Exploit or Explore?
An Empirical Study of Resource Allocation in Scientific Labs

Ran Zhuo*

October, 2022

CLICK HERE FOR LATEST VERSION

Abstract

Allocating innovation resources to their most productive uses is a challenge for innovators because they have incomplete information about which projects will be most productive. I empirically study how a group of large scientific labs traded off the exploitation of existing opportunities versus the exploration of new ones, that is whether they pursued safe projects to maximize short-term productivity or undertook high-variance projects to acquire information and improve long-term productivity. To recover how these labs made the exploitation-exploration tradeoff, I estimate a dynamic model of decision-making, assuming the labs approximated the value of exploration with a simple Upper Confidence Bound (UCB) index. The type of index is well-studied in theory and well-used in practice but has not been applied to estimation of empirical decision models. The index model captures the labs' decision-making well. Estimates of its free parameters suggest that the labs explored extensively. Counterfactual simulations show that, had the labs not explored, their output quantity would have decreased by 51%, and their citations would have decreased by 57%.

*Department of Economics, Harvard University. Email: rzhuo@g.harvard.edu. I am very grateful to my advisors Shane Greenstein, Myrto Kalouptsidi, Robin Lee, Ariel Pakes, and Elie Tamer for their time, patience, and insightful advice. I would also like to thank John Everett, John Norvell, and Peter Preusch for thoughtful discussions on the field of structural biology, the NIH, the Protein Structure Initiative, and the operations of structural biology labs. I thank Shengmao Cao, Varanya Chaubey, Chaim Fershtman, Yannai Gonczarowski, Tianxiao Han, John Hill, Ryan Hill, Charles Hodgson, Louis Kaplow, Max Kasy, Jacqueline Ng Lane, Lucas De Lima, Zhi Lin, Alexander MacKay, Kyle Myers, Frank Pinter, Devesh Raval, Tim Simcoe, Ken Simons, Chris Snyder, Paula Stephan, Jeff Strabone, Senmiao Sun, Wei Yang Tham, Audrey Tiew, Nataliya Langburd Wright, Hanbin Yang, Ron Yang, David Zhang, participants of the Harvard Industrial Organization workshop, WICK#8 Doctoral Workshop, EARIE 2022, and NBER Productivity Seminar for helpful suggestions. I thank Bob Freeman and staff at Harvard Research Computing for technical assistance and the Doctoral Office at Harvard Business School for financial assistance. All errors are my own.

1 Introduction

Innovators often need to make a tradeoff between exploitation and exploration in allocating innovation resources. How they do it has a large impact on their productivity. Exploitation prioritizes safe projects that innovators deem the most productive in the present. Exploration prioritizes high-variance projects that would yield more information to improve allocation and productivity in the future. Innovators need to make this tradeoff because they have incomplete information about which projects will be productive. The purpose of exploration is to acquire information about productivity—by allocating input to the projects that innovators have poor information about and observing those projects' output. Extensive theoretical literature, starting with seminal works of Gittins (1979) and Weitzman (1979), has recognized the importance of exploration in resource allocation for innovation and has theorized about optimal exploitation-exploration tradeoff.

Little research has studied empirically how innovators make this tradeoff. One reason is a lack of data with sufficient granularity, which needs to contain resources allocated to, and output produced by, every research project in an innovator's choice set. Another reason is the computational challenge. Innovators typically allocate resources among lots of projects over horizons that last years. Workhorse estimation methods for dynamic choice models, developed by an empirical literature starting with Pakes (1986) and Rust (1987), do not work well with large choice sets and long horizons due to the curse of dimensionality.

Employing novel data and proposing a new estimation approach, this paper studies empirically how a group of large scientific labs made the exploitation-exploration tradeoff. I find that a simple model captures the labs' decision-making and that the estimates of the model's free parameters suggest the labs explored extensively. I also find that, under counterfactual simulations, had the labs not explored, their output quantity and citations would have decreased substantially.

These answers are nonobvious and important. They confirm intuition from theoretical literature about the value of exploration for resource allocation under incomplete information. They also suggest sophistication in real-world decision-making—the labs took into account the dynamic incentives to explore and acquire information in their decision-making, as they should. Moreover, they have policy implications for the large innovation industry in which US spending per year exceeds $500 billion (Boroush, 2021). If these results extend to the whole industry in any degree,

exploration of ideas makes a substantial contribution to innovation productivity through the channel of improving the resource allocation process, and has generated hundreds of billions of dollars of economic value through this channel.

The empirical setting of this paper is a group of large structural biology labs funded by an NIH program of $1.3 billion over a horizon of sixteen years. This is an ideal setting to study because it is realistically complex and affords highly granular data. Moreover, the labs thoroughly documented how they used an AI-assisted approach to analyze information observed during past allocations and to inform future decision-making, providing guidance for modeling. A few additional NIH policy features, such as restricting the pool of projects the labs could choose from, also make this setting particularly clean for model identification.

The main data is the labs' resource allocation and output across projects. I observe, at the daily frequency, the input allocated to, and output produced by, each research project that each lab ever attempted over the sixteen-year horizon. A project in this setting is the determination of the three-dimensional structure of a protein molecule, and the molecule uniquely identifies the project. A unit of input is a distinct experimental trial on the project. The output is whether the trial produced a publication of the structure and the number of citations and downloads of that publication. Those are *observed* output from input actually allocated. The labs instead made allocation decisions based on their *beliefs* about output from potential allocations. To construct the labs' posterior beliefs about output, a key model ingredient, I made a best-effort replication of the labs' AI-assisted approach of belief formation, which was based on analyzing information observed during past allocations.

To recover how the labs made the exploitation-exploration tradeoff, I estimate a dynamic model of the labs' decision-making, assuming the labs used a simple index to approximate the complex underlying value function. This kind of index is well-studied in theory (Lai & Robbins, 1985; Agrawal, 1995; Katehakis & Robbins, 1995; Auer et al., 2002; Bubeck & Cesa-Bianchi, 2012) and well-used in practice (Yang et al., 2020; Scale, 2021; Zhang, 2021) but has not been applied to estimation of empirical models of decision-making. Under my model, in each period a lab first analyzes past information to form posterior beliefs about output from potential allocations. The lab then uses this posterior to compute a simple index to approximate the value function associated with allocating input to each project. The lab then allocates input to projects with the highest index

values. The index is simple to compute because computation only requires information available to the labs in the current period. Workhorse dynamic models based on backward induction, on the other hand, tie the value function to the evolution of information in *future* periods and, consequently, suffer from the curse of dimensionality as the number of periods grows. Due to its computational simplicity, index approximation is also a more empirically plausible model of the labs' decision-making than backward induction.

My model modifies a well-used index called Upper Confidence Bound (UCB). Intuitively, UCB approximates the upper confidence bound of an allocation's posterior expected payoff. Allocating input to a project that the lab has poor information about has a high UCB. As the lab allocates more input to the project and has better information about its productivity, the UCB of further allocations decreases. To proxy for the unobserved factors during the allocation process, such as project-specific learning staying with individual researchers over time, I make a modification to the baseline UCB index. The modification is an additional term that captures time-discounting of the value of older projects that the lab has attempted in the distant past. As robustness checks, I specify many alternative models with other types of indices discussed in the theoretical literature. These include the seminal Gittins (1979) index, which prescribes optimal decisions for some stylistic dynamic allocation problems.

I validate the model in two steps. In the first step, I estimate the free parameters in the model by maximizing the log likelihood of the observed allocation decisions. Identification of model parameters is based on revealed preferences and is very intuitive. I gauge model fit by computing the predicted likelihood of making the observed allocation decisions at convergence of the model. In the second step, I use the estimated parameters from the model to forward simulate the labs' entire history of input allocation and output. I compare the patterns of input allocated and output from the simulated data to those from the actual data to determine whether the model could generate patterns similar to those in the actual data. I repeat this two-step procedure for the alternative models to benchmark the fit of my main model against these alternatives.

I find my main model fits the data extremely well and captures the labs' decision-making. During the maximum likelihood step, the main model by far has the smallest magnitude of log likelihood at convergence among the many alternative models I tested. With the same number of parameters, its log likelihood at convergence is only 59% to 72% of the second best fit model

across labs. On average, the main model predicts a 70% to 84% likelihood of allocating trials that were actually allocated, and a smaller than 0.5% likelihood of allocating trials that were actually *not* allocated, far better than any alternative model. During the simulation step, the main model generates input allocation patterns and output that are very similar to those in the data. For all labs, the deviations of the simulated output from the actual output are within 10%. The alternative models fail to generate patterns matching the data as closely.

Based on this well-fitting model, I find that the labs explored extensively and that exploration had a large impact on the labs' productivity. Based on the estimates of the free parameters of the model, I am able to reject the hypothesis that the labs did not explore at a 95% confidence level for all labs. Moreover, I am able to reject the hypothesis that there were no unobserved factors that impacted the labs' incentives to explore a project over time at a 95% confidence level. Counterfactual simulations show that exploration boosted the labs' productivity substantially. Had the labs not explored, they would have missed many low-hanging fruit and misallocated resources towards less productive projects. As a result, their output quantity would have decreased by 51%, and their citations would have decreased by 57%. The decrease is equivalent to forgoing at least $650 to $720 million of economic value.

This paper contributes to the empirical literature on science of science. This literature has studied extensively the nature of idea development and the value of experimentation (Cohen & Levinthal, 1989; Henderson & Cockburn, 1996; Azoulay et al., 2011; Ederer & Manso, 2013; Ganglmair et al., 2019; Krieger, 2021; Hoelzemann et al., 2022; Khmelnitskaya, 2022; Lane et al., 2022), but few works have examined exploration of ideas as a micro-determinant of productivity at this level of detail. This paper is a rigorous empirical counterpart to the theoretical literature on resource allocation for innovation (Arrow, 1962; Roberts & Weitzman, 1981; Bergemann & Välimäki, 1996), and its findings add to the discussion of the impact of policies on innovation productivity (Jaffe, 2002; Furman & Stern, 2011; Azoulay, 2012; Cantoni & Yuchtman, 2014; Lane et al., 2015; Azoulay et al., 2019; Myers, 2020).

