is that it may not be credible to test researchers' findings on a raw dataset that is kept secret. For the sake of transparency, the statistical agency may be required to publish the dataset that is used to validate the reported causal specification and make policy decisions. This raises the question of whether we can keep screening out p-hackers if the same raw dataset must be reused to answer different research questions over time. To address this problem, we study a dynamic model with periodic noisy releases of an original dataset (see Section 5). In our dynamic model, a finding submitted for validation on February of 2022 is tested against the March 2022 release of noisy data. We show that it remains optimal to release data with a strictly positive amount of noise, but over time the hackers' access to all past data releases diminishes the effectiveness of noise. We show that it is optimal to eventually give up on noise, and return to a world in which p-hacking power is unchecked.

1.4 Alternative Solutions to p-Hacking

As already mentioned, the most common proposal to remedy p-hacking is pre-registration. This requires researchers to detail the analysis that they plan to carry out, before actually starting to analyze the data. One promising policy for journal publications is so-called "registered reports," whereby a paper is evaluated for publication based on the question it seeks to answer, and the methods it plans to use, before any results are obtained.[5] Pre-registration is a very good idea in many scientific areas, but it is of limited use for observational studies, which are ubiquitous in the social sciences. Not only does it preclude useful exploratory work, it is also impossible to audit or enforce because publicly available data can be privately accessed by a researcher before pre-registration.

A second solution is to change statistical conventions to make p-hacking more difficult. An extreme example is banning the use of statistical inference altogether (Woolston, 2015), arguably a case of throwing out the baby with the bath water. A less drastic proposal is contained in the collective manuscript Benjamin et al. (2018), which proposes to redefine the p-value threshold for statistical significance by an order of magnitude — from 5% to 0.5%. Of course this makes p-hacking harder, but a p-hacker armed with a sufficiently "wide" dataset and cheap enough computation power can discover spurious correlations that satisfy any significance threshold. We address this idea within our model (see Proposition 3 in Section 2.3) and argue that our proposed use of dissemination noise is largely complementary to requiring more demanding statistical significance.

An idea related to our proposal is simply to reserve data for out-of-sample testing. In fact, our baseline model can be understood as reserving all of the original raw data for out-

[5]Registered reports are used, for example, by Nature Human Behaviour.

of-sample testing, while the noisy data is released publicly. This approach differs from the usual out-of-sample testing procedure, where the raw data is partitioned in two portions. One portion is released publicly, and the rest is a "hold-out" dataset reserved for out-of-sample testing. We focus on a model of noise where each observation of each covariate is independently perturbed, which more closely resembles the kind of dissemination noise currently in use for privacy purposes. Our central message is that the current implementation of noise can be repurposed to screen out p-hacking.

The out-of-sample approach is the focus of Dwork et al. (2015a), who propose to give researchers free access to a portion of the data while only allowing them limited access to the portion of the data reserved for validation – the holdout data. By using tools from differential privacy, these authors ensure that the holdout can be re-used through controlling how much information about the hold-out dataset is leaked in each query. The work of Dwork et al. (2015a) is connected to our work, because differential privacy rests on adding noise to the data, but the mechanisms analyzed in their paper are very different from ours. There is no role in their proposal for the bait that we plant for p-hackers. We consider a world with two kinds of researchers and the dissemination noise here serves a screening role and aims to separate the two types who act strategically to maximize their expected payoffs.

1.5 Related Literature

There is an extensive literature outside of Economics documenting the prevalence and effects of p-hacking. We are not going to review this literature here.[6] In Economics, a series of papers seeks to understand the incentives and trade-offs behind p-hacking. Henry (2009); Felgenhauer and Schulte (2014); Felgenhauer and Loerke (2017); Di Tillio, Ottaviani, and Sørensen (2021); Henry and Ottaviani (2019) all study different games between a researcher (an agent) and a receiver (a principal). The agent has access to some p-hacking technology, which can amount to reporting a subset of results (only the positive results), or to stop sampling when they are ahead. These papers seek to better understand the interaction between p-hacking agents and their principals, and study how such interactions are affected by variations in the hacking technology. These papers do not consider the problem of expertise screening, which is our central focus. In our world, the principal's main problem is to provide sufficiently informative data to one type of agent (the maven) while distorting the data enough to mislead another type of agent (the hacker) who tries to make up for a lack of expertise with p-hacking. Another key difference is that in these other papers, the agent faces some cost from hacking. We instead consider hackers who incur zero cost from p-hacking, motivated by our focus on researchers who data mine an existing dataset (which is essentially free with powerful computers) instead of researchers who acquire new data at a cost. The equilibria in the papers in this literature would be uninteresting with free hacking. Our focus is instead on a specific intervention, dissemination noise, that can help screen out even very powerful p-hackers who face no hacking costs.

Di Tillio, Ottaviani, and Sørensen (2017) also studies a game between a p-hacker and a principal, but gives the agent some private information and the ability to select an area to

[6] For a casual read, see the Wired article "We are all p-hacking now" in the November 26 (2019) issue. For (literally!) an illustration, see xkcd # 882 (https://xkcd.com/882/).

do research in. This is a mechanism for hacking that is outside of the scope of our paper.

2 Model and Optimal Dissemination Noise

2.1 The Baseline Model

We propose a model that captures the essence of how dissemination noise allows for expertise screening in an environment where agents can p-hack, while keeping the model tractable enough to allow for analytic solutions. We discuss a number of extensions in Section 3, and show dissemination noise continues to help in these more general environments. Section 4 contains a numerical simulation that shows the idea and the basic trade-offs for using dissemination noise also hold in a more realistic empirical setting that is richer than the simple model where we derive our analytic results. Finally, we incorporate dynamic considerations in Section 5.

2.1.1 The Raw Dataset

Consider an environment where each unit of observation is associated with an outcome Y and a set A of potential causal covariates $(X^a)_{a \in A}$. The outcome and each covariate is binary. Suppose the dataset is "wide," so the set of potential causes for the outcome is large relative to the number of observations. In fact, we assume a continuum of covariates, so $A = [0, 1]$ (this assumption is relaxed in Section 3.3). For instance, the covariates may indicate the presence or absence of different SNPs in a person's genetic sequence, while the outcome refers to the presence or absence of a certain disease.

There is one covariate a^*, the *true cause*, that is perfectly correlated with Y, so $Y = X^{a^*}$. For instance, a^* is the one SNP that causes the disease in question. There is also a *red herring* covariate $a^r \in A$ with $Y = 1 - X^{a^r}$. The red herring represents a theoretically plausible mechanism for the outcome Y that can only be disproved with data. For instance, a^r might be a SNP that seems as likely to cause the disease as a^* based on a biological theory about the roles of different SNPs. For the baseline model, we analyze the easiest case where even a small amount of data can rule out the red herring, so we model X^{a^r} as perfectly negatively correlated with the outcome. We relax the assumption of perfect negative correlation in Section 3.2.

Nature draws the true cause a^*, and the red herring a^r, independently and uniformly from A. Then Nature generates the raw dataset $(Y_n, (X_n^a)_{a \in [0,1]})$ for observations $1 \leq n \leq N$. The N observations are i.i.d. given (a^*, a^r), with each Y_n equally likely to be 0 and 1, and X_n^a where $a \in A \backslash \{a^*, a^r\}$ being independent Bernoulli(1/2) variables that are also independent of Y_n. In other words, once a^* and a^r are drawn, we have fixed a joint distribution between Y and the covariates $(X^a)_{a \in A}$, and the raw dataset consists of N independent draws from this joint distribution. For instance, the dataset contains the complete genetic sequences of N individuals, and shows whether each person suffers from the disease in question.

2.1.2 Players and Their Incentives

There are three players in the model: a principal, an agent, and a policymaker. The *principal* owns the raw dataset, but lacks the ability to analyze the data and cannot influence policy-making norms. The principal disseminates a noisy version of the dataset, which we describe below. The *agent* uses the disseminated data to propose a policy, $\hat{a}$. Think of this as the agent proposing a cause for Y, which is then used in a decision about policy. Finally, a *policymaker* evaluates the agent's proposal on the raw dataset using an exogenous test.

The policymaker's role is purely passive, and restricted to deciding whether the agent's proposal passes the exogenous test. We say that *a passes* if the covariate X^a equals the outcome Y in all N observations, that is $Y_n = X_n^a$ for all $1 \leq n \leq N$, and that it *fails* otherwise. The policymaker will adopt a policy proposal if and only if it passes the test on the raw data. Passing the test is a necessary but insufficient condition for a to be the true cause of Y, for we could have $Y_n = X_n^a$ by random chance. So it is possible that a passing proposal still leads to a misguided policy, targeting an incorrect covariate $a \neq a^*$.

The agent is either a maven (with probability $1-h$) or a hacker (with probability h). These types of agents differ in their expertise. The maven knows that the true specification is either $Y = X^{a^*}$ or $Y = X^{a^r}$, and assigns them equal probabilities, but the hacker is ignorant about the realizations of a^* and a^r. The idea is that the maven uses domain knowledge (e.g., theory about the outcome Y) to narrow down the true cause to the set $\{a^*, a^r\}$. The hacker, in contrast, is completely uninformed about the mechanism causing Y.

The agent's payoffs reflect both a desire for reporting the true cause and a desire for policy impact. If the agent proposes a when the true cause is a^*, then his payoff is

$$w \cdot \mathbf{1}_{\{a=a^*\}} + (1-w) \cdot \mathbf{1}_{\{Y_n = X_n^a : 1 \leq n \leq N\}}.$$

Here we interpret $\mathbf{1}_{\{a=a^*\}}$ as the effect of proposing a on the agent's long-run reputation when the true cause a^* of the outcome Y eventually becomes known some years into the future. The other component $\mathbf{1}_{\{Y_n = X_n^a : 1 \leq n \leq N\}}$ models the agent's gain from proposing a policy that passes and is implemented by the policymaker. The relative weight $w \in [0,1]$ on these two components may differ for the two agent types. For our main results, we can have any $0 \leq w \leq 1$ for the hacker, but we need to assume $w > 1/2$ for the maven — that is, mavens care more about reporting the true cause than making a proposal that passes the test and gets implemented as policy.

The principal obtains a payoff of 1 if a true cause passes, a payoff of -1 if any other $a \neq a^*$ passes, and a payoff of 0 if the agent's proposal is rejected. The principal wants to maximize the positive policy impact of the research done on her data. A policy targeting the true cause is helpful and a misguided policy targeting any other covariate is harmful, relative to the default option of rejecting the proposal and implementing no interventions.

2.1.3 Dissemination Noise

The principal releases a noisy dataset $\mathcal{D}(q)$ by perturbing the raw data. Specifically, she chooses a *level of noise* $q \in [0, 1/2]$ and every binary realization of each covariate is flipped independently with probability q. So the noisy dataset $\mathcal{D}(q)$ is $(Y_n, (\hat{X}_n^a)_{a \in A})$, where $\hat{X}_n^a = X_n^a$ with probability $1-q$, and $\hat{X}_n^a = 1 - X_n^a$ with probability q. The principal's choice of q is common knowledge. A covariate a that is perfectly correlated with the outcome in $\mathcal{D}(q)$ but not in the original dataset, that is $\hat{X}_n^{(a)} = Y$ for every $1 \leq n \leq N$ but $X_n^{(a)} \neq Y$ for some n is called a *bait*.