This paper also contributes to the empirical literature on single-agent dynamic choices. This literature has relied on recursive and simulation methods for estimation (Pakes, 1986; Rust, 1987; Hotz & Miller, 1993; Rust, 1994; Timmins, 2002; Aguirregabiria & Mira, 2010) which face significant computational challenges with large state-spaces and long horizons. This paper develops the

index approximation approach to model dynamic allocations, which overcomes the computational challenges in estimation. An extensive theoretical literature on dynamic allocation inspires this approach. Theoretical literature has often called dynamic allocation problems like that in our setting "multi-armed bandits" and has studied models of optimal exploitation-exploration tradeoff in the form of indices. Gittins & Jones (1979), Weitzman (1979), Lai & Robbins (1985), Whittle (1988), Bergemann & Välimäki (1996), Bolton & Harris (1999), Auer et al. (2002), Keller et al. (2005), Bubeck & Cesa-Bianchi (2012), and Russo et al. (2017) are some works in this literature. A couple of empirical works have analyzed bandit-like problems in other economic settings (Miller, 1984; Erdem & Keane, 1996; Crawford & Shum, 2005; Dickstein, 2018; Li et al., 2020; Caria et al., 2020). The index approximation approach formalizes many methodological insights from those earlier papers.

The rest of this paper is organized as follows. Section 2 discusses the empirical setting. Section 3 describes the data. Section 4 builds the model. Section 5 describes model fitting. Section 6 shows model fitting results. Section 7 shows counterfactual results. Section 8 concludes.

2 Empirical Setting

The empirical setting of this paper is a group of large structural biology labs funded by a $1.3 billion NIH program from 2000–2015. This is an ideal setting to study for many reasons. First, this setting captures the complexity of decision-making typical in resource allocation for innovation and affords highly granular data. Second, the labs thoroughly documented how they used information observed during past allocations to inform future decision-making, providing guidance for modeling. A few additional NIH policy features also make this setting particularly clean for model identification.

A particularly clean feature of this setting is that research projects are clearly defined and distinct from each other, making it easy to compile highly granular data that associates input and output with specific projects. Moreover, the labs recorded input allocation in discrete units, making it easy to track the amount of input allocated to each project.

A project in this setting is the determination of the three-dimensional structure of a protein molecule, and the molecule uniquely identifies the project. These projects are important research.

Knowing a protein's structure is often critical for developing drugs that target the protein. Proteins consist of building blocks called amino acids. These amino acids are arranged into a chain, which folds up onto itself, creating a three-dimensional structure (Hill & Stein, 2020, 2021). Figure 1 shows one structure, published by one of the labs in my data. Publications of protein structures have aided the development of therapeutics such as oncology drugs (Van Montfort & Workman, 2017) and COVID-19 vaccines (Wrapp et al., 2020).

A unit of input in this setting is a distinct experimental trial of the project. Each trial is a bundle of labor, capital, and materials associated with working on the molecule. A trial proceeds in multiple sequential stages. The outcome of each stage is either a success or a failure. If a trial succeeds in all stages, it produces a unit of output—a publication of the molecule's structure. Trials are very risky. 98% of trials in my data failed at various stages, and there were large variations in trial outcomes even within projects.[1] Output is thus a random variable. I will assume that a trial stops if and only if the trial fails, which means the labs do not actively kill ongoing trials. This assumption significantly simplifies the allocation problem and is mostly supported by the patterns in the data.[2]

This setting captures a realistically complex resource allocation problem over a long horizon. Figure 2 illustrates the decisions a structural biology lab faces. In each period, the lab decides to allocate units of input among a portfolio of research projects. The portfolio contains older projects the lab has worked on in the past, and the lab can draw new projects to include in the portfolio. The objective is to maximize over the horizon welfare from the output. The lab or the NIH determines the welfare from each unit of output according to some welfare weights unknown to us the economists. A key challenge in the lab's decision-making is that trial outcomes are very noisy and the lab has incomplete information about which trials are likely to produce output.

The labs used information they observed during past allocations to inform decision-making and thoroughly documented the process, providing guidance for modeling for this paper (Slabin-

[1]Much of the work in these trials involves trial-and-error of different combinations of experimental conditions for physicochemical reactions. As a result, researchers often need to perform many trials for a project before successfully producing a structure for publication. Luck is an important element in the process (Chruszcz et al., 2008).

[2]For 15% of the trials, I observe the reasons for trial terminations. The reasons are mostly exogenous, e.g., "expression failed", "purification failed", "poor diffraction" and so on. In some cases, the trial termination reason is "duplicate target found" when the labs found a highly similar project was already published. Upon further investigation, I find the ongoing trials in most of those cases were allowed to keep going. Some of those trials were even successful at all stages and produced publications.

ski et al., 2007a,b; Jaroszewski et al., 2008; Price et al., 2009a,b; Babnigg & Joachimiak, 2010; Jahandideh et al., 2014). Figure 3 illustrates the process. As a lab performs trials and observes their outcomes, it collects and organizes the information about the trials' characteristics and outcomes. The lab then analyzes that information, fitting machine learning models to draw correlations between trial characteristics and outcomes. The fitted models allow the lab to predict a new trial's probability of producing output based the trial's characteristics. That prediction forms the lab's posterior belief about the trial's output. The lab then allocated trials based its posterior beliefs about different trials' output and its welfare weights. As the lab performs more trials and observes more information over time, it refits its machine learning models periodically.

Several NIH policy features simplify this setting and mitigate worries about modeling and identification, one of them being limits on the labs' choices of projects. The NIH periodically drew families of novel molecules and solicited nominations of molecules from the biomedical research community for the labs to work on. The labs could also pursue projects of their own interest but had to communicate with the NIH about those projects well in advance (NIGMS, 2007a,b, 2011b). Those processes limited the projects in the labs' choice sets, allowing me to construct choice sets that align well with the actual ones.

Another feature is the collaborative nature of the grant program, which mitigates worries about competition between the labs confounding with the exploitation-exploration tradeoff. The NIH funded the program, called the Protein Structure Initiative (PSI), through a collaborative U01 mechanism, rather than the competitive R01 mechanism. Throughout the program, the labs faced little competition for funding (and research questions due to the NIH assigning projects).[3]

The NIH also evaluated the labs' productivity periodically based on a set of metrics,[4] which limited what the labs placed nonzero welfare weights on. Metrics included the quantity of structures published, novelty, biomedical importance, human proteins, eukaryotic proteins,[5] and mem-

[3]A more typical lab faces additional tradeoffs due to competition. Hill & Stein (2021) found structural biology labs traded off quality of research and speed of publication because of competition to establish priority in publication. However, they also found the structural biology labs in my data (labelled as " structural genomics (SG)" labs in their paper) were not as motivated by competition to sacrifice quality for speed.

[4]The NIH published statistics on those metrics online. The archived versions of those publications can be found by searching the urls http://targetdb.pdb.org/metrics/milestonestables.html and http://targetdb.pdb.org/Metrics/SummaryTable.html on the Internet Archive.

[5]Eukaryotes include all living organisms other than the eubacteria and archaebacteria. An eukaryote is an organism consisting of a cell or cells in which the genetic material is DNA in the form of chromosomes contained within a distinct nucleus.

brane proteins.[6]

Moreover, an exogenous change of NIH's preferences over the metrics in 2009 provides a nice robustness check. A well-fitting model of the labs' decision-making should capture this shift in its estimates of the labs' welfare weights. Before 2009, the NIH had a strong preference for the quantity of publications and novelty (NIGMS, 2008a, 2011a). By mid-2008, the lack of emphasis on biomedically important projects had sparked heated debates in the community (Petsko, 2007; Moore, 2007); this prompted the NIH to boost its preference for biomedically important projects since 2009 (NIGMS, 2007c, 2008b, 2009a,b). To facilitate the change, the NIH partnered with outside researchers to identify biomedical important projects and gave more attention to biomedical importance in its evaluation process.

In addition, the NIH's close involvement in those labs makes it reasonable to assume away the principal-agent problem, which further simplifies the setting.

3 Data

The main data in this paper is those structural biology labs' resource allocation and output across projects. I observe, at the daily frequency, the input allocated to and output produced by each research project each lab ever attempted over the horizon. Highly granular data is available thanks to an NIH requirement for data collection and sharing.[7] Available in this data are all projects in a lab's portfolio, all trials performed on each project, dates and outcomes of stages of each trial, output of structures (if any), and identifiers of projects and structures to link to other bioinformatics databases. Labs could see each others' data through a shared information system at all times.

This section describes the key variables on input allocation, observed output, and the labs' posterior beliefs about output. Table 1 shows a list of the key variables and their sources. Appendix A describes variable construction and shows additional variables.

[6]Membrane proteins are proteins found in the cell membrane. Membrane proteins are particularly hard for structure determination due to their physicochemical properties.

[7]This data includes 27 labs that received NIH funds and 13 international labs that contributed data on a voluntary basis. A lab was typically a consortium that had sub-labs handling different stages of the trials. The number of sub-labs captured in this data is 147.

3.1 Input Allocation

The input allocation data consists of units of input (trials) each lab allocated to individual projects on each day, and those projects' and trials' characteristics.

A project is identified by the distinct biological molecule it studied. I use the letter i to denote a project. The data contains 335,553 projects.

A trial is identified by its association to a specific project and the order in which the trial occurred. I use j_i to denote the jth trial of project i. The data contains 961,260 unique trials and captures 3,783,026 records of trial progress updates at the daily level. 99% of these records were between 2000 and 2015. Figure 4 shows a distribution of the number of trials allocated to each project. The distribution has a prominent right skew. 71% of projects had only a single trial. In contrast, less than 4% of projects had more than 10 trials. This pattern is consistent with extensive exploration, where the labs sampled projects they had poor information about in input allocation to learn which of those projects would be promising.[8]

Different labs in this data operated on very different scales, and I will focus on analyzing the four largest labs.[9] These four labs accounted for 71% of the projects and 85% of the trials in this data. The other labs were considerably smaller and had high variations in their research interests and funding levels.

I use n_{lt} to denote the amount of input available for allocation at lab l on day t and let n_{lt} equal the actual number of trials allocated at the lab on that day. For the four largest labs, n_{lt} has a mean of 35 trials per day per lab.

Several variables capture a lab's rationale for allocating a trial to a project, and these variables correspond to the NIH's evaluation metrics for the labs' productivity, among which a key metric is the novelty of the project. The variable $novel_i$ is a binary and is equal to 1 if the labs cited novelty

[8]The high proportion of single-trial projects could be consistent with an allocation model that only exploits, but that would require the first trial of each project to provide a lot of information about the project. It is possible that the labs started the first trials of a lot of new projects thinking the projects were high-payoff. Then they learned enough information about each project's potential from a single trial to identify low-payoff projects whose future trials were unnecessary. However, given that trial outcomes were very noisy, it is unlikely that the first trial would convey so much information to generate the observed patterns. Moreover, with updating of posterior beliefs, it is unclear how the labs sustained the belief that new projects were high-payoff under this model over time, given that most new projects turned out to be low-payoff ones after a single trial.

[9]They are Joint Center for Structural Genomics (JCSG), Midwest Center for Structural Genomics (MCSG), New York Structural Genomics Research Consortium (NYSGRC), and Northeast Structural Genomics Consortium (NESG).

as a reason to allocate trials to project i in the information system.[10] When the labs cited a project i as being novel, they often emphasized that there were no or few already published structures in the same protein family as i. I therefore construct $prevStruct_{iy}$, a continuous variable that captures the changes over year (subscripted with the letter y) of the number of published structures in the same protein family as i.

Another key NIH evaluation metric is the biomedical importance of the project. The variable $biomed_i$ is a binary and is equal to 1 if the labs cited biomedical importance as a reason to allocate trials to project i in the information system.[11] As an additional proxy for the biomedical importance of a molecule, I look into the number of publications related to the molecule, including structures and other types of publications. I construct $prevPub_{iy}$, a continuous variable that captures the changes over year of the number of publications on molecule i.

Additional NIH evaluation metrics correspond to whether the project was related to human beings, eukaryotes,[12] and the cell membrane. The variable $human_i$ captures how similar molecule i is to any molecules from human beings.[13] The variable $eukaryote_i$ captures how similar molecule i is to any molecules from eukaryotes.[14] The variable $membrane_i$ is a binary and is equal to 1 if molecule i is related to the cell membrane.

I also obtain the dollar value of funds each lab received as an alternative measure of input. The variable $funding_{ly}$ captures the total amount of funds lab l received from the NIH in year y. Funding information comes from two sources. First, the NIH released a series of funding opportunity announcements (FOAs) directly tied to the grant program (NIH, 2019),[15] which allows me to search directly all grants associated with those FOAs on NIH RePORT database (NIH, 2021). Second, labs sometimes received supplementary funds from the NIH so I also perform a direct search of the labs' names and abbreviations using RePORT's advanced search functionality. I then

[10] The information system contains textual descriptions of why labs allocated trials to a project. The relevant fields are populated for 84% of projects at the four largest labs. Construction of $novel_i$ and $biomed_i$ is based on keywords in those descriptions. See the full list of keywords and additional details in Appendix A.1.

[11] Same as footnote 10.

[12] See definition in footnote 5.

[13] When a lab worked on a "human" molecule, often the molecule was actually from bacteria but was very similar to a molecule from human beings and was much easier than the human molecule. Therefore, the right construction for $human_i$ is molecule i's degree of similarity to human molecules rather than being a human molecule itself. I learned this from a conversation with an NIH program officer in charge of the grant program.

[14] Same as footnote 13.

[15] See the full list of FOAs in Appendix A.2.

aggregate each lab's sum of research grants by year from the search results. Figure 5 shows the level of funding for the four largest labs.

3.2 Observed output

The main measure of output is the number of unique publications. A trial produced a publication if it succeeded in all stages. I use a binary Y_{ijt} to denote the outcome of trial j_i on day t. $Y_{ijt} = 1$ if the trial succeeded and produced a publication. Only 1.6% of trials in the data succeeded. I also observe the outcomes of all stages of each trial and use a binary Y_{ijkt} to denote the outcome of stage k of trial j_i on day t.[16] I use these intermediate outcomes to construct the labs' posterior beliefs about output.[17]

Uniqueness of a publication is important because more publications do not necessarily mean more welfare when lots of publications are duplicates of earlier ones. Duplicate publications happened in our setting because a lab often allocated multiple trials on the same project at roughly the same time and sometimes more than one trial succeeded. These successful trials produced structures of the same molecule often of very similar qualities. I will therefore only count the first publication on each project in measuring a lab's output. The data contains 15,848 publications of structures, among which 10,501 are unique. This represents 15% of world output of structures between 2000 and 2015.

I supplement the main output measure with the number of citations and downloads of each published project. The variable $citation_{iy}$ captures the five-year citations and mentions of a project i published in year y.[18] The variable $download_{im}$ captures the number of downloads of a project i

[16] $k = 0,1,2,3,4$. $Y_{ij0t} = 1$ if DNA was successfully cloned. Y_{ij1t} is only defined when $Y_{ij0t} = 1$ and is equal to 1 if protein was successfully expressed. Y_{ij2t} is only defined when $Y_{ij0t} = 1$ and $Y_{ij1t} = 1$ and is equal to 1 if protein was successfully purified. Y_{ij3t} is only defined when $Y_{ij0t}, Y_{ij1t}, Y_{ij2t} = 1$ and is equal to 1 if protein was successfully prepared for studying its structure (through X-ray crystallography or NMR or cryo-EM). Y_{ij4t} is only defined when all previous stages were successful and is equal to 1 if the structure was successfully produced and deposited to the Protein Data Bank (PDB) for publication.

[17] See Section 3.3 and Appendix B.

[18] When multiple structures of the same project were published, I take the mean of five-year citations and year of publication of those structures. I do not take the sum of five-year citations because those structures were often cited together.

in month m between August 2007 and November 2013.[19]

The two measures capture quality in different dimensions. Citations in the academic literature reflect the amount of follow-on research directly built on the published project. Downloads offer a window into the more intangible, such as the amount of interest generated by the publication and the use of the publication in developing new technologies in related fields (for example, DeepMind (2020)).

Figure 6 shows the observed output for the four largest labs.

3.3 Posterior Beliefs about Output

A key model ingredient of this paper is the labs' posterior beliefs about output from potential allocations at the moment of making allocations.

My measures of the labs' posterior beliefs about output are based on best-effort replications of the labs' documented approach of belief formation using information they observed during previous allocations. Variable construction uses several hundreds of variables, coming from a variety of sources. A full list of these variables is available in Appendix A.4.

The four largest labs formed and updated their posterior beliefs about whether a trial would succeed through supervised machine learning (Slabinski et al., 2007a,b; Jaroszewski et al., 2008; Price et al., 2009a,b; Babnigg & Joachimiak, 2010; Jahandideh et al., 2014). This involved using machine learning models to fit observed trial outcomes (successes or failures) on those trials' characteristics. The fitted model would be able to predict a new trial's probability of success based on the trial's characteristics. The labs periodically refitted the models when newer trial data became available to make updates. The labs started using this approach to form posterior beliefs as early as the beginning of 2005, when they had accumulated a considerable amount of trial data in the initial years of operation (Slabinski et al., 2007a,b). Over time and lab-wise, they used a variety of models, ranging from logistic regressions to support vector machine to random forest. Jahandideh et al. (2014) conducted a comparison study and found random forest worked best based on multiple

[19]This variable is not directly useful because downloads have strong lifecycle trends—downloads were high when a structure just became published then fell over time—and comparing different publications at different stages of their lifecycles is meaningless. In Section 3.3, I construct a variable corresponding to the five-year downloads to make comparison across projects meaningful. For that variable, when multiple structures of the same project were published, I take the mean of five-year downloads and year of publication of those structures. I do not take the sum of five-year downloads because those structures were likely downloaded together as they were often cited together.

metrics, including prediction accuracy.[20]

To mimic how the labs updated their posterior for trial success probability, I fit a series of machine learning models over time between 2005 and 2015. To fit a model corresponding to the labs' posterior beliefs at time t, I use a training sample consisting of trial outcomes realized before t and those trials' characteristics. The characteristics include the ones the labs used and fall under three categories:

- Physicochemical properties of the molecule based on scientific reasoning.
- Other characteristics of the project, such as novelty, biomedical importance, and the number of prior publications on the molecule.
- Past successes and failures of trials on the same project.

I use random forest to fit each model. Appendix B.1 shows the details.

The fitted models allow me to construct two variables, the posterior expectation of the probability of success $\widehat{E}_{\tilde{F}_t}(p_{ijt})$ and the posterior variance $\widehat{Var}_{\tilde{F}_t}(p_{ijt})$. Let the probability of success of a trial j_i at time t be p_{ijt}. The fitted model $\tilde{F}_t$ makes predictions about p_{ijt} based on the trial's characteristics, and these predictions constitute the posterior belief about p_{ijt}. The fitted model $\tilde{F}_t$ is a random forest consisting of an ensemble of submodels, each called a decision tree. Each decision tree fits the training sample independently and makes an independent prediction $\hat{p}_{ijt}^{ntree}$. The collection of these predictions forms the estimated posterior distribution. The mean of this distribution is the posterior expectation $\widehat{E}_{\tilde{F}_t}(p_{ijt})$, and the variance is the posterior variance $\widehat{Var}_{\tilde{F}_t}(p_{ijt})$.

My best-effort replication may deviate from the labs' actual posterior beliefs, but I do not expect the deviations to bias estimation results of the allocation model. I note a list of possible deviations in Appendix B.1. One example is that the labs did not always predict $\widehat{Var}_{\tilde{F}_t}(p_{ijt})$. When they did, the variable took the form of comparing predictions from multiple models side by side (Slabinski et al. (2007a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)). It is reasonable to believe the labs knew the value of analyzing the variation in predictions from different models, though they did not formally use a variable to represent that variation. $\widehat{E}_{\tilde{F}_t}(p_{ijt})$ and $\widehat{Var}_{\tilde{F}_t}(p_{ijt})$ therefore contain errors in the sense they may differ from the labs' actual posterior

[20]Based on my conversation with a project coordinator at one of the labs, the accumulation of trial data rather than the choice of the machine learning model was the main driver in improving the quality of predictions.

beliefs. But as long as the errors are random or at least not correlated with the actual allocation decisions, they should not bias the estimates of the allocation model. The errors may be correlated with the actual allocations if I fail to include in the training sample variables the labs actually used to form beliefs or make allocations. To minimize this risk, I include in my training samples all variables the labs ever mentioned using and all NIH evaluation metrics.