The injection of noise in our model is motivated by the dissemination noise currently in use by statistical agencies, like the US Census Bureau. One could imagine other ways of generating a "noisy" dataset, such as selecting a random subset of the observations and making them fully uninformative, which corresponds to reserving the selected observations as a "hold-out" dataset for out-of-sample testing. Our analysis suggests that we can repurpose the existing practice of dissemination noise, which resembles perturbing each data entry independently, rather than withholding some rows of the dataset altogether.[7]

For some data-generating processes, adding noise to each entry is more sensible than fully withholding some observations. This happens when records are not independent realizations of a given statistical model. If the observations represent individuals on a social network, where neighbors influence each other, or time-series observation of some economic indicators, then it is not obvious how the data could be partitioned. More generally, it may be hard to find any reasonable way to partition the data before knowing how various researchers intend to use the data. And while the baseline model assumes an iid data generating process, we show in Section 3.1 that the dissemination noise in our model can continue to screen out hackers even when the raw dataset consists of non-i.i.d. observations.

2.1.4 Remarks about the Model

We make four remarks about the model.

First, this model features very powerful p-hackers. Some h fraction of researchers are totally ignorant about the true cause, but they are incentivized to game the system and try to fish for some covariate that plausibly explains the outcome variable and passes the policymaker's test. This kind of p-hacking by multiple hypothesis testing is made easy by the fact that they have a continuum of covariates to search over, and incur no cost from data mining. This represents today's "wide" datasets and fast computers that enable ever easier p-hacking. Our analysis suggests that dissemination noise can improve social welfare even in settings where p-hacking is costless.

Second, we view the principal as an entity that wishes to maximize the positive social impact of the research done using its data, but has limited power in influencing the institutional conventions surrounding how research results are evaluated and implemented into policies. In the model, the principal cannot change the policymaker's test. Examples include private firms like 23andMe that possess a unique dataset but have little say in government policy-making, and agencies like the US Census Bureau that are charged with data collection

[7] For instance, the Bureau publishes the annual Statistics of U.S. Businesses that contains payroll and employee data on small American businesses. Statisticians at the Bureau point out that separately adding noise to each business establishment's survey response provides "an alternative to cell suppression that would allow us to publish more data and to fulfill more requests for special tabulations" (Evans, Zayatz, and Slanta, 1998). The dataset has been released with this form of dissemination noise since 2007 (US Census Bureau, 2021b). More recently, the Bureau has finalized the parameters of the noise infusion system for the 2020 Census redistricting data in June 2021 (US Census Bureau, 2021a). The noise will be added through the new differentially private TopDown Algorithm that replaces the previous methods of data suppression and data swapping (Hawes and Rodriguez, 2021).

and data stewardship but do not directly evaluate research conclusions. Such organizations already introduce intentional noise in the data they release for the purpose of protecting individual privacy, so they may be similarly willing to use the same tool to improve the quality of policy interventions that come from studies done on their data.

In line with this interpretation of the principal, she cannot influence the research process or the policymaker's decision, except through changing the quality of the disseminated data. In particular, the principal cannot impose a cost on the agent to submit a proposal to the policymaker, write a contract to punish an agent in the future if it becomes known that his proposal led to a misguided policy, or change the legislating norms that govern how proposals get tested and turned into policies.

Third, the dataset in our model contains just one outcome variable, but in reality a typical dataset (e.g., the US Census data) contains many outcome variables and can be used to address many different questions. We can extend our model to allow for a countably infinite number of outcome variables $Y^1, Y^2, \ldots$, with each outcome associated with an independently drawn true cause and red herring. After the principal releases a noisy version of the data, one random outcome becomes research relevant and the agent proposes a model for this specific outcome. Our analysis, including the characterization of the optimal level of noise, remains unchanged in this world where the research question is not known at the time when the principal publishes the data. The crucial aspect of the dataset is that it is wide, which is captured by having a countable number of outcomes but an uncountable number of covariates a. The more realistic setting where data is released before a relevant question emerges provides a foundation for the principal not being able to screen the agent types by eliciting their private information about the true cause without giving them any data. Who is a maven depends on the research question and the outcome variable being studied, and it is infeasible to test a researcher's domain expertise with respect to every conceivable future research question.

Fourth, the policymaker's exogenous test only evaluates how well the the agent's proposal explains the raw dataset, and does not provide the agent any other way to communicate his domain expertise. Such a convention may arise if domain expertise is complex and difficult to convey credibly: for instance, perhaps an uninformed hacker who has found a strong association in the data can always invent a plausible-sounding story to justify why a certain covariate causes the outcome. We also assume that the policymaker's test is mechanically set, and does not adjust to the presence of p-hackers. This represents a short-run stasis in the science advocacy process or publication norms — for instance, while we know how to deal with multiple hypotheses testing, a vast majority of academic journals today still treat $p < 0.05$ as a canonical cutoff for statistical significance. Our analysis suggests that dissemination noise can help screen out misguided policies in the short run, when the principal must take as given a policymaking environment that has not adapted to the possibility of p-hacking.

Finally, the model presumes that there exists a correct explanation for the variable of interest in the dataset. In Section 3.4 we relax this assumption.

2.2 Optimal Level of Noise

To find the optimal level of noise, we first derive the behavior of the hacker and the maven from their payoffs given a noise level q.

Lemma 1. *For any $q \in [0, 1/2]$, it is optimal for the hacker to propose any $a \in A$ that satisfies $\hat{X}_n^a = Y_n$ for every $1 \leq n \leq N$, and it is optimal for the maven to propose $a \in \{a^*, a^r\}$ that maximizes the number of observations n for which $\hat{X}_n^a = Y_n$ (and randomize uniformly between the two covariates if there is a tie). Under any optimal behavior of the agents, the hacker's proposal is equal to the true cause with probability 0, while the maven's proposal is equal to the true cause if and only if it passes the policymaker's test.*

If the principal releases data without noise, then a maven will be able to discover the true cause, but a hacker will also find an almost surely misguided policy based on a covariate that is perfectly correlated with Y in the raw data. The payoff to the principal from releasing the data without noise is therefore $1 - 2h$. More generally, when the agents follow the optimal behavior described in Lemma 1, the principal's expected utility from choosing noise level q is

$$-hV_{\text{hacker}}(q) + (1-h)V_{\text{maven}}(q),$$

where $V_i(q)$ is the probability that agent type i's proposal passes the policymaker's test in the raw data, when the data was released with noise level q.

Our next observation represents the key idea in the paper: A small amount of noise does not harm the maven's chances to find a passing policy, but it creates baits for the hacker that hinders their ability to find a passing policy.

Lemma 2. *If $V_i(q)$ is the probability that agent type i's proposal passes the policy maker's test in the raw data, then $V'_{maven}(q) = -\binom{2N-1}{N}Nq^{N-1}(1-q)^{N-1}$ and $V'_{hacker}(q) = -N(1-q)^{N-1}$. In particular, $V'_{maven}(0) = 0$ while $V'_{hacker}(0) = -N$.*

The intuition is that a small amount of noise does not prevent the agent from finding a passing policy if he has a small set of candidate covariates in mind before seeing the data. But if the agent has a very large set of candidate covariates, then there is a good chance that the noise turns several covariates out of this large set into baits. For example if $N = 100$ and $q = 0.01$, the probability that a covariate that perfectly correlates with Y in the noisy dataset is a bait is 63.4%. But the probability that one of the maven's two covariates (a^* or a^r) is a bait, given it is equal to Y in every observation in the noisy dataset, is close to 0%. That is, hackers fall for baits at a higher rate than mavens precisely because they engage in p-hacking and try out multiple hypotheses. Yet p-hacking is the hackers' best response, even though they know the dataset contains baits.

We can show the principal's overall objective $-hV_{\text{hacker}}(q) + (1-h)V_{\text{maven}}(q)$ is strictly concave and therefore the first-order condition characterizes the optimal q, provided the solution is interior:

Proposition 1. *If $\frac{h}{1-h} \leq \binom{2N-1}{N}(1/2)^{N-1}$ then the optimal noise level is*

$$q^* = \left(\frac{h}{1-h}\frac{1}{\binom{2N-1}{N}}\right)^{1/(N-1)}.$$

More noise is optimal when there are more hackers and less is optimal when there are more observations. If $\frac{h}{1-h} \geq \binom{2N-1}{N}(1/2)^{N-1}$ *then the optimal noise level is* $q^* = 1/2$.

Proposition 1 gives the optimal dissemination noise in closed form. With more hackers, screening out their misguided policies becomes more important, so the optimal noise level increases. With more observations, the same level of noise can create more baits, so the principal can dial back the noise to provide more accurate data to help the mavens.

The principal cannot hope for an expected payoff higher than $1 - h$. This first-best benchmark corresponds to the policymaker always implementing the correct policy when the agent is a maven, and not implementing any policy when the agent is a hacker (recall that hackers have zero probability of proposing the true cause). As the number of observations N grows large, the principal's expected payoff under the optimal noise approaches this first-best benchmark.

Proposition 2. *For any* $0 < h < 1$, *the principal's expected payoff under the optimal noise level approaches* $1 - h$ *as* $N \to \infty$.

That is to say, injecting the optimal level of noise is asymptotically optimal among all mechanisms for screening the two agent types, including mechanisms that involve a hold-out dataset, or take on more complex forms that we have not considered in our analysis.

2.3 Dissemination Noise and p-Value Thresholds

So far, we have taken the policymaker's test as exogenously given, so that the agent's proposal a passes only if $X_n^a = Y_n$ for every observation n. Now suppose the principal can choose both the level of noise $q \in [0, 1/2]$ and a passing threshold $\underline{N} \in \{1, \ldots, N\}$ for the test, so that a proposal passes whenever $X_n^a = Y_n$ for at least $\underline{N}$ out of the N observations.

Proposition 3. *When the principal can optimize over both the passing threshold and the noise level, the optimal threshold is* $\underline{N} = N$, *and the optimal noise level is the same as in Proposition 1.*

The intuition for Proposition 3 is that the passing threshold does not influence either the hacker or the maven's behavior. In particular, the hacker's payoff-maximizing strategy always involves proposing a policy a so that $X_n^a = Y_n$ in every observation n, as this maximizes the probability of passing a test with any threshold. Lowering the passing threshold hurts the principal when she faces a hacker, since it means the policymaker implements misguided policies more often.

We can interpret this result to say that stringent p-value thresholds and dissemination noise are *complementary tools* for screening out p-hackers and misguided policies. Think of different passing thresholds as different p-value thresholds, with the threshold $N = \underline{N}$ as the most stringent p-value criterion that one could impose in this environment. Benjamin et al. (2018)'s article about lowering the "statistical significance" p-value threshold for new findings includes the following discussion:

> "The proposal does not address multiple-hypothesis testing, P-hacking, [...] Reducing the P value threshold complements — but does not substitute for — solutions to these other problems."

Our result formalizes the sense in which reducing p-value threshold complements dissemination noise in improving social welfare from research.

3 Extensions of the Baseline Model

The model we have laid out is clearly stylized to focus on the central trade-offs in adding dissemination noise. We explore several relaxations of our simplifying assumptions. First, we consider non-i.i.d. observations, such as those in time-series data, or data from social networks. Second, we look at a model in which the maven can face a red herring that is harder to disprove with data than we have assumed so far. Third, we relax the assumption of a continuum of potential covariates. Finally, we relax the assumption that there exists a correct explanation for the outcome variable in the data.

3.1 Non-i.i.d. Observations

In the baseline model, for each $a \in A$ the raw data contains N i.i.d. observations of the a covariate X^a. This gives a vector $X^a \in \{0,1\}^N$ with independent and identically-distributed components X_n^a. The i.i.d. assumption rules out certain applications where there is natural dependence between different observations of the same covariate, such as data from social networks, panel data, or time-series data. We now relax this assumption. For each policy a there is associated a covariate $X^a \in \{0,1\}^N$, but the ex-ante distribution of X^a is given by an arbitrary, full-support $\mu \in \Delta(\{0,1\}^N)$. (Full support means that $\mu(x) > 0$ for every $x \in \{0,1\}^N$.)

The model is otherwise the same as the baseline model of Section 2. In particular, the true cause and the red herring covariates still exhibit perfect correlation and perfect negative correlation with the outcome variable, viewed as random vectors in $\{0,1\}^N$. To be concrete, Nature first generates a^* and a^r independently and uniformly from A. Then, Nature generates the outcome variable as a vector, $Y \sim \mu$. Then Nature sets $X^{a^*} = Y$ and $X^{a^r} = 1 - Y$. Finally, for each $a \in A \backslash \{a^*, a^r\}$, Nature draws the vector $X^a \sim \mu$ independently (and independently of Y).

When the principal prepares the noisy dataset $\mathcal{D}(q)$, noise is still added to each observation of each covariate independently with probability q. We first show that the hacker's payoff-maximizing strategy is still to propose a covariate a with $Y_n = \hat{X}_n^a$ for every observation n in the noisy data. That is, regardless of how the N observations are correlated, there is nothing more "clever" that a hacker could do to increase the probability of passing the test than to "maximally p-hack" and propose a covariate that appears perfectly correlated with the outcome variable in the noisy dataset.

Lemma 3. *For any $y \in \{0,1\}^N$, $\mathbb{P}[X^a = y \mid \hat{X}^a = x]$ is maximized across all $x \in \{0,1\}^N$ at $x = y$, for any $0 \leq q \leq 1/2$ and full-support μ.*

Using the hacker's optimal behavior in Lemma 3, we can show that a small amount of dissemination noise will differentially impact the two types' chances of passing the test, thus it improves the principal's expected payoff as in the case when the observations are i.i.d.

Proposition 4. *For any full support $\mu \in \{0,1\}^N$, $V'_{maven}(0) = 0$ while $V'_{hacker}(0) < 0$. In particular, there exists $\bar{q} > 0$ so that any noise level $0 < q \leq \bar{q}$ is strictly better than $q = 0$.*

When the raw dataset consists of correlated observations — for example, data on N individuals who influence each other in a social network or N periods of time series data for a very large number of economic indicators — it may be unreasonable for the principal to only release some of the observations (e.g., only the time series data for even-number years) and keep the rest of the raw dataset as a secret holdout set to test the agent's proposal and identify the p-hackers. Our procedure of releasing all of the observations infused with i.i.d. dissemination noise (which resembles the current implementation of privacy-preserving noise) may be more reasonable in such contexts. Proposition 4 shows our main insight continues to be valid. Even when the observations have arbitrary correlation, which the hackers may take advantage of in their data mining, a small amount of dissemination noise still strictly improves the principal's expected payoff.

3.2 More Misleading Red Herrings

In the baseline model of Section 2, we assumed that the "red herring" covariate is perfectly negatively correlated with Y. This corresponds to an extreme kind of complementarity between theory and data in learning the true cause, as even a small amount of data can disprove the theoretically plausible alternative and identify the truth.

We now consider a situation where the red herring is more misleading, and not always easily ruled out by the data. We allow the red herring to be just like any other covariate in A, so that it is simply uncorrelated with the outcome instead of perfectly negatively correlated with it. So, the only modification relative to the baseline model is that X^{a^r}, like X^a for $a \notin \{a^*, a^r\}$, is also independent of Y.

It is easy to see that the change in how we model the red herring covariate does not affect the optimal behavior of either the hacker or the maven. A hacker proposes some covariate a that perfectly correlates with Y in the noisy dataset. A maven chooses between a^* and a^r according to how they correlate with Y in the noisy data, randomizing if there is a tie. When the red herring covariate is independent of the outcome in the raw dataset, the maven falls for the red herring with a higher probability for every level of noise. Also, unlike in the baseline model where the maven gets rejected by the policymaker if he happens to propose the red herring, here the maven may propose a misguided policy that passes the test if all N realizations of X^{a^r} perfectly match that of the outcome Y in the raw dataset.

Our next result implies that a strictly positive amount of dissemination noise still improves the principal's expected payoffs given "reasonable" parameter values.

Proposition 5. *The derivative of the principal's expected payoff, as a function of the noise level q, is $hN - (1-h)N(N+1)2^{-(N+1)}$ when evaluated at $q = 0$. This derivative is strictly positive when $h > \frac{N+1}{2^{N+1}+N+1}$. In particular, when this condition on h is satisfied, there exists $\bar{q} > 0$ so that any noise level $0 < q \leq \bar{q}$ is strictly better than $q = 0$.*

When the red herring covariate is perfectly negatively correlated with the outcome variable, we found that the optimal level of noise is always strictly positive. Proposition 5 says this result remains true even when the red herring can be more misleading, provided there are enough hackers relative to the number of observations in the data. The lowest amount of hackers required for dissemination noise to be useful converges to 0 at an exponential rate as N grows. For example, even when there are only $N = 10$ observations, the result holds whenever more than 0.53% of all researchers are p-hackers.

3.3 Finite Number of Covariates

In the baseline model, we imagine there is a continuum of covariates $a \in A = [0, 1]$. This represents an environment with a very "wide" dataset, where there are many more candidate explanatory variables and econometric specifications than observations. But the main idea behind our result remains true if there is a finite but large number of covariates.

Suppose $A = \{1, 2, ..., K\}$, so there are $2 \leq K < \infty$ covariates. As in the baseline model, a true cause and a red herring are drawn from the set of all covariates, with all pairs $(a^*, a^r) \in A^2$, $a^* \neq a^r$ equally likely. The outcome Y is perfectly positively correlated with the true cause, so $Y = X^{a^*}$. The other covariates (including the red herring) are independent of the outcome, as in the extension in Section 3.2. Once (a^*, a^r) are drawn, we have fixed a joint distribution among the $K + 1$ random variables $(Y, X^1, ..., X^K)$. The raw dataset consists of N independent observations drawn from this joint distribution. The principal releases a noisy version of the dataset with noise level $q \in [0, 1/2]$ as before.

As the number of covariates K grows, there is more scope for p-hacking to generate misguided policies. This happens for two reasons. First, holding fixed the policymaker's test and the number of observations, it is easier for the p-hacker to find a covariate that passes the test when there are more covariates to data mine. Second, the probability that the p-hacker proposes an incorrect covariate also increases with K. When the number of covariates is finite, a p-hacker has a positive probability of stumbling upon the true cause by chance, but this probability converges to 0 as K goes to infinity. As the statistical environment becomes more complex and the number of potential models explodes ($K \to \infty$), not only is the p-hacker more likely to pass the test, but his proposal also leads to a misguided policy with a higher probability conditional on passing.

In fact, the social harm of a p-hacker converges to that of the baseline model with a continuum of covariates as $K \to \infty$. As a result, we can show that a small amount of dissemination noise improves the principal's payoffs relative to no noise when K is finite but large, provided the fraction of hackers is not too close to 0.

Proposition 6. *Let the number of observations N and the fraction of hackers $0 < h < 1$ be fixed, and suppose $h > \frac{N+1}{2^{N+1}+N+1}$. There exists a noise level $q' > 0$ and an integer $\underline{K}$ so that when there are K covariates with $K \geq \underline{K}$, the principal does strictly better with noise level q' than noise level 0.*

The lower bound on the fraction of hackers in this result is mild and matches the condition from Proposition 5. If there are 10 observations and more than 0.53% of researchers are uninformed, then the principal can improve her expected payoff with a non-zero amount of noise whenever the (finite) dataset contains enough covariates.

3.4 No True Cause

We turn to a version of our environment where all models can be wrong. Suppose that, with some probability, none of the covariates in the dataset is the causal mechanism behind the outcome. As in the baseline model, Nature draws a^* and a^r uniformly at random from $[0,1]$. With probability $0 < \beta \leq 1$, the covariate a^* is the true cause and X^{a^*} is perfectly correlated with Y. But with the complementary probability, a^* is another red herring and X^{a^*} is perfectly negatively correlated with Y (just as X^{a^r} is). The maven observes a^r and a^* — the maven does not know which is which, and does not know whether a^* is the true cause or another red herring.

The agent can either propose a covariate $a \in A$, or report $\varnothing$ indicating that none of the covariates is the true cause. If the agent proposes a covariate, the policymaker implements it if and only if it passes the policymakers' test (that is, if it is perfectly correlated with Y in the original dataset). The principal gets 1 if the true cause is implemented, 0 if the proposal is rejected, and -1 if any other covariate is implemented: so when the data does not contain the true cause, the principal gets -1 no matter which policy gets implemented. If the agent reports $\varnothing$, then no policy is implemented and the principal gets 0.

The agent gets $0 < w < 1$ from being right (either proposing the true cause when there is one, or reporting $\varnothing$ when no true cause exists in the dataset), and gets $1 - w$ when the reported covariate is implemented. (Note that agents would never abstain from proposing a covariate even if they had this option in the baseline model or in the previous extensions, since they always get zero utility from abstaining but expect to get strictly positive utility from proposing a random covariate.)

Proposition 7. *Suppose $w > 3/4$ and $\beta > w$. Then there exists some $\bar{q} > 0$ so that the principal strictly prefers any q level of noise with $0 < q \leq \bar{q}$ to 0 noise.*

This result says that even when there is some probability that none of the covariates is the true cause, provided this probability is not too high and agents put enough weight on being right, a small enough amount of dissemination noise is still strictly better than no noise. A stronger assumption on w is needed for this result compared to the previous results. This ensures that when the maven is sufficiently confident that neither a^* nor a^r is the true cause, he would rather report $\varnothing$ (and get the utility for being right) than report some wrong covariate that passes the policymaker's test.

4 A Numerical Example of Dissemination Noise and p-Hacking in Linear Regressions

While our baseline model is highly stylized, we show through a numerical example that dissemination noise may play a similar role in screening the researcher's expertise in more realistic empirical settings that do not satisfy all of our model's simplifying assumptions.

We consider a linear regression setting with a continuous outcome variable and some continuous covariates, where the outcome is the sum of three covariates plus noise. The three causal covariates are randomly selected from a set of potential explanatory variables, and the principal would like to implement a policy that correctly targets the causal covariates.

An uninformed agent may p-hack by trying out all regressions involving different triplets of explanatory variables to game the policymaker's test, which simply evaluates the agent's econometric specification based on its explanatory power on the raw dataset.

To be concrete, there are 20 covariates $X^1, \ldots, X^{20}$, with each $X^i \sim \mathcal{N}(0,1)$, where $\mathcal{N}(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2. It is known that the outcome Y is generated from a linear model $Y = X^{i_1^*} + X^{i_2^*} + X^{i_3^*} + \epsilon$ where $1 \leq i_1^* < i_2^* < i_3^* \leq 20$ are three of the covariates, with all triplets equally likely, and $\epsilon \sim \mathcal{N}(0,4)$ is an error term. Without loss, we analyze the case when the causal covariates have the realization $(i_1^*, i_2^*, i_3^*) = (1,2,3)$. The principal's raw dataset consists of 20 independent observations of the outcome variable and the covariates from their joint distribution.

The principal disseminates a noisy version of the data to the agent by adding an independent noise term with the distribution $\mathcal{N}(0, \sigma^2_{noise})$ to every realization of each covariate in the dataset. The noise variance σ^2_{noise} controls how much dissemination noise the principal injects into the released data.

The agent analyzes the data and proposes a model $(\hat{\imath}_1, \hat{\imath}_2, \hat{\imath}_3)$ for Y. Then a policymaker tests the proposed model on the raw data and implements a policy targeting the covariates $(\hat{\imath}_1, \hat{\imath}_2, \hat{\imath}_3)$ if the model passes the test. Suppose $(\hat{\imath}_1, \hat{\imath}_2, \hat{\imath}_3)$ passes the test when the linear regression's R^2 on the raw data exceeds a critical value, otherwise the proposal is rejected. The critical value is the 95-percentile R^2 when a triplet of covariates is chosen uniformly at random from all possible ones. The principal gets utility 1 if the correct specification $(1,2,3)$ passes the test, utility -1 if any other specification passes the test, and utility 0 if the proposal is rejected.