For the posterior beliefs about citations and downloads, I make the simplifying assumption that the labs' posterior beliefs stayed the same in all periods and were the same as the ground truth. This is equivalent to assuming that the labs only had incomplete information about the possibility of production and had perfect information about the kind of output produced if production does happen. Given that the projects were very well-defined, this assumption is reasonable. Another evidence in support of this assumption is that, unlike the probability of success of a trial, the labs did not form and update their posterior beliefs about the number of citations and downloads of a publication in a systematic way.

I use ridge regressions to model the posterior expectations about citations $\hat{E}(citation_{iy})$ and about downloads $\hat{E}(download_{iy})$. To align well with the ground truth, these models have to be able to make small prediction errors out of the training sample. Appendices B.2 and B.3 show details of model fitting and visual evidence of out-of-sample model fit in cross validation.

Figure 7 shows the estimated posterior beliefs and provides descriptive evidence that the labs explored high-variance projects in resource allocation and, in doing so, forwent some opportunities to exploit projects with high posterior expected output.

Figure 7a shows the distribution of $\hat{E}_{\tilde{F}_t}(p_{ijt})$ for trials the four largest labs actually allocated versus that for a random sample of trials in those labs' choice sets. While the distribution of $\hat{E}_{\tilde{F}_t}(p_{ijt})$ for the actual allocations is less right-skewed than that of the possible allocations, a considerable proportion of actual allocations had extremely low posterior expected probabilities of success. This is unlikely due to exhaustion of good possible allocations with high posterior expected probabilities of success. The actual allocations accounted for only 0.7% of the random sample of the possible allocations.

Figure 7b shows the distribution of $\sqrt{\widehat{Var}_{\tilde{F}_t}(p_{ijt})}$. The figure shows that on average the actual allocations had considerably more posterior variance in the probability of success than the possible allocations, providing evidence that the labs allocated a considerable amount of resources to high-

variance projects.

Figures 7c and 7d show that the distributions of $\hat{E}(citation_i)$ and $\hat{E}(download_i)$. Both show the trials the labs allocated to were not very different from the trials the labs passed in terms of expected citations and downloads they would generate upon publication. This suggests choosing high-variance projects was associated with not being chiefly motivated by citations and downloads, providing evidence that the labs allocated to high-variance projects for the sake of exploration.

4 Model

I present an estimable model of the labs' allocation decision-making whose key feature is index approximation of value function. Under this model, in each period a lab first analyzes past information to form posterior beliefs about output from potential allocations. The lab then uses this posterior to compute a simple index to approximate the value function associated with allocating input to a project. The lab then allocates input to projects with the highest index values. This model has the advantage of being computationally tractable and of being empirically plausible, in the sense that the labs could have easily applied this model in actual decision-making.

I build intuition for the model as follows. First, I formalize the labs' input allocation problem into an objective function. I then discuss the intractability of a standard model of decision-making based on backward induction. I explain how index approximation differs from the standard model and why it is more tractable. I wrap up with a discussion of the theoretical foundation for index approximation.

I define the main model as well as several alternative models for robustness checks at the end of this section.

4.1 Objective Function

I formalize the labs' input allocation problem into an objective function that maximizes welfare from output over a finite time horizon. This objective function frames the allocation problem in line with the theoretical literature.

I use C_{lt} to denote a lab l's choice set on day t. This choice set consists of further trials of older projects already in the lab's portfolio and initial trials of new projects. For example, consider a day

t where the lab has capacity to allocate $n_{lt} = 2$ new trials. Suppose the lab has had three trials of project i up to day $t-1$; then the fourth and fifth trials of project i are in day t's choice set. For another project i' the lab has received from the NIH and has not tried up to day $t-1$; the first and second trials of project i' are in day t's choice set.

I use a_{ijt} to denote input allocation decisions. $a_{ijt} \in \{0,1\}$ where 1 represents allocating trial j_i to project i on day t and 0 represents not allocating j_i on day t. The vector $\boldsymbol{a}_{lt}$ has a length equal to the cardinality of the choice set C_{lt} and represents the action on each of the project-trials in the choice set.

I use Y_{ijt} to denote the outcome of trial j_i on day t. $Y_{ijt} = 1$ if the trial succeeded and produced a publication. I use Y_{ijkt} to denote the outcome of stage k of trial j_i on day t.

I use Ω_t to denote the labs' information set on day t. Because the labs could observe each others' allocations and output at all times, all labs share the same information set on a day. Ω_t includes the actions $\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_{t-1}$ and outcomes and stage-specific outcomes $\boldsymbol{Y}_1, ..., \boldsymbol{Y}_{t-1}$ observed before t from all labs.

I use p_{ijt} to denote the probability of success of trial j_i on day t. The labs and the economist have incomplete information about p_{ijt}. Let the prior distribution of p_{ijt} be $F_t(p_{ijt})$. The labs formed and updated posterior $\tilde{F}_t(p_{ijt}|\Omega_t)$ with supervised machine learning, and I made best-effort replication of the labs' posterior offline in Section 3.3.

I use $\pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})$ to denote the payoff (in welfare units) from project-trial j_i. It depends on three variables. $\pi_{ijt}(\cdot)$ depends on the actions $\boldsymbol{a}_{lt}$. Whenever $a_{ijt} = 0$, $\pi_{ijt}(\cdot) = 0$ because a trial not allocated does not pay off. When $a_{ijt} = 1$, $\pi_{ijt}(\cdot)$ also depends on p_{ijt}, the probability of success of the trial. $\pi_{ijt}(\cdot)$ also depends on $\boldsymbol{\theta}_{Xl}$, which are welfare weights predetermined by the lab.

$\pi_{ijt}(a_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})$ is not the same as output. One can think of it as a utility function that translates output into welfare according to the welfare weights $\boldsymbol{\theta}_{Xl}$. For example, consider a trial j_i that would produce a novel and biomedically important structure upon its success. The structure is a unit of output. Suppose the lab has a welfare weight of 5 on a novel structure and a welfare weight of 3 on a biomedically important structure; the welfare the lab receives from this trial upon its success is 8. The lab knew its welfare weights but the economist does not, so I will estimate $\boldsymbol{\theta}_{Xl}$ based on the lab's observed allocations. I describe the estimation procedure in Section 5.1.

A lab's objective is to maximize the posterior expected payoff over the horizon by choosing a sequence of actions. The objective function is as follows:

$$\begin{aligned}\max_{\boldsymbol{a}_{l1}(\Omega_1),\ldots,\boldsymbol{a}_{lT}(\Omega_T)} & \sum_{t=1}^{T} \sum_{j_i \in C_{lt}(\Omega_t)} \int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t), \\ & \text{subject to} \sum_{j_i \in C_{lt}(\Omega_t)} a_{ijt} = n_{lt}, \\ & \text{and for all } j_i < j_i' \in C_{lt}(\Omega_t), \text{ if } a_{ijt} = 0 \text{ then } a_{ij't} = 0.\end{aligned} \tag{1}$$

The first constraint is the capacity constraint. The second constraint rules out allocating a fifth trial to a project when the fourth trial has not been allocated.

4.2 Intractability of Backward Induction

The most standard way of modeling how agents choose actions in a problem like equation (1) is to assume they specify the value function of different actions as a Bellman equation and solve the Bellman equation by backward induction. The value function of a state is the value attained by equation (1) at this state at the sequence of optimal actions $\boldsymbol{a}_{l1}^*, \boldsymbol{a}_{l2}^*, \ldots, \boldsymbol{a}_{lT}^*$. The state variable in our setting is the information set Ω_t.

The Bellman equation has two additively separable components, one component representing the posterior expected payoff in the current period and the other representing the continuation value. We can define the action-specific value function $V_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})$ of project-trial j_i on day t as follows

$$V_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}) = \underbrace{\int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t)}_{\text{posterior expected payoff}} + \underbrace{E_{\Omega'_{t+1}}[\max_{\boldsymbol{a}_{l,t+1}} V_{i,j',t+1}(\Omega'_{t+1}, \boldsymbol{a}_{l,t+1})|\Omega_t, \boldsymbol{a}_{lt}]}_{\text{continuation value}}. \tag{2}$$

The value function $V_{ijt}(\Omega_t; \boldsymbol{\theta}_{Xl})$ of project-trial j_i on day t is equal to $V_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})$ evaluated at the optimal $\boldsymbol{a}_{l1}^*, \boldsymbol{a}_{l2}^*, \ldots, \boldsymbol{a}_{lT}^*$.

As the continuation value integrates over the future evolutions of the state, solving the Bellman equation with backward induction and obtaining the optimal sequence of actions $\boldsymbol{a}_{l1}^*, \boldsymbol{a}_{l2}^*, \ldots, \boldsymbol{a}_{lT}^*$ is computationally intractable in our setting due to the curse of dimensionality. The state variable

Ω_t consists of all previous actions and outcomes. The number of possible actions on a day are sometimes on the order of millions. The time horizon has thousands of days. The number of possible states at the end of the horizon is astronomical.

4.3 Index Approximation

I present an alternative way of modeling how the labs chose actions, that is, assuming they used a simple index to approximate the value function.

Index approximation overcomes the curse of dimensionality because the approximated continuation value does not integrate over the future evolutions of the state. Let $V^A_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_l)$ represent the lab's approximation to the action-specific value function in equation (2),

$$V^A_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_l) = \underbrace{\int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t)}_{\text{as before}} + \underbrace{B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Bl})}_{\text{approximation to continuation value}} . \tag{3}$$

It has two additively separable components in direct parallel to the Bellman equation. The first component represents the posterior expected payoff in the current period and is identical to that in equation (2). The second component is an unknown function that the lab is assumed to use to approximate the continuation value. The key difference between this component and the continuation value in equation (2) is that the former is not a function of the future state Ω'_{t+1}. The lab is assumed to use a heuristic $B_{ijt}(\cdot)$, such as the posterior variance of the payoff, to approximate the value of continuing exploration of a project, and this heuristic is based only on information available to the lab at time t. Theoretical literature would call $B_{ijt}(\cdot)$ the "exploration bonus."