With some probability, the agent is a hacker who is uninformed about (i_1^*, i_2^*, i_3^*) and runs all $\binom{20}{3} = 1140$ linear regressions of the form $Y = X^{i_1} + X^{i_2} + X^{i_3} + \epsilon$ for different choices of the three covariates i_1, i_2, i_3 in the noisy data. The agent then proposes the model with the highest R^2 value. With complementary probability, the agent is a maven whose expertise in the subject lets him narrow down the causal model of Y to either the true $Y = X^1 + X^2 + X^3 + \epsilon$, or the incorrect model $Y = X^4 + X^5 + X^6 + \epsilon$. The maven runs two regressions using the noisy data, and proposes either $(1,2,3)$ or $(4,5,6)$ to the policymaker, depending on which regression has a higher R^2. (Unlike in the model where we derive optimal behavior for the agents, for this example their behavior are exogenously given.)

We draw a few comparisons between the example and our baseline model. Like our model, the example captures a setting with a wide dataset, in the sense that there are many more potential specifications (more than 1000) than there are observations in the data (20). The true cause a^* corresponds to the triplet $(1,2,3)$, and the red herring a^r corresponds to the triplet $(4,5,6)$. Unlike in the baseline model, this example does not feature a continuum of potential specifications, independence between different specifications (since two triplets may share some explanatory variables), or perfect correlation between the outcome and the true cause (since there is a noise term). Nevertheless, we numerically show that injecting dissemination noise in the form of choosing a strictly positive σ^2_{noise} has some of the same properties and trade-offs for the principal as in our simple baseline model.

Figure 1 depicts the expected utility of the principal conditional on the agent's type, as a function of the amount of dissemination noise that the principal adds to the covariates before releasing the dataset. The expected social harm from a hacker agent is mitigated when there is more noise. The idea is that when a hacker analyzes a noisy dataset, the model (i_1, i_2, i_3)

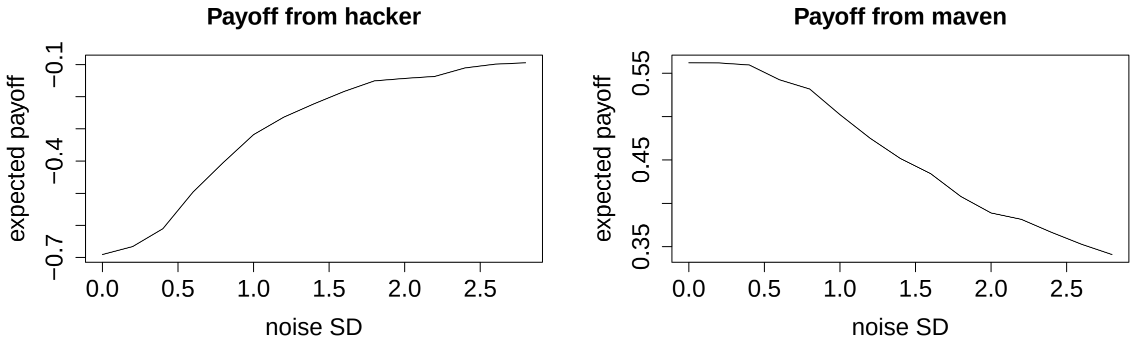

Figure 1: The principal's expected utility conditional on the agent being a hacker or a maven, as a function of the amount of dissemination noise.

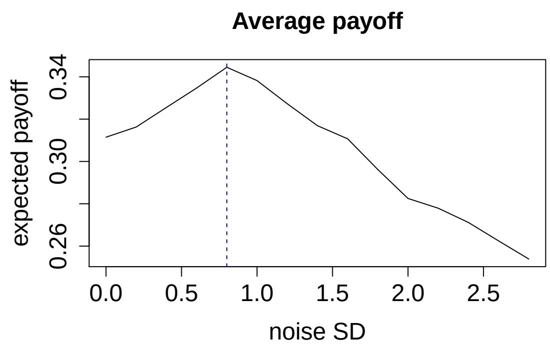

Figure 2: Expected utility of the principal as a function of the standard deviation of dissemination noise, when 20% of the agents are hackers and 80% are mavens.

with the highest regression R^2 in the noisy data is often a *bait* with poor R^2 performance in the true dataset. The covariates i_1, i_2, i_3 look correlated with the outcome Y only because they were hit with just the right noise realizations, but a hacker who falls for these baits and proposes the model (i_1, i_2, i_3) will get screened out by the policymaker's test, which is conducted in the raw data.

Of course, a maven is also hurt by the noise. The principal's expected payoff when facing a maven falls when more dissemination noise is added to data. The maven needs to use the data to compare the two candidates (1, 2, 3) and (4, 5, 6). Noisier data makes it harder to identify the true causal model.

Suppose 20% of the agents are hackers and 80% are mavens. Figure 2 shows the expected social welfare as a function of the amount of dissemination noise. The optimal dissemination noise trades off screening out hackers, using the baits created by noise, versus preserving data quality for mavens to identify the correct model.

The optimal amount of dissemination noise is strictly positive because a small amount of noise hurts hackers more than mavens. The intuition in this example, as in the model, is that it is likely that noise creates some baits in the disseminated dataset, but it is unlikely that the specification $(4, 5, 6)$ happens to contain one of the baits. The maven, who only

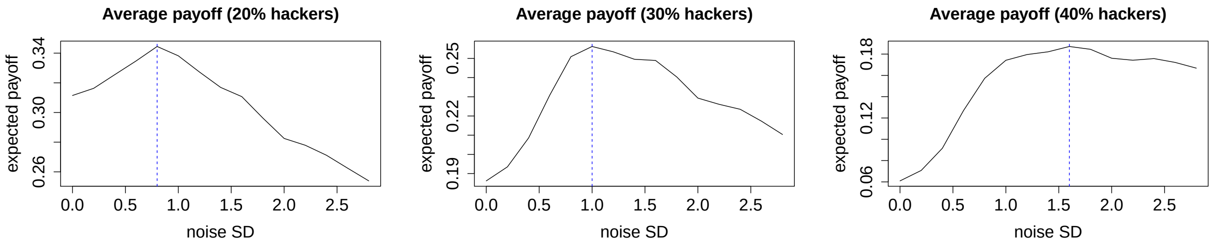

Figure 3: Comparative statics of the optimal amount of dissemination noise with respect to the fraction of hackers.

considers the two candidate specification $(1, 2, 3)$ and $(4, 5, 6)$, is much less likely to fall for a bait than the hacker, who exhaustively search through all possible specification.

Finally, Figure 3 illustrates the comparative statics of how the optimal level of dissemination noise varies with changes in the environment. When the fraction of hackers increases, more noise is optimal.

5 Dynamic Environment with Data Reuse

The baseline model presumes that a test performed on the raw dataset determines whether the agent's proposal is implemented as policy, but the raw data itself is never publicly revealed. In practice, if these tests determine high-stakes policy decisions, it may be impossible to credibly conduct them on a secret dataset. Transparency concerns may require that the data used to test proposals must be made public in a timely manner.[8]

We now turn to a model where the principal owns a dataset that is used by various researchers to study different questions over a long period of time. (For instance, the US Census is only conducted every ten years and the same dataset is used to answer a large number of research questions in the social sciences.) As researchers propose policy interventions to address different research questions, their proposals must all be tested in a transparent way. Suppose the principal is legally bound to periodic releases of noisy data – multiple noisy "waves" of the data are made public over time, which are then used to test the most recent policy proposals. A policy proposal made in February of 2022, for example, would be tested using the March 2022 release of noisy data. A March proposal would be tested against the April release, and so on. We assume that the principal cannot delay the release of the noisy data used to test past proposals, even though such public data releases become accessible to future p-hackers who will try to data mine the same dataset.

In our dynamic model, time works in the hackers' favor, as p-hacking becomes easier when data is reused. Hackers can exploit all past releases of the noisy data to propose policies that are increasingly likely to pass the policymaker's test. As we shall see, in the end, the principal will rationally give up on adding noise to test data, and will release the

[8]Indeed secret access is the focus of the work of Dwork et al. (2015b), who propose methods for differentiably-private access to the raw data. Their work is motivated by the same concerns over re-use that we turn to in this section. Our results in Section 5 may be read as validating the use of the methods in Dwork et al. (2015b).

original raw dataset. At that point, the hacker can always find misguided policies that pass the test and get implemented by the policymaker. The promise of using noisy data to deal with p-hacking is real, but finitely lived.

Time is discrete and infinite: $t = 0, 1, 2, \ldots$ In period 0, the principal receives a raw dataset as before, but with the following changes compared to the baseline static model:

- The dataset contains a continuum of covariates, $(X^a)_{a \in A}$. But, there is a countably infinite number of outcome variables, $(Y^t)_{t=0,1,2,\ldots}$. A true cause $a_t^* \in A$ is drawn uniformly at random from A for each outcome Y^t. The principal does not receive more data in later periods: no additional outcomes, covariates, or observations will arrive. The "dynamic" aspect of the model concerns how a fixed dataset must be reused over time.

- Suppose for simplicity the maven knows the true cause of every outcome, so red herrings are not generated.

- For simplicity, suppose there is only a single observation $N = 1$ of the outcomes and the covariates. This is for tractability so that the state space of the resulting model becomes one-dimensional. It is, of course, an extreme version of the assumption of a wide dataset.

- Suppose the unconditional distribution of each outcome variable and each covariate is Bernoulli(κ) for some $0 < \kappa < 1$. The baseline model looked at the case where $\kappa = 0.5$.

These simplifying changes allow us to focus on the intertemporal trade-offs facing the principal. In each period, she generates a noisy version of the raw dataset to evaluate the agent's proposal. This testing dataset must be publicly released before another agent uses the same dataset to propose the causal covariate behind another outcome variable. The principal will have a short-term incentive to decrease noise and thus improve the quality of tests for current proposals, but a long-term incentive to increase noise so as to plant baits for future hackers. The intertemporal trade-off will be affected by a "stock of randomness" that is decreased as time passes.

In each period, the principal releases a possibly noisy version of the raw data: in period t, she releases a dataset $\mathcal{D}(q_t)$ after adding a level q_t of noise to the raw dataset. The parameter q_t is, as before, the probability that each X^a is flipped. (As in the baseline model, the principal only perturbs covariates, not outcome variables.) Each release is a *testing dataset*. Note that the principal always adds noise to the raw dataset, not to the previous iteration of the noisy dataset.

In each period $t = 1, 2, \ldots$, society is interested in enacting a policy to target the true cause behind the outcome $Y^{m(t)}$, where $m(t)$ is the t-th outcome with a realization of 1 in the principal's dataset. So, in the dynamic model we interpret an outcome realization of 0 as benign, and an outcome realization of 1 as problematic and requiring intervention. A short-lived agent arrives in each period t; the agent is a hacker with probability h and a maven with complementary probability. If the agent is a maven, recall that we are assuming the agent always knows and proposes the true cause of $Y^{m(t)}$. If the agent is a hacker, he uses all of the testing datasets released by the principal up to time $t - 1$ to make a proposal that maximizes the probability of being implemented. After receiving the agent's proposal a, the principal generates and publishes period t's testing dataset $\mathcal{D}(q_t)$. The policymaker implements policy a if $Y^{m(t)} = \hat{X}^a$ in this period's (possibly noisy) testing dataset. In period t, the principal gets a payoff of 1 if the true cause for $Y^{m(t)}$ passes the test, -1 if any other covariate passes the test, and 0 if the proposal is rejected. The principal maximizes expected discounted utility with discount factor $\delta \in (0,1)$

In each period $t \geq 2$, a hacker proposes a policy a with $\hat{X}^a = 1$ in all of the past testing datasets. Such a exists because there are infinitely many policies. (In the first period, the hacker has no information and proposes a policy uniformly at random.) Suppose a covariate a that shows as "1" in all the noisy testing datasets up to period $t-1$ has some b_t chance of being a *bait*, that is $X^a \neq 1$ in the raw data. Then the principal's expected payoff today from releasing a testing dataset with noise level q_t is

$$u(q_t; b_t) := (1-h)(1-q_t) + h(-(1-b_t)(1-q_t) - b_t q_t).$$

In the expression for u, $(1-h)(1-q_t)$ is the probability that the agent is a maven and the value of the true cause for Y^t in the period t testing dataset, $\hat{X}^{a_t^*}$, has not been flipped. The term $(1-b_t)(1-q_t)$ represents the probability that the hacker's policy is not a bait and its covariate value has not been flipped in the testing dataset. Finally, $b_t q_t$ is the probability that the hacker's policy is a bait, but its covariate value has been flipped in the testing dataset.

The principal's problem is similar to an intertemporal consumption problem. We can think of b_t as a stock variable that gets consumed over time. But rather than a stock of some physical capital, it measures the *stock of randomness* in the principal's raw dataset. This stock depletes as more and more noisy versions of the data are made public. We view $u(q; b)$ as the principal's flow utility from "consuming" $\frac{1}{2} - q$, where the stock of randomness left is b, and the stock evolves according to $b_{t+1} = \frac{b_t q_t}{(1-b_t)(1-q_t)+b_t q_t}$.

The intertemporal trade-offs faced by the principal are captured by $\frac{\partial u}{\partial q} < 0$, $\frac{\partial u}{\partial b} > 0$, and $\frac{\partial b_{t+1}}{\partial q_t} > 0$. In words, adding less noise to the testing dataset today gives higher utility today, since the maven's correct policy is more likely to pass the test, and the hacker's misguided policy is more likely to get screened out. But this depletes the stock of randomness faster, and makes it harder to defend against future hackers.

Our next result shows that, in every optimal solution to the principal's problem, the stock of randomness is always depleted in finite time. The basic idea is that noise has decreasing returns: the marginal effect of noise on slowing the decline of b_{t+1} is reduced as b_t decreases. There is a time t^* at which the principal abandons the use of noise.

Proposition 8. *Suppose that $h < 1/2$ and $\kappa \in (0,1)$. Let $\{(b_t, q_t)\}$ be a solution to the principal's problem. Then, for all t, $q_t < 1/2$ and b_t is (weakly) monotonically decreasing. There t^* such that*

- *If $t < t^*$ then $b_{t+1} < b_t$;*
- *If $t \geq t^*$ then $q_t = 0$ and $b_{t+1} = 0$.*

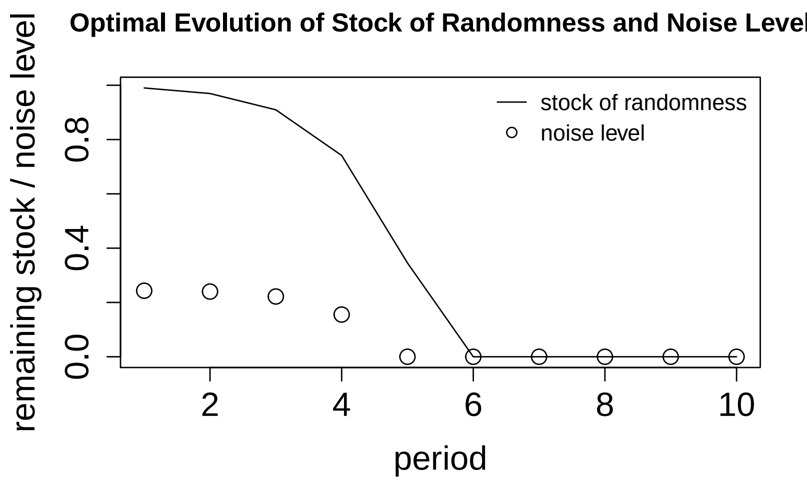

Figure 4: The evolution of the stock of randomness (i.e., the probability b_t that the hacker's best guess a is a bait with $X^a \neq 1$ in the raw dataset) and the noise level in an environment with 45% hackers, discount factor $\delta = 0.99$, and an unconditional probability $\kappa = 0.01$ that each covariate is equal to 1.

Figure 4 shows an example with $\kappa = 0.01$, $\delta = 0.99$, and $h = 0.45$. In period 1, a hacker has a 1% chance of guessing a covariate that would validate in the raw dataset. The principal releases noisy testing datasets at the end of periods 1, 2, 3, and 4. In period 5, a hacker can look for a covariate that has a value of "1" in each of the four testing datasets from the previous periods, and propose it as the model for today's outcome variable $Y^{m(5)}$. This proposal will validate in the raw dataset with more than 65% probability, reflecting a weakening defense against p-hackers as data is reused and the stock of randomness depletes. At this point, the principal finds it optimal to give up on dissemination noise and releases the raw dataset as the testing dataset at the end of period 5. In every subsequent period, both agent types will propose passing policies, so the policymaker implements correct policies 55% of the time and misguided policies 45% of the time.

6 Conclusion

We argue that infusing data with noise before making data public has benefits beyond the privacy protection guarantees for which the practice is currently being used. When noise is added to a dataset, it serves to bait uninformed p-hackers into finding correlations that can be shown to be spurious. The paper investigates these ideas in a simple model that captures the trade-off between preventing hackers from passing off false findings as true, and harming legitimate research that seeks to test an ex-ante hypothesis.

The paper is focused on the basic trade-offs involved in whether noise should be added at all, and does not address some of the more practical issues in implementing our proposal. One practical issue is that noisy data leads to biased statistical estimates, but for many common statistical procedures there is a simple fix (at least for large datasets). We imagine that the statistical agency would release information on the probability distribution of the noise it used. It is then possible to compute, and thus at least asymptotically correct for, the bias induced by noise. Another issue is related to the periodic releases of noisy data we have discussed in Section 5. How frequent should they be? There would likely be a conflict of interest between researchers who want frequent releases, and policy makers (or journal editors) who wish to preserve the stock of randomness. One solution might involve combining our proposal with the differentially private access to the hold-out data advocated by Dwork et al. (2015a). Finally, our model is stylized and not suited to a precise quantitative recommendation of how much noise should be added. There is clearly scope for further research in refining these questions.

References

Abrams, E., J. Libgober, and J. A. List (2021): "Research Registries: Taking Stock and Looking Forward," *Working Paper*.

Altmejd, A., A. Dreber, E. Forsell, J. Huber, T. Imai, M. Johannesson, M. Kirchler, G. Nave, and C. Camerer (2019): "Predicting the replicability of social science lab experiments," *PloS one*, 14, e0225826.

Benjamin, D. J., J. O. Berger, M. Johannesson, B. A. Nosek, E.-J. Wagenmakers, R. Berk, K. A. Bollen, B. Brembs, L. Brown, C. Camerer, et al. (2018): "Redefine statistical significance," *Nature Human Behaviour*, 2, 6–10.

Camerer, C. F., A. Dreber, E. Forsell, T.-H. Ho, J. Huber, M. Johannesson, M. Kirchler, J. Almenberg, A. Altmejd, T. Chan, E. Heikensten, F. Holzmeister, T. Imai, S. Isaksson, G. Nave, T. Pfeiffer, M. Razen, and H. Wu (2016): "Evaluating replicability of laboratory experiments in economics," *Science*, 351, 1433–1436.

Camerer, C. F., A. Dreber, F. Holzmeister, T.-H. Ho, J. Huber, M. Johannesson, M. Kirchler, G. Nave, B. A. Nosek, T. Pfeiffer, et al. (2018): "Evaluating the replicability of social science experiments in Nature and Science between 2010 and 2015," *Nature Human Behaviour*, 2, 637–644.

Chao, M.-T. and W. Strawderman (1972): "Negative moments of positive random variables," *Journal of the American Statistical Association*, 67, 429–431.

Di Tillio, A., M. Ottaviani, and P. N. Sørensen (2017): "Persuasion bias in science: can economics help?" *The Economic Journal*.

Di Tillio, A., M. Ottaviani, and P. N. Sørensen (2021): "Strategic sample selection," *Econometrica*, 89, 911–953.

Dwork, C., V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and A. Roth (2015a): "The reusable holdout: Preserving validity in adaptive data analysis," *Science*, 349, 636–638.

——— (2015b): “The reusable holdout: Preserving validity in adaptive data analysis,” *Science*, 349, 636–638.

Evans, T., L. Zayatz, and J. Slanta (1998): “Using Noise for Disclosure Limitation of Establishment Tabular Data,” *Journal of Official Statistics*, 14, 537–551.

Felgenhauer, M. and P. Loerke (2017): “Bayesian persuasion with private experimentation,” *International Economic Review*, 58, 829–856.

Felgenhauer, M. and E. Schulte (2014): “Strategic private experimentation,” *American Economic Journal: Microeconomics*, 6, 74–105.

Hawes, M. and R. Rodriguez (2021): “Determining the Privacy-loss Budget: Research into Alternatives to Differential Privacy,” https://www2.census.gov/about/partners/cac/sac/meetings/2021-05/presentation-research-on-alternatives-to-differential-privacy.pdf.

Henry, E. (2009): “Strategic disclosure of research results: The cost of proving your honesty,” *The Economic Journal*, 119, 1036–1064.

Henry, E. and M. Ottaviani (2019): “Research and the approval process: the organization of persuasion,” *American Economic Review*, 109, 911–55.

Imai, T., K. Zemlianova, N. Kotecha, and C. F. Camerer (2017): “How common are false positives in laboratory economics experiments? Evidence from the p-curve method,” *Working Paper*.

Ioannidis, J. P. (2005): “Why most published research findings are false,” *PLoS medicine*, 2, e124.

Open Science Collaboration (2015): “Estimating the reproducibility of psychological science,” *Science*, 349.

Simmons, J. P., L. D. Nelson, and U. Simonsohn (2011): “False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant,” *Psychological science*, 22, 1359–1366.

Simonsohn, U., L. D. Nelson, and J. P. Simmons (2014): “P-curve: a key to the file-drawer.” *Journal of experimental psychology: General*, 143, 534.

US Census Bureau (2021a): “Census Bureau Sets Key Parameters to Protect Privacy in 2020 Census Results,” https://www.census.gov/newsroom/press-releases/2021/2020-census-key-parameters.html.

——— (2021b): “Technical Documentation for Statistics of U.S. Businesses,” https://www.census.gov/programs-surveys/susb.html.

Woolston, C. (2015): “Psychology journal bans P values,” *Nature*, 519, 9–9.

7 Proofs

7.1 Proof of Lemma 1

Proof. Any strategy of the hacker leads to zero probability of proposing the true cause, so the hacker finds it optimal to just maximize the probability of the proposal passing the test. If the hacker proposes a covariate that matches Y in n_1 observations and mismatches in n_0 observations, then the distribution of the number of matches in the raw dataset is $\text{Binom}(n_0, q) + \text{Binom}(n_1, 1-q)$. A covariate that matches the outcome variable in every observation in noisy dataset will have a distribution of $\text{Binom}(n_0+n_1, 1-q) = \text{Binom}(n_0, 1-q) + \text{Binom}(n_1, 1-q)$ as its number of matches in the raw dataset, and $\text{Binom}(n_0, 1-q)$ strictly first-order stochastic dominates $\text{Binom}(n_0, q)$ if $n_0 \geq 1$ and $q < 1/2$. Therefore the hacker finds it optimal to propose any $a \in A$ that satisfies $\hat{X}_n^a = Y_n$ for every $1 \leq n \leq N$.