Because the computation of the approximation term only uses information currently available, the lab can solve equation (3) directly, without the need to backward induct. Assume the lab chose actions based on the approximate value function, acting as if the approximate value function was the true value function, it would simply choose actions $\boldsymbol{a}^{A*}_{lt}$ on each day t to maximize the sum of

the approximate values on that day, as follows:

$$\begin{aligned}\boldsymbol{a}_{lt}^{A*} = &\underset{\boldsymbol{a}_{lt}}{\operatorname{argmax}} \sum_{j_i \in C_{lt}(\Omega_t)} V_{ijt}^{A}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_l), \\ &\text{subject to} \sum_{j_i \in C_{lt}(\Omega_t)} a_{ijt} = n_{lt}, \\ &\text{and for all } j_i < j_i' \in C_{lt}(\Omega_t), \text{ if } a_{ijt} = 0 \text{ then } a_{ij't} = 0.\end{aligned} \tag{4}$$

With some very intuitive functional form assumptions about the payoff function $\pi_{ijt}(\cdot)$ (see Appendix C.1 for details), we can make $V_{ijt}^{A}(\cdot)$ only a function of a_{ijt}, not the full $\boldsymbol{a}_{lt}$ vector, and make the second constraint always holds at the solution. With those assumptions, $\boldsymbol{a}_{lt}^{A*}$ is equivalent to an index rule: the lab computes index $V_{ijt}^{A}(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l)$ for each project-trial in C_{lt}, and $\boldsymbol{a}_{lt}^{A*}$ are just allocating to the n_{lt} trials with the highest index values.

Unlike actions based on backward induction, actions made using the index approximated value functions are not optimal. Actions made in this way just represent what might produce a reasonably good payoff. Extensive literature on dynamic allocation provides theoretical grounding for approximating the value function using an index in our setting.

Equation (1) is a finite-horizon multi-armed bandit (MAB). Allocation problems characterized by objective functions like equation (1) are called multi-armed bandits because of a classic example of this type of problem. Imagine you are playing a five-armed bandit slot machine and you have ten opportunities to pull. In what way should you allocate your pulls across arms to maximize the expected total payoff if you have little prior information about the payoff of each arm? Intuitively, you want to use a few pulls to sample payoffs from different arms to learn which arms are promising then use the remaining opportunities to pull those arms. In our setting, each project is an arm and each trial of a project is a pull of an arm.

Theoretical literature has found optimal or nearly optimal indices for some MABs. In the case of the standard infinite-horizon discounted MAB,[21] the optimal index is the Gittins index (Gittins & Jones, 1979; Gittins, 1979). In the case of the standard finite-horizon MAB defined by the

[21] Defined by the objective function

$$\max_{\boldsymbol{a}_{l1}(\Omega_1), \boldsymbol{a}_{l2}(\Omega_2), \ldots} \sum_{t=1}^{\infty} \beta^t \sum_{j_i \in C_{lt}(\Omega_t)} \int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}^{(i)}(p_{ijt} | \Omega_t^{(i)}), \text{subject to} \sum_{j_i \in C_{lt}(\Omega_t)} a_{ijt} = 1 \text{ for all } t. \tag{5}$$

objective function

$$\max_{\boldsymbol{a}_{l1}(\Omega_1),\ldots,\boldsymbol{a}_{lT}(\Omega_T)} \sum_{t=1}^{T} \sum_{j_i \in C_{lt}(\Omega_t)} \int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}^{(i)}(p_{ijt}|\Omega_t^{(i)}),$$
$$\text{subject to} \sum_{j_i \in C_{lt}(\Omega_t)} a_{ijt} = 1 \text{ for all } t, \tag{6}$$

Lai & Robbins (1985) and Lai (1987) show that index rules do not provide exact solutions but are asymptotically optimal as the number of periods T goes to infinity, and have nearly optimal performance from both the Bayesian and frequentist viewpoints for moderate and small T. These nearly optimal index rules can be interpreted as the upper confidence bounds (UCB) for the posterior expected payoffs. Their results inspired a literature on indices that approximate the UCB (Agrawal, 1995; Bubeck & Cesa-Bianchi, 2012; Cappé et al., 2013). The simplest UCB index is just a function of the number of previous pulls of the arm (Auer et al., 2002).

Equation (1) differs from the standard finite-horizon MAB in a few ways, and, as a result, theoretical literature has yet to find an optimal index for it, but the literature has also shown indices may be good heuristics in those cases.

The first difference is that equation (1) does not assume stationarity of payoffs. $\tilde{F}_t(\cdot)$ in equation (1) is indexed with the t subscript while $\tilde{F}(\cdot)$ in equation (6) is not. As science advances continuously, the payoff of a trial might evolve over time independent of the states and actions. For example, a new technology that becomes available on day t could increase the probabilities of success of all trials in the future. Non-stationary MABs are called "restless bandits" (Whittle, 1988). Theoretical literature has found indices to be suboptimal for restless bandits in the general case (Ortner et al., 2012; Lattimore & Szepesvári, 2020). However, if one is willing to assume the nature of change in $\tilde{F}_t(\cdot)$, such as abrupt changes in unknown periods, Garivier & Moulines (2011) show that UCB-like indices can match the lower bound on regret up to a logarithmic factor. Such indices are based on the idea of "forgetting" and would discount information learned in early periods or use sliding windows to exclude that information.

Another difference is that equation (1) has a much more complex action space. Equation (1) allows for allocating multiple trials to the same project or to different projects in a period if $n_{lt} > 1$ and n_{lt} can change from period to period. All allocated trials reveal their payoffs. In contrast,

equation (6) only allows for allocating one trial to one project in each period. MABs that allow for pulling multiple arms in each period, where all pulled arms reveal their payoffs, are called "combinatorial semi-bandits." Works have shown that indices could work well for stochastic combinatorial semi-bandits (Kveton et al., 2015; Chen et al., 2016; Wang & Chen, 2018), albeit under quite strong assumptions on the action space and/or the functional form of the payoff function. Recent works push towards relaxing those assumptions (Lattimore & Szepesvári, 2020; Chen et al., 2021).

A further difference is that equation (1) does not assume independence between arms. Equation (1) integrates the payoff over the posterior distribution $\tilde{F}_t(p_{ijt}|\Omega_t)$, which is conditioned on the full information set at t. In contrast, equation (6) integrates over $\tilde{F}^{(i)}(p_{ijt}|\Omega_t^{(i)})$, where the information set is specific to each project i and learning about one project from its trial outcomes does not spill over to another project. Many research projects share similar characteristics, so their payoffs could be correlated. For example, 24% of projects in my data are membrane proteins, whose trials in general had low probabilities of success.[22] When the agent has this kind of contextual information to help predict the payoff, the MABs are called "contextual bandits" (Woodroofe, 1979; Langford & Zhang, 2007). Recent works have shown that UCB-like indices can achieve a nearly optimal regret guarantee on the order of $\tilde{O}(\sqrt{T})$ (Guan & Jiang, 2199; Zhou et al., 2020). These indices build models to correlate the contextual characteristics with the observed outcomes and then use the models to predict the UCB of the payoff of each pull.

Extensive practical applications of index heuristics in bandit-like problems also suggest using index approximation without tight theoretical justification might not be as alarming as it sounds. Algorithms based on index heuristics, especially the UCB, nowadays power applications ranging from product recommendations (Scale (2021)) to dynamic pricing (Yang et al. (2020)) to self-driving cars (Zhang (2021)). The real-world bandit problems are often too complex and messy to find an exact theoretical result so the uses of indices are often without tight theoretical justifications. This does not seem to concern practitioners. The practitioners' goal is not to find the optimal decision model, but to find what works reasonably well and better than the alternatives. Given that the current theoretical literature often fails to find optimal solutions for complex MABs and

[22]Only 1.2% of trials on membrane proteins succeeded while 1.8% of trials on non-membrane proteins succeeded in my data.

needs to rely on index heuristics, it is implausible to think that the labs actually took the optimal actions. I therefore argue that index approximation is not just for computational convenience. Index approximation is a plausible empirical model of the labs' decision-making because the labs could have applied this simple model in decision-making.

4.4 Models of Index Approximation

The main model modifies the simple, well-used UCB index from Auer et al. (2002). Under Auer et al. (2002), $B_{ijt}(\cdot)$ is a convex decreasing function of the amount of input previously allocated to the project.[23] Intuitively, UCB approximates the upper confidence bound of an allocation's posterior expected payoff. Allocating input to a project that the lab has poor information about has a high UCB. As the lab allocates more input to the project and has better information about its productivity, the UCB of further allocations decreases. My modification to Auer et al. (2002) is an additional term that captures time-discounting of the value of older projects that the lab has attempted in the distant past. This term proxies for the unobserved changes in the value of exploring a project over time from the lab's perspective, for example due to project-specific learning staying with individual researchers and personnel changes in the lab. The main model is as follows.

Main Model (UCB+D). *The labs set* $B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Bl}) = a_{ijt}[\sqrt{\frac{\theta_{B1,l}}{j}} + \theta_{B2,l} \cdot (t - t'_{i,t})]$.[24]

The free parameter $\theta_{B1,l}$ captures the amount of exploration. The first few trials of a project have larger $B_{ijt}(\cdot)$ compared to later trials of the same project under this model and $\theta_{B1,l}$ determines how much larger. If a lab did not explore at all, the first few trials of its projects should not have larger $B_{ijt}(\cdot)$ relative to later trials. In that case, $\theta_{B1,l} = 0$. The free parameter $\theta_{B2,l}$ captures the amount of discounting of older projects. The variable $(t - t'_{i,t})$ measures the duration between the current period t and the period $t'_{i,t}$ in which the last previous trial on project i happened. A negative

[23] Under Auer et al. (2002),

$$B_{ijt}(\cdot, a_{ijt} = 1) = \begin{cases} \infty & \text{if } j = 1 \\ \sqrt{\frac{2\ln(N_{lt})}{j-1}} & \text{if } j = 2, 3, 4... \end{cases} \tag{7}$$

where N_{lt} is the total number of pulls the agent has done so far. More recent implementation uses a fixed value θ_{B1} rather than $2\ln(N_{lt})$ (Lattimore & Szepesvári, 2020). For my main model, I do not use an infinite value for $B_{ijt}(\cdot)$ when $j = 1$. If I do, V^A_{ijt} would be infinite and cause problems in estimation via maximum likelihood and in identifying θ_{B1}.

[24] If $j = 1$, $t'_{i,t} = t$.

coefficient estimate of $\theta_{B2,l}$ would suggest the lab discounted projects it tried a while ago when it decided which projects to explore further. If the lab did not discount older projects, $\theta_{B2,l} \geq 0$.

I also define several alternative models for robustness checks. Those models are obvious candidates but one can surely add additional models. Theoretical literature and practical applications have proposed a wide range of other indices for MABs and we can in principle test each one of them. Ultimately, the models tested should be reasonably close to the allocation model the labs actually used. My conversations with NIH program officers and a project coordinator at one of the four largest labs reveal that the trial allocation process was quite heuristic. The labs used a "high-throughput" approach to trial allocation, where they allocated one initial trial to a lot of projects. If the projects were important or if the projects showed promise of success in previous trials, then they allocated more trials to those projects. As such, simple indices could well capture the actual allocation model. I therefore only chose to test the most parsimonious indices well-studied in theoretical literature.

The first of those alternative models is the static model. Under this model, the labs set the exploration bonus $B_{ijt}(\cdot)$ to zero and completely ignore exploration. This model does not have any free parameters in $B_{ijt}(\cdot)$ and is equivalent to restricting $\theta_{B1,l}$ and $\theta_{B2,l}$ in the main model to zeros.

Alternative Model 1 (static). *The labs set* $B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}) = 0$.

The second alternative model is the Gittins index, a seminal index in theoretical literature (Gittins & Jones, 1979; Gittins, 1979). The Gittins index prescribes optimal actions for the standard infinite-horizon discounted MAB discussed in Section 4.2. Equation (1) was different, so under this model the labs would be using the Gittins index without its optimality guarantee. Computing the exact Gittins index is difficult. I therefore use Brezzi & Lai (2002)'s approximation to the index. This approximation explicitly includes the posterior variance of payoff $Var(\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}))$ in $B_{ijt}(\cdot)$.[25] As the first alternative model, this model does not have any free parameters, other than $\boldsymbol{\theta}_{Xl}$, in $B_{ijt}(\cdot)$.

[25]
$$Var(\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})) = \int (\pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl}) - \int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t))^2\, d\tilde{F}_t(p_{ijt}|\Omega_t). \quad (8)$$

Alternative Model 2 (Gittins). *The labs set* $B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}) = \psi(\cdot) \cdot \sqrt{Var(\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}))}$.[26]

The third alternative model is a simple UCB without discounting that closely resembles Auer et al. (2002). Compared to alternative models 1 and 2 which have no additional free parameters in $B_{ijt}(\cdot)$, this model has one free parameter $\theta_{B1,l}$ that needs to be estimated. The model is equivalent to restricting $\theta_{B2,l}$ of the main model to zero.

Alternative Model 3 (UCB). *The labs set* $B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \theta_{B1,l}) = a_{ijt}\sqrt{\frac{\theta_{B1,l}}{j}}$.

The fourth alternative model relaxes how posterior variance enters $B_{ijt}(\cdot)$ compared to the Gittins index. It also has one free parameter in $B_{ijt}(\cdot)$.

Alternative Model 4 (FlexGittins). *The labs set*
$B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}) = \theta_{B1,l} \cdot \psi(\cdot) \cdot \sqrt{Var(\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}))}$.

The last alternative model adds the term for time-discounting to the FlexGittins model. It has two free parameters, the same as the main model.

Alternative Model 5 (FlexGittins+D). *The labs set*
$B_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}) = \theta_{B1,l} \cdot \psi(\cdot) \cdot \sqrt{Var(\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl}))} + a_{ijt} \cdot \theta_{B2,l} \cdot (t - t'_{i,t})$.

5 Model Fitting

I developed a two-step method to estimate and validate the model. In the first step, I estimate the free parameters in each model by maximizing the likelihood of the observed allocation decisions. In the second step, I use the estimated parameters to forward simulate the labs' entire history of input allocation and output. I compare the patterns of input allocated and output from the simulated data to those from the actual data to determine model fit.

[26] The function $\psi(\cdot)$ is defined as

$$\psi(s) = \begin{cases} \sqrt{s/2} & \text{if } s \leq 0.2 \\ 0.49 - 0.11 s^{-1/2} & \text{if } 0.2 < s \leq 1 \\ 0.63 - 0.26 s^{-1/2} & \text{if } 1 < s \leq 5 \\ 0.77 - 0.58 s^{-1/2} & \text{if } 5 < s \leq 15 \\ \{2log(s) - log(log(s)) - log(16\pi)\}^{-1/2} & \text{if } s > 15, \end{cases} \tag{9}$$

where $s = \frac{Var(p_{ijt}|\Omega_t)}{-ln(\beta)E(p_{ijt}|\Omega_t)(1-E(p_{ijt}|\Omega_t))}$. I set the discount factor $\beta = 0.95$.

5.1 Estimation

In this first step, I estimate the free parameters $\boldsymbol{\theta}_l$ in each model by maximizing the likelihood of the observed allocation decisions. Our model assumes that the labs chose actions based on the approximate value function; therefore the observed actions $\boldsymbol{a}_{l1}^o, \boldsymbol{a}_{l2}^o, ..., \boldsymbol{a}_{lT}^o$ are the solutions $\boldsymbol{a}_{l1}^{A*}, \boldsymbol{a}_{l2}^{A*}, ..., \boldsymbol{a}_{lT}^{A*}$ to equation (4). We can use the likelihood for the observed allocations $P(\boldsymbol{a}_{lt}^o; \boldsymbol{\theta}_l)$ to estimate $\boldsymbol{\theta}_l$.

To form $P(\boldsymbol{a}_{lt}^o; \boldsymbol{\theta}_l)$, I start by rewriting the allocation problem in equation (4) as follows

$$\begin{aligned} \boldsymbol{a}_{lt}^o = &\underset{\boldsymbol{a}_{lt}}{\operatorname{argmax}} \sum_{j_i \in C_{lt}} V_{ijt}^A(\Omega_t, a_{ijt}; \boldsymbol{\theta}_l) + \varepsilon_{it}, \\ &\text{subject to } \sum_{j_i \in C_{lt}} a_{ijt} = n_{lt}. \end{aligned} \tag{10}$$

I replaced $\boldsymbol{a}_{lt}^{A*}$ with $\boldsymbol{a}_{lt}^o$ in equation (10) to reflect that the observed allocations are the optimal actions based on the approximate value function. I introduced an additive error term ε_{it}. I assume all $\varepsilon \overset{iid}{\sim}$ Type I Extreme Value and capture the unobservables in decision-making. This will help me to write $P(\boldsymbol{a}_{lt}^o; \boldsymbol{\theta}_l)$ as a closed-form function of $\boldsymbol{\theta}_l$. I also made some very intuitive functional form assumptions about the payoff function $\pi_{ijt}(\cdot)$, so that $V_{ijt}^A(\cdot)$ is only a function of a_{ijt}, and not a function of the full $\boldsymbol{a}_{lt}$ vector. These assumptions allow me to replace $V_{ijt}^A(\Omega_t, \boldsymbol{a}_{ijt}; \boldsymbol{\theta}_l)$ with $V_{ijt}^A(\Omega_t, a_{ijt}; \boldsymbol{\theta}_l)$. The functional form assumptions also guarantee that the second constraint of equation (4) holds at the solution, allowing me to drop the second constraint for equation (10). See Appendix C.1 for details of the functional form assumptions and derivations.

The solution $\boldsymbol{a}_{lt}^o$ to equation (10) is equivalent to the following index rule: one first computes index $V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) + \varepsilon_{it}$ for each project-trial in C_{lt} and then allocates to the n_{lt} trials with the highest index values.

For computational tractability, I assume the lab made iid decisions whether to allocate each trial, using a threshold rule. The lab allocated a trial whenever the trial's index value was greater than the threshold on that day. Doing so allows me to express the likelihood $P(\boldsymbol{a}_{lt}^o; \boldsymbol{\theta}_l)$ with a simple closed form. Let $V_{lt}^{n_{lt}}(\boldsymbol{\theta}_l)$ denote the n_{lt}th highest value of $V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l)$ on day t.

Define the threshold value which is equal to $V_{lt}^{n_{lt}}(\boldsymbol{\theta}_l)$ plus an error term ε_{lt}:[27]

$$threshold_{lt} = V_{lt}^{n_{lt}}(\boldsymbol{\theta}_l) + \varepsilon_{lt}. \tag{11}$$

The likelihood of observing $a_{ijt}^o = 1$ is equal to the likelihood of $V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) + \varepsilon_{it}$ being greater than this threshold, so the likelihood $P(a_{ijt}^o = 1; \boldsymbol{\theta}_l)$ is

$$\begin{aligned} P(a_{ijt}^o = 1; \boldsymbol{\theta}_l) &= P(V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) + \varepsilon_{it} > V_{lt}^{n_{lt}}(\boldsymbol{\theta}_l) + \varepsilon_{lt}) \\ &= \frac{exp(V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l))}{exp(V_{ijt}^A(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l)) + exp(V_{lt}^{n_{lt}}(\boldsymbol{\theta}_l))}. \end{aligned} \tag{12}$$

The total likelihood function sums over the log likelihood of the observed action for each project-trial in choice sets $C_{l1}, C_{l2}, ..., C_{lT}$. These likelihoods include $P(a_{ijt}^o = 1; \boldsymbol{\theta}_l)$ for trials actually allocated and $1 - P(a_{ijt}^o = 1; \boldsymbol{\theta}_l)$ for trials in those choice sets but which were *not* allocated:

$$\boldsymbol{\theta}_l^* = \underset{\boldsymbol{\theta}_l}{\text{argmax}} \sum_t^T \left(\underbrace{\sum_{j_i \in C_{lt},\, a_{ijt}^o = 1} log(P(a_{ijt}^o = 1; \boldsymbol{\theta}_l))}_{\text{actual allocations}} + \underbrace{\sum_{j_i \in C_{lt},\, a_{ijt}^o = 0} log(1 - P(a_{ijt}^o = 1; \boldsymbol{\theta}_l))}_{\text{actual nonallocations}} \right). \tag{13}$$

One can then estimate $\boldsymbol{\theta}_l$ by maximizing the above likelihood function. Due to the simplicity of computing index approximations, estimation is feasible even though the choice sets contain millions of possible actions over thousands of days. A further trick to reduce computational burden is to compute in each iteration the log likelihoods for a random sample of the possible allocations in the choice sets, rather than for the full choice sets. The number of possible allocations in a full choice set could be large because n_{lt} could be large. Recall that the mean of n_{lt} for the four largest labs is 35. When $n_{lt} = 35$, the $(j+1)$th, $(j+2)$th,..., $(j+35)$th trials of every project in the lab's portfolio are in the full choice set. See Appendix C.2 for how I sample from the choice sets.