For the maven, since the weight on being correct is more than $1/2$, it is never optimal to propose covariates other than a^* or a^r since these have zero chance of being the true cause. Out of the two candidate covariates that the maven narrows down to, the one that matches Y in more observations in the noisy dataset has a higher posterior probability of being the true cause. Note that if the maven proposes a^r, the policymaker always rejects the proposal since $X^{a^r} = 1 - Y$ in the raw dataset. Also, if the maven proposes a^*, it always passes the test since $X^{a^*} = Y$ in the raw dataset. □

7.2 Proof of Lemma 2

Proof. Without loss, look at the case where $Y_n = 1$ in every observation n. The hacker picks a policy a where $X_n^a = 1$ in every observation in the noisy dataset, so

$$V_{\text{hacker}}(q) = (1-q)^N.$$

For the maven, there are $2N$ bits of observations on the variables X^{a^*} and X^{a^r}. If strictly fewer than N bits are flipped, then the maven recommends the correct policy. If exactly N bits are flipped, then the maven recommends the correct policy $1/2$ of the time. So,

$$V_{\text{maven}}(q) = (\mathbb{P}[\text{Binom}(2N, q) < N] + \frac{1}{2}\mathbb{P}[\text{Binom}(2N, q) = N])$$

We have

$$\begin{aligned}
V'_{\text{maven}}(q) =& \frac{d}{dq}(\mathbb{P}[\text{Binom}(2N, q) < N] + \frac{1}{2}\mathbb{P}[\text{Binom}(2N, q) = N]) \\
=& \frac{d}{dq}(\mathbb{P}[\text{Binom}(2N, q) \leq N] - \frac{1}{2}\mathbb{P}[\text{Binom}(2N, q) = N]) \\
=& -2N \cdot \mathbb{P}[\text{Binom}(2N-1, q) = N] - \frac{1}{2}\frac{d}{dq}(q^N(1-q)^N\binom{2N}{N}),
\end{aligned}$$

where the last step used the identity that $\frac{d}{dq}\mathbb{P}[\text{Binom}(M, q) \leq N] = -M \cdot \mathbb{P}[\text{Binom}(M-1, q) =$

$N]$. Continuing,

$$
\begin{aligned}
&-2N\cdot q^N(1-q)^{N-1}\binom{2N-1}{N}-\frac{1}{2}\binom{2N}{N}N(q^{N-1}(1-q)^N-q^N(1-q)^{N-1})\\
=&\binom{2N-1}{N}Nq^{N-1}(1-q)^{N-1}\left(-2q-\frac{1}{2}\cdot 2\cdot((1-q)-q)\right),
\end{aligned}
$$

using the identity $\binom{2N}{N}=2\cdot\binom{2N-1}{N}$. Rearranging shows the lemma. □

7.3 Proof of Proposition 1

Proof. Using the Lemma 2,

$$
\frac{d}{dq}[-hV_{\text{hacker}}(q)+(1-h)V_{\text{maven}}(q)]=hN(1-q)^{N-1}-(1-h)\binom{2N-1}{N}Nq^{N-1}(1-q)^{N-1}.
$$

The FOC sets this to 0, so $h-(1-h)\binom{2N-1}{N}q^{N-1}=0$. Rearranging gives $q^*=\left(\frac{h}{1-h}\frac{1}{\binom{2N-1}{N}}\right)^{1/(N-1)}$. We know $h\mapsto\frac{h}{1-h}$ is increasing, so $\frac{\partial q^*}{\partial h}>0$. We know $N\mapsto\binom{2N-1}{N}$ is increasing in N, therefore both the base and the exponent in q^* decrease in N, so $\frac{\partial q^*}{\partial N}<0$. □

7.4 Proof of Proposition 2

Proof. For any fixed noise level $0<q<0.5$, the principal's expected payoff with N observations is $-h\cdot(1-q)^N+(1-h)\cdot[\sum_{k=0}^{N-1}q^k(1-q)^{2N-k}\cdot\binom{2N}{k}+\frac{1}{2}\cdot q^N\cdot(1-q)^N\cdot\binom{2N}{N}]$. We have $\lim_{N\to\infty}(1-q)^N=0$, while $\sum_{k=0}^{N-1}q^k(1-q)^{2N-k}\cdot\binom{2N}{k}$ is the probability $\mathbb{P}[B(2N,q)\le N-1]$ with $B(2N,q)$ a binomial random variable with $2N$ trials and a success rate q strictly less than 0.5. We have $\lim_{N\to\infty}\mathbb{P}[B(2N,q)\le N-1]=1$. The limit of this payoff expression as $N\to\infty$ is $1-h$. The principal's expected payoff using the optimal level of noise for each observation size N must be even higher, so it must also converge to $1-h$. □

7.5 Proof of Proposition 3

Proof. The principal's expected utility conditional on the agent being a maven is the same for every $\underline{N}\in\{1,...,N\}$, since the maven always proposes either a^* or a^r depending on which covariate matches Y in more observations, and the proposal passes the $\underline{N}$ threshold if and only if it is a^*, since $X^{a^r}=1-Y$ does not match the outcome in any observation in the raw dataset.

As shown in the proof of Lemma 1, the distribution of the number of matches between X^a and Y in the raw dataset increases in the first-order stochastic sense with the number of matches between $\hat{X}^a$ and Y in the noisy dataset. So, for any test threshold $\underline{N}$, the hacker finds it optimal to propose a covariate a with $\hat{X}^a_n=Y_n$ for every n.

Therefore, the only effect of lowering $\underline{N}$ from N is to increase the probability of the hacker's misguided policies passing the test. □

7.6 Proof of Proposition 8

Proof. Define 1 minus the state, $f = 1 - b$. Define $u(q, f)$ as the principal's expected utility today from releasing testing set with noise level q when the hacker's best guess has $1 - f$ chance of being a bait in the raw dataset. We are studying the Bellman equation

$$v(f) = \max\{u(q, f) + \delta v\left(\frac{f(1-q)}{f(1-q) + (1-f)q}\right) : q \in [0, 1/2]\}$$

First we argue that $v : [0, 1] \to \mathbb{R}$ is monotone decreasing and convex. Let $C_B([0, 1])$ denote the set of continuous bounded functions on $[0, 1]$. Recall that v is the unique fixed point of the Bellman operator $T : C_B([0, 1]) \to C_B([0, 1])$, with

$$Tw(f) = \max\{u(q; b) + \delta w\left(\frac{bq}{(1-b)(1-q) + bq}\right) : q \in [0, 1/2]\}.$$

Observe that $b \mapsto \frac{bq}{(1-b)(1-q)+bq}$ is concave when $q \leq 1/2$ (its second derivative is $\frac{q(1-q)(2q-1)}{[(1-b)(1-q)+bq]^3}$). Then when w is convex and monotone decreasing, so is $b \mapsto w\left(\frac{bq}{(1-b)(1-q)+bq}\right)$, as the composition of a concave function and a monotone decreasing convex function is convex. Finally, Tw is convex because $f \mapsto u(q, f)$ is convex (linear), and Tw thus is the pointwise maximum of convex functions. So Tw is monotone decreasing. The fixed point v of T is the limit of $T^n w$, starting from any monotone decreasing and convex $w \in C_B([0, 1])$, so v is monotone decreasing and convex.

Observe that $f \leq \frac{f(1-q)}{f(1-q)+(1-f)q} = \theta(q, f)$, so along any path (q_t, f_t), f_t is monotone (weakly) increasing. In consequence, if f_t is large enough, $f_{t'}$ will be large enough for all $t' \geq t$.

Recall that

$$u(q, f) = (1 - h)[1 - q] - h[f(1 - q) + (1 - f)q],$$

so

$$\partial_q u(q, f) = -1 + 2hf < 0$$

as $h < 1/2$. Hence, we have that

$$u(0, f) = 1 - h - hf \geq u(q, f) \geq -0.5(1 - h) - h(1/2) = u(1/2, f).$$

Note that $\theta(1/2, f) = f$, so that

$$v(f) \geq \frac{1 - h - 1/2}{1 - \delta}.$$

We proceed to show that $q_t < 1/2$. Observe that if, for some f_t it is optimal to set $q_t = 1/2$ then $f_{t+1} = \theta(q_t, f_t) = f_t$, and it will remain optimal to set $q_{t+1} = 1/2$. This means that, if it is optimal to set $q = 1/2$ for f, then $v(f) = \frac{1-h-1/2}{1-\delta}$. Since $h < 1/2$, u is strictly decreasing in q. So there is a gain in decreasing q from $1/2$, which will result in transitioning to $f' = \theta(q, f) > f = \theta(1/2, f)$. But recall that $\frac{1-h-1/2}{1-\delta}$ is a lower bound on v. So $v(f') \geq v(f)$. Hence,

$$\begin{aligned} u(q, f) + \delta v(f') - [u(1/2, f) + \delta v(f)] &= (2hf - 1)(q' - (1/2)) + \delta(v(f') - v(f)) \\ &\geq (2hf - 1)(q' - (1/2)) > 0. \end{aligned}$$

Now we show that for f large enough, but bounded away from 1, it is optimal to set $q = 0$. Given that v is convex, it has a subdifferential: for any f there exists $\partial v(f) \in \mathbb{R}$ with the property that $v(f') \geq v(f) + \partial v(f)(f' - f)$ for all f'. Since v is monotone decreasing, $\partial v(f) \leq 0$. Moreover, we can choose a subdifferential for each f so that $f \mapsto \partial v(f)$ is monotone (weakly) increasing.

Let $q' < q$. Suppose that q results in $f' = \theta(q, f)$ and q' in $f'' = \theta(q', f)$. The function θ is twice differentiable, with derivatives

$$\partial_x \theta(x, f) = \frac{-f(1-f)}{[f(1-x) + x(1-f)]^2} \text{ and } \partial_x^2 \theta(x, f) = \frac{2f(1-f)(1-2f)}{[f(1-x) + x(1-f)]^2}.$$

Hence, $q \mapsto \theta(q, f)$ is concave when $f \geq 1/2$.

Now we have:

$$\begin{aligned} u(q', f) + \delta v(f'') - [u(q, f) + \delta v(f')] &= (2hf - 1)(q' - q) + \delta(v(f'') - v(f')) \\ &\geq (2hf - 1)(q' - q) + \delta \partial v(f')(f'' - f') \\ &\geq (2hf - 1)(q' - q) + \delta \partial v(f') \partial_q \theta(q, f)(q' - q) \\ &> \left[(1 - 2h) + \delta \underbrace{\partial v(f') \frac{f(1-f)}{[f(1-q) + (1-f)q]^2}}_{A} \right] (q - q'), \end{aligned}$$

where the first inequality uses the definition of subdifferential, and the second the concavity of θ, so that $f'' - f' \leq \partial \theta(q, f)(q' - q'')$, and the fact that $\partial v(f') \leq 0$. The last inequality uses that $f < 1$. Recall that $1 - 2h > 0$.

For f close enough to 1, and since $\partial v(f') \leq 0$ are monotone increasing and therefore bounded below, we can make A as close to zero as desired. Thus, for $f < 1$ close to 1, we have that $u(q', f) + \delta v(f'') - u(q, f) + \delta v(f') > 0$ when $q' < q$. Hence the solution will be to set $q = 0$.

To finish the proof we show that $f_t \uparrow 1$ and hence there is t^* at which f_t is large enough that it is optimal to set $q_t = 0$.