The estimated $\hat{\boldsymbol{\theta}}_l$ includes two kinds of parameters. $\hat{\boldsymbol{\theta}}_{Bl}$ are the free parameters in the exploration bonus term $B_{ijt}(\cdot)$, if there are any. $\hat{\boldsymbol{\theta}}_{Xl}$ are welfare weights that enter into the payoff function $\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})$. I let $\boldsymbol{\theta}_{Xl}$ have eight parameters. Seven of these parameters correspond to the NIH

[27] Note that this value is not necessarily the n_{lt}th highest index value on day t due to the iid error terms. In principle, one can obtain the n_{lt}th highest index value on day t with simulation draws of ε's and use that as the threshold. Doing so would be computationally more challenging than my approach.

evaluation metrics discussed in Section 3.1. For example, $\theta_{biomedl,l}$ corresponds to $biomed_i$ and captures the amount of welfare lab l receives when it publishes a biomedically important structure. One additional parameter $\theta_{quant,l}$ captures the baseline amount of welfare lab l receives per unique structure. See Appendix C.1 for the specification of the payoff function $\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})$.

Identification of model parameters is based on revealed preferences and is very intuitive. Let us consider $\theta_{biomed,l}$. When we have a large positive $\theta_{biomed,l}$, it would increase the posterior expected payoff of biomedically important projects and increase V^A_{ijt} of those projects relative to $V^{n_{lt}}_{lt}$, which is based on the distribution of V^A_{ijt} of all project-trials in the lab's choice sets. In contrast, a large positive $\theta_{biomed,l}$ would decrease V^A_{ijt} for non-biomedically important projects relative to $V^{n_{lt}}_{lt}$. A large positive $\theta_{biomed,l}$ therefore increases the likelihood of allocating to biomedically important projects and the likelihood of not allocating to non-biomedically important projects. Correspondingly, if the lab actually allocated a high proportion of trials to biomedically important projects, such a data pattern will lead to a large positive estimate of $\hat{\theta}_{biomed,l}$. Variation in the binary variable $biomed_i$ in the choice sets therefore identifies $\theta_{biomed,l}$. Identification of other parameters follows the same intuition.

5.2 Simulation

In this second step, I use the estimated parameters of each model to forward simulate the labs' entire history of input allocation and output. Doing so would allow me to compare the patterns of input allocated and the output from the simulated data to that from the actual data to determine model fit.

I initialize each simulation with the following conditions.

- **Data generating process (DGP)**

  We need to have a model of the "true" probability of success of a trial to simulate the outcomes of counterfactually allocated trials. I denote this model by F^*. This model is different from the posterior $\tilde{F}_t$ in Section 3.3. $\tilde{F}_t$ captures how the labs formed posterior beliefs. It does not have to produce an unbiased estimate of the true probability of success of a trial, while F^* has to. As such, F^* deviates from $\tilde{F}_t$ in several ways to correct the potential bias of and improve upon the machine learning models the labs described in their published journal

articles. See Appendix B.1 for these implementation details. F^* is trained on Ω_{T+1}, the full information set consisting of all trial outcomes and characteristics in the entire data.

- **Project space**

  F^* may do a poor job imputing the probabilities of success of trials on projects the labs never attempted and are far out in the project space, so I restrict the project space to the projects the labs actually attempted. Applying this restriction still leaves the labs ample room to explore and/or exploit projects as only a tiny fraction of projects ever attempted successfully produced publications.

- **Allocation model**

  This sets the allocation model and its corresponding parameters $\boldsymbol{\theta}'_{Xl}$ and $\boldsymbol{\theta}'_{Bl}$ during the simulation. To validate a model, $\boldsymbol{\theta}'_{Xl}$ and $\boldsymbol{\theta}'_{Bl}$ are equal to the estimated parameters $\hat{\boldsymbol{\theta}}_{Xl}$ and $\hat{\boldsymbol{\theta}}_{Bl}$ under the model.

- **Prior allocations and output**

  Allocations and outcomes before 2005 are used as prior data and not simulated.

I then do the following to forward simulate the labs' entire history of input allocation and output since 2005.

Algorithm 1: Forward simulation of trial allocations and outcomes

For *day t in 2005–2015,*

- **Update posterior beliefs** Counterfactual information set Ω'_t consists of the actual and simulated actions and outcomes observed before t. Refit model of the lab's posterior belief $\tilde{F}'_t(\Omega'_t)$.
- **Make allocations** Form choice set C'_{lt}. Compute $\widehat{V^A}'_{ijt}$ of each choice in the choice set based on $\tilde{F}'_t(p'_{ijt}|\Omega'_t)$, $\boldsymbol{\theta}'_{Xl}$ and $\boldsymbol{\theta}'_{Bl}$. Draw iid ε'_{it} from a Type-I Extreme Value distribution. Allocate to n_{lt} trials with the largest $\widehat{V^A}'_{ijt} + \varepsilon'_{it}$ values.
- **Simulate outcomes** Generate the true trial success probability $p^*_{ijt}|F^*(\Omega_{T+1})$ for the allocated trials. Draw outcomes $Y'_{ijt} \sim Bernoulli(\hat{p}^*_{ijt})$ for these trials.
- **Update information set** Add to the information set actions and outcomes observed in this period. Go to the next period.

End

6 Model Fitting Results

I find the main model fits the data extremely well and captures the labs' decision-making. During the maximum likelihood step, the main model by far has the smallest magnitude of log likelihood at convergence among the many alternative models I tested. It fits the data better with no or few additional parameters as compared to the alternatives. During the simulation step, the main model generates input allocation patterns and output that are very similar to those in the data. For all labs, the deviations of the simulated output from the actual output are within 10%. The alternative models fail to generate patterns matching the data as closely.

Based on the estimated parameters of this well-fitting model, I am able to reject the hypothesis that the labs did no explore at 95% confidence for all labs.

6.1 Estimation Results

Table 2 shows that the main model fits the data much better with no or few additional parameters than the alternatives through a comparison of log likelihoods. With the same number of parameters, the main model UCB+D has a magnitude of log likelihood that is only half the size of that of the FlexGittins+D model. In fact, the fit of the simple UCB model without discounting is already remarkable. With fewer parameters, the simple UCB model has a magnitude of log likelihood that is similar to that of the FlexGittins+D model. This table shows results for one large lab. Appendix Tables D1–D3 show results for the other large labs which are qualitatively similar. Among those labs, the main model has a magnitude of log likelihood that is 28% to 41% smaller than the FlexGittins+D model.

Table 2 also shows that the main model fits better through a comparison of the predicted likelihoods of the observed allocation decisions. It shows that, among the project-trials the lab actually allocated to, the main model on average predicts a 83.7% likelihood of allocating to each project-trial. Among the project-trials the lab did *not* actually allocate to, the main model on average predicts a 99.9% likelihood of *not* allocating to each project-trial. These predicted likelihoods are by far the highest among the models fitted. Appendix Tables D1–D3 show results for the other large labs which are qualitatively similar.

As a further robustness check, I examine the out-of-sample fit of the main model. To do so, I first fit the main model using only observed allocation decisions in odd years. I then use the estimated parameters to compute the average predicted likelihoods and log likelihoods in-sample and out-of-sample. I compute the in-sample results using decisions in odd years which I used to fit the model. I compute the out-of-sample results using decisions in even years. Appendix Table D4 shows a comparison of the in-sample and out-of-sample results and suggests that the out-of-sample fit of the main model is quite similar to the in-sample fit for all labs.

Table 3 shows estimates of main parameters of interest in the main model. As these parameters reflect the lab's preferences, the estimates are not in dollar values and are therefore hard to interpret on their own. Still, one can make interpretations by comparing the estimates with zero or with each other in different periods. The table shows results for one large lab. Appendix Table D6 shows results for the other large labs which are qualitatively similar.

Estimates in Table 3 reject the hypothesis that the lab did not explore. Recall that θ_{B1} captures the amount of exploration under the main model. If the lab did not explore, $\theta_{B1} = 0$. Estimates of θ_{B1} in Table 3 are large and significantly different from zero at 95% confidence level, allowing us to reject the hypothesis that the lab did not explore. For all other large labs, we can similarly reject no exploration at 95% confidence.

Estimates in Table 3 also show that the lab discounted older projects in making allocations. Discounting of older projects proxies for many unobserved factors in the allocation process. For example, if there were personnel changes in the lab over time and there were project-specific learning that stayed with individual researchers that frequently left, the lab would be less inclined to attempt projects that it had attempted in the distant past. Recall that a negative θ_{B2} captures discounting under the main model. If the lab did not discount older projects, $\theta_{B2} \geq 0$. Estimates of θ_{B2} are negative and significantly different from zero at 95%, allowing us to reject no discounting. For all other large labs, we can similarly reject no discounting at 95% confidence.

As another robustness check, I compare the estimates for θ_{biomed} during 2005–2008 and during 2009–2015. Recall that the NIH boosted its preference for biomedically important projects in 2009. As expected, the estimate during 2009–2015 is significantly larger than the estimate during 2005–2008 as their 95% confidence intervals do not overlap. This relationship holds for estimates from all other large labs.

We have shown the main model fits better than many alternatives. However, it is difficult to jump to the conclusion that the model correctly captures the actual allocation decisions. To investigate whether this model could generate patterns similar to those in the actual data, I use simulations.

6.2 Simulation Results

Table 4 shows simulated outcomes using estimated model parameters for one large lab. Results for other labs are in Appendix Table D7 and are qualitatively similar.