Suppose that $f_t \uparrow f^* < 1$. Note that if $f' = \theta(q, f)$ then $q = \frac{f(1-f')}{f(1-f') + (1-f)f'}$. Thus (using K for the terms that do not depend on q or f)

$$u(q_t, f_t) = K - hf_t - (1 - 2hf_t) \left[\frac{f_t(1 - f_{t+1})}{f_t(1 - f_{t+1}) + (1 - f_t)f_{t+1}}. \right] \to K - hf^* - (1 - 2hf^*)\frac{1}{2} = K - \frac{1}{2}.$$

Then for any ε there is t such that $v(f_t) = \sum_{t' \geq t} \delta^{t'-t} u(q_{t'}, f_{r'}) < \frac{K - \frac{1}{2}}{1 - \delta} + \varepsilon$.

On the other hand, if the principal sets $q_t = 0$ it gets $u(0, f_t) = K - hf_t$, and transitions to $1 = \theta(0, f_t)$. Hence the value of setting $q = 0$ at t is

$$u(0, f_t) + \delta \frac{u(0, 1)}{1 - \delta} = K - hf_t + \delta \frac{K - h}{1 - \delta} > \frac{K - h}{1 - \delta} > \frac{K - 1/2}{1 - \delta}.$$

as $f_t \leq f^* < 1$ and $h < 1/2$.

Now choose ε such that $\frac{K-\frac{1}{2}}{1-\delta}+\varepsilon<\frac{K-h}{1-\delta}$. Then for t large enough we have

$$v(f_t)<u(0,f_t)+\delta\frac{u(0,1)}{1-\delta},$$

a contradiction because setting $q_t=0$ gives the principal a higher payoff than in the optimal path. $\square$

7.7 Proof of Lemma 3

Proof. Let $y\in\{0,1\}^N$ and $q\in[0,1/2]$ be given. Let $\mu_q\in\Delta(\{0,1\}^N\}$ be the distribution of covariate realizations in the noisy dataset with q level of noise. We have $\mathbb{P}[X^a=y\mid\hat{X}^a=y]=\frac{(1-q)^N\cdot\mu(y)}{\mu_q(y)}$. Also, for any $x\in\{0,1\}^N$ so that y and x differ in k of the N coordinates, we have $\mathbb{P}[X^a=y\mid\hat{X}^a=x]=\frac{(1-q)^{N-k}q^k\cdot\mu(y)}{\mu_q(x)}$. Note that

$$\mu_q(x)=\sum_{z\in\{0,1\}^N}\mu(z)\cdot q^{D(z,x)}(1-q)^{N-D(z,x)}$$

where $D(z,x)$ is the number of coordinates where z differs from x. By the triangle inequality, $D(z,y)\le D(z,x)+D(x,y)=D(z,x)+k$. This shows for every $z\in\{0,1\}^N$,

$$q^{D(z,y)}(1-q)^{N-D(z,y)}\ge q^{D(z,x)}(1-q)^{N-D(z,x)}\cdot(\frac{q}{1-q})^k.$$

So,

$$\mu_q(x)\le(\frac{q}{1-q})^k\cdot\sum_{z\in\{0,1\}^N}\mu(z)\cdot q^{D(z,y)}(1-q)^{N-D(z,y)}=(\frac{q}{1-q})^k\mu_q(y).$$

This shows

$$\frac{(1-q)^N\cdot\mu(y)}{\mu_q(y)}\ge\frac{(1-q)^N\cdot\mu(y)}{\mu_q(x)\cdot(\frac{1-q}{q})^k}=\frac{(1-q)^{N-k}q^k\cdot\mu(y)}{\mu_q(x)}.$$

$\square$

7.8 Proof of Proposition 4

Proof. First, observe the maven will choose the covariate $a\in\{a^*,a^r\}$ whose noisy realization $\hat{X}^a$ matches the outcome Y in more observations, regardless of μ. This is because the maven learns two candidates $a_1,a_2\in A$ and knows either $(X^{a_1}=Y,X^{a_2}=1-Y)$ or $(X^{a_1}=1-Y,X^{a_2}=Y)$, equally likely. The likelihood of the former is $\frac{1}{2}\cdot q^{(N-m_1)+m_2}(1-q)^{m_1+(N-m_2)}$ and the likelihood of the latter is $\frac{1}{2}\cdot q^{m_1+(N-m_2)}(1-q)^{(N-m_1)+m_2}$, where m_1,m_2 count the numbers of observations n where $\hat{X}^{a_1}_n=Y_n$ an $\hat{X}^{a_2}_n=Y_n$, respectively. Since $q\in[0,1/2]$, the first likelihood is larger if $m_1>m_2$, and vice versa. Also, maven's proposal is a^* if and only if it passes the policymaker's test. Thus we see that for any μ, $V_{\text{maven}}(q)$ is the same as when the observations are i.i.d.

Given the hacker's behavior in Lemma 3, to prove $V'_{\text{hacker}}(0)<0$ it suffices to show that for every $y\in\{0,1\}^N$ and μ, we have $\frac{\partial}{\partial q}\left[\mathbb{P}[X^a=y\mid\hat{X}^a=y]\right]_{q=0}<0$. For $z,x\in\{0,1\}^N$, let

$D(z,x)$ count the number of coordinates where z differs from x. Let $\mu_q \in \Delta(\{0,1\}^N\}$ be the distribution of covariate realizations in the noisy dataset with q level of noise. We may write (using the fact $N \geq 2$) that $\mu_q(y) = \mu(y) \cdot (1-q)^N + \mu(z : D(z,y) = 1) \cdot (1-q)^{N-1}q + f(q^2)$ where $f(q^2)$ is a polynomial expression where every term contains at least the second power of q. Therefore, $\frac{\partial}{\partial q}\left[\frac{(1-q)^N \mu(y)}{\mu_q(y)}\right]_{q=0}$ is:

$$\mu(y) \cdot \left[\frac{-N(1-q)^{N-1}\mu_q(y) - (1-q)^N \cdot [-N\mu(y)(1-q)^{N-1} + \mu(z : D(z,y) = 1) \cdot ((1-q)^{N-1} + g(q))]}{(\mu_q(y))^2}\right]_{q=0}$$

where $f(0) = 0$. Evaluating, we get $\mu(y) \cdot \frac{-N\mu(y) - [-N\mu(y) + \mu(z:D(z,y)=1)]}{(\mu(y))^2} = -\frac{\mu(z:D(z,y)=1)}{(\mu(y))}$. Since μ has full support, both the numerator and the denominator are strictly positive, so $\frac{\partial}{\partial q}\left[\mathbb{P}[X^a = y \mid \hat{X}^a = y]\right]_{q=0} < 0$. □

7.9 Proof of Proposition 5

In this proof we adopt the following notation: we write d_Y for the realized vector Y_n, d^a for the realized vector X_n^a, for the ath covariate. In the noisy data, we use $\tilde{d}^a$ for the realization of the noisy version of the a covariate. As in other results, it is without loss to analyze the case where $d_Y = \mathbf{1}$, so the policy maker will only accept a proposal a if it satisfies that $d^a = \mathbf{1}$ in the raw data.

First, we derive the posterior probability of $d^a = \mathbf{1}$ given a realization of $\tilde{d}^a$ in the noisy dataset, and the resulting behavior of the hacker and the maven. The n component of $\tilde{d}^a$ is denoted $\tilde{d}^a_n$.

Lemma 4. *Suppose that $\tilde{d}^a$ satisfies $\sum_n \tilde{d}^a_n = k$. We have $\mathbb{P}[d^a = \mathbf{1} \mid \tilde{d}^a] = (1-q)^k(q)^{N-k}$. In particular, the hacker chooses some action a with $\tilde{d}^a = \mathbf{1}$, and the maven chooses the policy with the higher number of 1's among $\tilde{d}^{a^*}$ and $\tilde{d}^{a^r}$.*

Proof. Consider any $\tilde{d}^a$ with $\sum_n \tilde{d}^a_n = k$. In the noisy dataset, for any q, every vector in $\{0,1\}^N$ is equally likely. So the probability of the data for policy a having realization $\tilde{d}^a$ is 2^{-N}. The probability of this realization in the noisy data and the realization being $d^a = \mathbf{1}$ in $\mathcal{D}$ is $2^{-N} \cdot (1-q)^k(q)^{N-k}$. So the posterior probability is $(1-q)^k(q)^{N-k}$.

The hacker chooses an action a as to maximize $\mathbb{P}[d^a = \mathbf{1} \mid \tilde{d}^a]$. The term $(1-q)^k(q)^{N-k}$ is maximized when $k = N$, since $0 \leq q \leq 1/2$.

The maven sees vectors with k_1, k_2 numbers of 1's. The likelihood of the data given the first action is the correct one is $(1-q)^{k_1}(q)^{N-k_1} \cdot 2^{-N}$ (since all vectors are equally likely in the noisy dataset conditional on $Y = \mathbf{1}$, for $a \neq a^*$). This is larger than $(1-q)^{k_2}(q)^{N-k_2} \cdot 2^{-N}$ when $k_1 \geq k_2$. □

Here is the expression for the principal's expected payoff as a function of q.

Lemma 5. *Let $A, C \sim$ Binom$(1-q, N)$ and $B \sim$ Binom$(1/2, N)$, mutually independent. The principal's expected payoff after releasing a noisy dataset $\mathcal{D}(q)$ is*

$$-h(1-q)^N + (1-h) \cdot \left[\sum_{k=0}^{N} \mathbb{P}(A = k) \cdot \left(\mathbb{P}(B < k) + \frac{1}{2}\mathbb{P}(B = k) - 2^{-N}(\mathbb{P}(C > k) + \frac{1}{2}\mathbb{P}(C = k))\right)\right].$$

Proof. With probability h, the agent is a hacker. By Lemma 4, the hacker recommends a policy $\hat{a}$ with $\tilde{d}^{\hat{a}} = \mathbf{1}$, which has $(1-q)^N$ chance of being accepted by the principal due to $d^{\hat{a}} = \mathbf{1}$.

With probability $1-h$, the agent is a maven. For the maven, $\sum_n \tilde{d}_n^{a^*} \sim \text{Binom}(1-q, N)$ and $\sum_n \tilde{d}_n^{a^r} \sim \text{Binom}(1/2, N)$ are independent. Whenever $\sum_n \tilde{d}_n^{a^*} > \sum_n \tilde{d}_n^{a^r}$, and with 50% probability when $\sum_n \tilde{d}_n^{a^*} > \sum_n \tilde{d}_n^{a^r}$, the maven recommend a^* by Lemma 4, which will be implemented by the principal.

When maven recommends a^r, the principal only implements it (and gets utility -1) if $d^{a^r} = \mathbf{1}$. The probability of $d^{a^r} = \mathbf{1}$ is 2^{-N}, and the probability of a^r being recommended given $d^{a^r} = \mathbf{1}$ and $\sum_n \tilde{d}_n^{a^*} = k$ is $\mathbb{P}(C > k) + \frac{1}{2}\mathbb{P}(C = k)$, interpreting C as the number of coordinates that did not switch from d^{a^r} to $\tilde{d}^{a^r}$. □

Now, with the formula for the principal's expected payoff in place, we can evaluate the derivative at $q = 0$ to prove Proposition 5.

Proof. We apply the product rule. First consider $\frac{d}{dq}\mathbb{P}(A = k)|_{q=0}$.

We have $\mathbb{P}(A = k) = (1-q)^k q^{N-k}\binom{N}{k}$. If $k < N-1$, then every term contains at least q^2 and its derivative evaluated at 0 is 0. For $k = N$, we get $(1-q)^N$ whose derivative in q is $-N(1-q)^{N-1}$, which is $-N$ evaluated at 0. For $k = N-1$, we get $(1-q)^{N-1}q \cdot N$, whose derivative evaluated at 0 is N.

We now evaluate, for $k = N-1, N$:

$$\left(\mathbb{P}(B < k) + \frac{1}{2}\mathbb{P}(B = k) - 2^{-N}(\mathbb{P}(C > k) + \frac{1}{2}\mathbb{P}(C = k))\right)|_{q=0}$$

When $k = N$ and $q = 0$, $\mathbb{P}(B < N) = 1 - 2^{-N}$, $\mathbb{P}(B = N) = 2^{-N}$, $\mathbb{P}(C > N) = 0$, $\mathbb{P}(C = N) = 1$. So we collect the term $-N((1-2^{-N}) + \frac{1}{2}\cdot 2^{-N} - 2^{-N}\cdot\frac{1}{2}) = -N(1-2^{-N})$.

When $k = N-1$ and $q = 0$, $\mathbb{P}(B < N-1) = 1 - 2^{-N} - N2^{-N}$, $\mathbb{P}(B = N-1) = N2^{-N}$, $\mathbb{P}(C > N-1) = 1$, $\mathbb{P}(C = N) = 0$. So we collect

$$N(1 - 2^{-N} - N2^{-N} + \frac{1}{2}N2^{-N} - 2^{-N}) = N(1 - 2^{-N}[2 + \frac{N}{2}]).$$

Next, we consider terms of the form

$$\mathbb{P}(A = k)|_{q=0} \cdot \frac{d}{dq}(\mathbb{P}(B < k) + \frac{1}{2}\mathbb{P}(B = k) - 2^{-N}(\mathbb{P}(C > k) + \frac{1}{2}\mathbb{P}(C = k)))|_{q=0}.$$

Note that $\mathbb{P}(A = k)|_{q=0} = 0$ for all $k < N$. The derivative of $\mathbb{P}(C > k)$ is $\frac{d}{dq}(1 - \mathbb{P}[C \leq k]) = -N\mathbb{P}[\text{Bin}(N-1, 1-q) = k]$. Evaluated at $q = 0$, this is 0 except when $k = N-1$, but in that case we have $\mathbb{P}(A = N-1) = 0$ when $q = 0$.

The derivative of $\mathbb{P}(C = k)$ evaluated at 0 is $-N$ for $k = N$, N for $k = N-1$, 0 otherwise. But $\mathbb{P}(A = N-1) = 0$ if $q = 0$, so we collect $1 \cdot (-2^{-N})\frac{1}{2}(-N)$.

Collecting the terms we have obtained, and adding up, we have that:

$$\begin{aligned}
&-N(1-2^{-N}) + N(1 - 2^{-N}[2 + \frac{N}{2}]) + (-2^{-N})\frac{1}{2}(-N) \\
=&N\left[-1 + 2^{-N} + 1 - 2^{-N}[2 + \frac{N}{2}] + \frac{2^{-N}}{2}\right] \\
=&N\left[2^{-N} - 2^{-N+1} - (N-1)2^{-N-1}\right] = -N(N+1)2^{-(N+1)}.
\end{aligned}$$

Overall, then, using the formula for the principal's payoff from Lemma 5, the derivative of payoffs evaluated at $q = 0$ is

$$hN - (1-h)N(N+1)2^{-(N+1)},$$

the sign of which equals the sign of $h/(1-h) - (N+1)2^{-(N+1)}$. $\square$

7.10 Proof of Proposition 6

Proof. Write $U_K(q)$ for the principal's expected utility from noise level q with K covariates, N observations, and h fraction of hackers. Write $U(q)$ for the principal's expected utility in the model with the same parameters from Section 3.2, but a continuum of covariates $A = [0, 1]$. From Proposition 5, $U'(0) > 0$, therefore there exists some $q' > 0$ so that $U(q') > U(0)$.

We argue that $U_K(q') > U(q')$ for every finite $K \geq 2$. Note that a maven has the same probability of proposing the true cause when $A = [0, 1]$ and when K is any finite number. This is because the maven's inference problem is restricted to only X^{a^*} and X^{a^r} and the presence of the other covariates does not matter. For the hacker's problem, note that the optimal behavior of the hacker is to propose the a that maximizes the number of observations where $\hat{X}^a$ matches the outcome variable Y in the noisy dataset. For a hacker who has no private information about a^*, such a covariate has the highest probability of being the true cause and the highest probability of passing the test. The principal's utility conditional on the hacker passing the test when $A = [0, 1]$ is -1, but this conditional utility is strictly larger than -1 when K is finite as the hacker has a positive probability of choosing the true cause. Also, the probability of the hacker passing the test with proposal a only depends on the number of observations where $\hat{X}^a$ matches Y, and the probability is an increasing function of the number of matches. When $A = [0, 1]$, the hacker can always find a covariate that matches Y in all N observations in the noisy dataset, but the hacker is sometimes unable to do so with a finite K. So overall, we must have $U_K(q') > U(q') > U(0)$.

Finally, we show that $U_K(0) - U(0) = h\left[2\frac{(1-[1-(1/2)^N]^K)}{(1/2)^N K} - 1\right] + h$, an expression that converges to 0 as $K \to \infty$. Clearly, if noise level is 0 and $A = [0, 1]$, then the principal's expected utility when facing the hacker is -1. For the case of a finite A, note X^{a^*} is perfectly correlated with Y. Each of the remaining $K - 1$ covariates has probability $(1/2)^N$ of being perfectly correlated with Y, so the number of perfectly correlated variables is $1 + B$, with $B \sim \text{Binom}((1/2)^N, K - 1)$.

The hacker will recommend a perfectly correlated action at random, so the recommendation is correct and yields of payoff of 1 with probability $1/(1 + b)$, and incorrect with probability $b/(1 + b)$, for each realization b of B. Hence the expected payoff from facing a hacker is $\mathbb{E}(\frac{1-B}{1+B})$. Using the calculation of $\mathbb{E}(1/(1 + B))$ in Chao and Strawderman (1972),

$$\mathbb{E}(\frac{1-B}{1+B}) = 2\mathbb{E}(\frac{1}{1+B}) - 1 = \frac{2(1-(1-p)^K)}{pK} - 1,$$

where $p = (1/2)^N$.

Combining the fact that $\lim_{K\to\infty}(U_K(0) - U(0)) = 0$ with $U_K(q') > U(q') > U(0)$, there exists some $\underline{K}$ so that $U_K(q') > U(q') > U_K(0)$ for every $K \geq \underline{K}$. $\square$

7.11 Proof of Proposition 7

Proof. First, there exists some $\bar{q}_1 > 0$ so that for any noise level $0 \leq q \leq \bar{q}_1$, the hacker finds it optimal to report a covariate a that satisfies $X_n^a = Y_n$ for every observation n in the data. Such a covariate has probability 0 of being correct but the highest probability of being implemented out of all covariates $a \in [0, 1]$. If the hacker instead reports $\varnothing$, the expected payoff is $1 - \beta$. When $q = 0$, the expected payoff from reporting a is $1 - w > 1 - \beta$ since $w < \beta$. The chance of such a covariate passing the policymaker's test is continuous in noise level, so there is some $\bar{q}_1 > 0$ so that for every noise level $0 \leq q \leq \bar{q}_1$, the hacker's optimal behavior involves reporting a covariate that perfectly matches the outcome in the noisy data.

This means for $0 \leq q \leq \bar{q}_1$, the principal's expected payoff with dissemination noise q when facing a hacker is $-V_{hacker}(q) = -(1-q)^N$, with $-V'_{hacker}(q) = N(1-q)^{N-1}$.

For any $0 \leq q \leq 1/2$, after the maven observes the two covariates $a_1, a_2 \in [0, 1]$ (one of them being a^* and the other being a^r, and there is some β probability that a^* is the true cause), it is optimal to either report the covariate $a \in \{a_1, a_2\}$ that satisfies $X_n^a = Y_n$ for a larger number of observations n, or to report $\varnothing$. To see that it is suboptimal to report any other covariate, note the maven knows that the correct report is either a_1, a_2, or $\varnothing$, and assigns some posterior belief to each. At least one of the three option must have a posterior belief that is at least $1/3$, therefore the best option out of a_1, a_2, or $\varnothing$ must give an expected payoff of at least $\frac{1}{3}w$. On the other hand, reporting a covariate $a \in [0, 1]\backslash\{a_1, a_2\}$ gives at most an expected payoff of $1 - w$. We have $\frac{1}{3}w > 1 - w$ by the hypothesis $w > \frac{3}{4}$.

We show that there is some $\bar{q}_2 > 0$ so that for any noise level $0 \leq q \leq \bar{q}_2$, if in the noisy dataset we have (i) $X_n^{a_1} = Y_n$ for all n, $X_n^{a_2} = 1 - Y_n$ for all n, or (ii) $X_n^{a_1} = Y_n$ for all n, $X_n^{a_2} = 1 - Y_n$ for all except one n; or (iii) $X_n^{a_1} = Y_n$ for all except one n, $X_n^{a_2} = 1 - Y_n$ for all n, then the maven reports a_1. It suffices to show that for small enough q, in all three cases the posterior probability of a_1 being the true cause exceeds $1/2$ (so that the expected utility from reporting a_1 exceeds that of reporting $\varnothing$). In case (i), this posterior probability is

$$\frac{0.5\beta(1-q)^{2N}}{0.5\beta(1-q)^{2N} + 0.5\beta q^{2N} + (1-\beta)(1-q)^N q^N},$$

which converges to 1 as $q \to 0$. In case (ii), this posterior probability is

$$\frac{0.5\beta(1-q)^{2N-1}q}{0.5\beta(1-q)^{2N-1}q + 0.5\beta q^{2N-1}(1-q) + (1-\beta)q^{N+1}(1-q)^{N-1}}.$$

Factoring out q from the numerator and the denominator, this converges to 1 as $q \to 0$. In case (iii), this posterior probability is

$$\frac{0.5\beta(1-q)^{2N-1}q}{0.5\beta(1-q)^{2N-1}q + 0.5\beta q^{2N-1}(1-q) + (1-\beta)q^{N-1}(1-q)^{N+1}}.$$

Factoring out q from the numerator and the denominator, this converges to 1 as $q \to 0$.

The principal's expected payoff from facing the maven is the probability that a true cause exists in the data and the maven reports a^*. This is because the maven either reports $\varnothing$ (so the principal gets 0), or reports a covariate that is either the true cause or gets rejected by the policymaker. When $q = 0$, the principal's expected payoff is β. A lower bound on the principal's payoff for $0 \leq q \leq \bar{q}_2$ is $L(q) := \beta \cdot \mathbb{P}[\text{noise level } q \text{ flips 0 or 1 entries in } X_n^{a^*}, X_n^{a^r}, 1 \leq n \leq N]$. If a^* is the true cause and the noise flips no more than 1 entry in $X_n^{a^*}, X_n^{a^r}$, then the maven sees one of cases (i), (ii), or (iii) in the noisy data, and by the argument before the maven will report a^* if $q \leq \bar{q}_2$. Note this lower bound is equal to the principal's expected payoff when $q = 0$.

We have

$$L(q) = \beta \cdot (1-q)^{2N} + 2N \cdot (1-q)^{2N-1} \cdot q.$$

The derivative is:

$$L'(q) = \beta \cdot [-2N(1-q)^{2N-1} + 2N \cdot (1-q)^{2N-1} - 2N \cdot (2N-1) \cdot (1-q)^{2N-2} \cdot q]$$

so $L'(0) = 0$. We have that $L(q) - V_{hacker}(q)$ is a lower bound on the principal's expected payoff with dissemination noise q for all $0 \leq q \leq \min(\bar{q}_1, \bar{q}_2)$, and the bound is equal to the expected payoff when $q = 0$. We have $L'(0) - V'_{hacker}(0) > 0$, therefore there exists some $0 < \bar{q} < \min(\bar{q}_1, \bar{q}_2)$ so that the lower bound on payoff $L(q) - V_{hacker}(q)$ is strictly increasing up to $\bar{q}$. This shows any noise level $0 < q < \bar{q}$ is strictly better than zero noise for the principal. □