The first column of Table 4 shows that the simulated number of projects attempted under the main model matches the actual number. The other models, in contrast, fall short by various degrees. Consistent with our intuition, the lab does not explore under the static model so the number of

projects attempted is small at 14,175. The models based on the Gittins index encourage exploration by adding a positive exploration bonus so the numbers of projects attempted are slightly larger. The two UCB models explore extensively and the numbers of projects attempted under the two models are much closer to the actual. Figure 8 shows a further breakdown of the simulated input allocation across projects, plotting the distribution of the number of trials per attempted project under each model. One immediately sees that the simulated distribution of the main model fits the actual distribution the best.

The second column of Table 4 shows that the simulated number of unique publications under the main model matches the actual number. Figure 9 shows a further breakdown, plotting the simulated number of publications by year under each model.[28] The first panel of Figure 9 shows that output quantity under the static model peaks in 2006, the year immediately after the start of the simulation, then gradually declines. This is consistent with static maximization that the lab immediately allocates input to projects that it believes to be the most likely to pay off. In later years the lab exhausts such projects as it does not explore many new projects. The models based on the Gittins index produce slightly more output than the static model, but the output patterns are qualitatively similar. The two UCB models in contrast maintain steady output over the years because extensive exploration of new projects under the two models replenishes the lab's portfolio. The two models generate output patterns that are very similar to the actual output patterns.

The third and fourth columns of Table 4 show that the simulated numbers of citations and downloads under the main model match the actual numbers. These results suggest that the kinds of projects the lab allocates input to under the model are similar to the kinds of projects the lab actually allocated input to. To investigate further, Figure 10 compares the characteristics of trials actually allocated with those of trials allocated in simulations. The first panel displays the proportions of trials on projects that the lab cited as biomedically important. The main model produces allocation patterns very similar to the actual allocation patterns. Starting in 2009, the lab drastically shifted towards attempting more biomedically important projects, consistent with the change in NIH preferences in that year. The model captures this drastic shift almost perfectly, indicating that the lab changed the mix of project characteristics in its portfolio mostly through introducing

[28]The actual series has a hump shape. Before 2009, the lab slowly built up capacity. In 2009, the lab became aware that the grant program would end in 2015 and started to slowly dismantle capacity.

new projects with the desired characteristics. In contrast, the increase is small for the static model around 2009 as the lab would rarely draw new projects under this model and only would change the mix of project characteristics by reallocating input within its existing portfolio. The second panel and the third panel show two more project characteristics that the NIH evaluated the lab's progress on. The fourth panel shows one project characteristic that the NIH did not evaluate on. Very similar to the first panel, these panels show that the main model produces allocation patterns very similar to the actual allocation patterns.

Results from other labs in Appendix Table D7 are very similar to those in Table 4. For all four large labs, the deviations of the simulated outcomes under the main model from the actual outcomes are within 10%.

As a further robustness check, I examine the out-of-sample simulation results of the main model. To do so, I first fit the main model using only observed allocation decisions in odd years. I then use the estimated parameters to simulate the lab's entire input allocation history and output, in both odd and even years. As I do not use the observed decisions in even years to fit the model for this exercise, I deem these simulated results out-of-sample. Appendix Table D8 shows a comparison of the in-sample and out-of-sample simulation results and suggests that the out-of-sample results are quite similar to the in-sample results for all labs.

I therefore claim with reasonable confidence that the main model captures the labs' actual allocation decision-making. Although we can never prove a model to be true because the scientific method only allows us to falsify untrue ones, we cannot reject the main model based on simulation results. Simulation results have shown that the main model could produce allocation patterns and outcomes that align well with the actual allocation patterns and outcomes.

7 Counterfactual—The Impact of Exploration on Productivity

With a well-fitting model of how the labs made the exploitation-exploration tradeoff, I study the effect of exploration on the labs' productivity through counterfactual simulations. Doing so involves simulating a counterfactual where the labs do not explore and comparing the outcomes with the baseline outcomes from the main model.

The simulation procedure for this counterfactual is largely identical to that for model valida-

tion in Section 5.2, with only differences in setting $\boldsymbol{\theta}'_{Xl}$ and $\boldsymbol{\theta}'_{Bl}$. To simulate this counterfactual, I set the welfare weights $\boldsymbol{\theta}'_{Xl}$ to $\hat{\boldsymbol{\theta}}_{Xl,UCB+D}$, the ones recovered from estimating the main model. Intuitively, doing so maintains the labs' estimated preferences for project characteristics and studies how changing only the labs' model of exploration impacts the output. I set $\boldsymbol{\theta}'_{Bl}$ to zeros to mimic no exploration. Intuitively, setting $\boldsymbol{\theta}'_{Bl}$ to zeros makes the exploration bonus term $B_{ijt}(\cdot)$ zero for all trials in the labs' choice sets. The labs thus make allocations as if they follow the static model of decision-making—allocating only trials that they believe would have the highest posterior expected payoff.

Table 5 displays the counterfactual results for one large lab. Appendix Table D9 shows results for other labs. Those results are qualitatively similar.

Based on results in the above tables, I find exploration was extensive and had a large impact on the labs' productivity. For the lab displayed in Table 5, no exploration would result in the lab attempting 79% fewer projects, producing 64% fewer unique publications and 65% fewer citations. Summing up results from all four large labs, no exploration would result in the labs attempting 82% fewer projects, producing 51% fewer unique publications and 57% fewer citations.

Applying back-of-the-envelope calculations with some simple assumptions, no exploration would result in a loss of at least $650 to $720 million of economic value. Suppose that the other smaller labs under the $1.3 billion NIH program would have the same percentage decrease in output quantity had they not explored, and that the economic value of each unit of output is uniform; in that case the 51% decrease in output quantity would be equivalent to forgoing at least $650 million of economic value, if the NIH had a nonnegative economic return to the program. Suppose instead that the other smaller labs would have the same percentage decrease in citations had they not explored, and that the economic value of each citation is uniform; in that case the 57% decrease in citations would be equivalent to forgoing at least $720 million of economic value.

Figure 11 plots the simulated number of unique publications per year for one large lab under no exploration, compared to that under the original UCB+D model. The figure offers further evidence of how exploration boosted the labs' productivity. As the figure shows, output under no exploration is as high as that under the UCB+D model in 2005, the initial year simulated. In the following years, however, output under no exploration declines steadily as compared to that under the UCB+D model. Exploration enables the lab to find low-hanging fruits, which are in parts of

the project space that the lab is not aware of at the beginning of 2005. In contrast, purely exploiting projects that the lab believes to be the most productive misses those low-hanging fruit. Over the years, no exploration results in substantial misallocation of resources to less productive projects.

8 Conclusion

Extensive theoretical literature has recognized the importance of making the exploitation-exploration tradeoff in resource allocation for innovation, yet little research has studied empirically how innovators make this tradeoff. This paper overcomes the empirical challenges with novel data and a new estimation method, and it studies how a group of large scientific labs made the exploitation-exploration tradeoff.

I find that a simple model captures the labs' decision-making extremely well. This model embodies extensive exploration and strongly resembles the simple UCB index of Auer et al. (2002). The only modification is that I added an additional variable to capture time-discounting of older projects. During the maximum likelihood step, I find this model fits the data the best with no or few additional parameters compared to the many alternative models I tested. During the simulation step, I find this model generates input allocation patterns and output that are very similar to those in the actual data. For all labs, the deviations of the simulated output from the actual output are within 10%.

Based on this well-fitting model, I find that the labs explored extensively. For all labs, I am able to reject no exploration at 95% confidence based on estimates of the model's free parameters. Counterfactual simulations show that exploration was extensive and that it had a large impact on the labs' productivity. Had the labs not explored, their output quantity would have decreased by 51%, and their citations would have decreased by 57%. The decrease is equivalent to a loss of least \$650 to \$720 million of economic value.

These results are valuable for several reasons. First, they confirm intuition from the theoretical literature about the value of exploration for resource allocation under incomplete information. Moreover, they suggest sophistication in real-world decision-making. The labs were forward-looking and took into account the dynamic incentives to explore and acquire information, rather than being myopic and only performing static maximization. These results corroborate conven-

tional wisdom related to resource allocation for innovation. For example, funding agencies often designate small early career grants to many young researchers who have scant track records. As researchers mature, track records play a more significant role in follow-on funding. Moreover, these answers have policy implications for the multi-billion-dollar innovation industry. If these results extend to the whole industry by any degree, exploration has a nontrivial contribution to innovation productivity through the channel of improving the resource allocation process. This research offers a new lens to look at the value of exploration of ideas.

This paper also opens up many possibilities for future research, one of them being further studies of how resource allocation policies could improve innovation productivity. Many open questions remain. For example, which policy features of funding programs make it easier for innovators to choose good allocation models? Besides the allocation model, does the design of the allocation process, such as the duration and scale, affect productivity? The answers to these questions can inform future policies. For example, when a funding agency designs early career grants, the level and duration of such grants are important considerations.

Another possibility is to apply the method developed in this paper to many other economic problems that embody the exploitation-exploration tradeoff. In principle, whenever economic agents need to make that tradeoff in dynamic resource allocation, we can use the index approximation approach to model their decisions. The approach overcomes the computational challenges in the workhorse estimation methods due to the curse of dimensionality. It allows one to recover an empirical model of decision-making in a problem with thousands of periods and millions of choices. The potential areas of application are numerous, e.g., the study of purchase decisions of experience goods, targeted marketing and product recommendations, investment decisions in new markets, and so on.

References

Agrawal, Rajeev (1995) "Sample Mean Based Index Policies with O(log n) Regret for the Multi-Armed Bandit Problem," *Advances in Applied Probability*, 27 (4), 1054–1078.

Aguirregabiria, Victor and Pedro Mira (2010) "Dynamic Discrete Choice Structural Models: A Survey," *Journal of Econometrics*, 156 (1), 38–67.

Arrow, Kenneth (1962) "Economic Welfare and the Allocation of Resources for Invention," in *The Rate and Direction of Inventive Activity: Economic and Social Factors*, 609–625, Princeton: Princeton University Press.

Auer, Peter, Nicolo Cesa-Bianchi, and Paul Fischer (2002) "Finite-Time Analysis of the Multi-armed Bandit Problem," *Machine Learning*, 47 (2), 235–256.

Azoulay, Pierre (2012) "Turn the Scientific Method on Ourselves: How Can We Know Whether Funding Models for Research Work? By Relentlessly Testing Them Using Randomized Controlled Trials," *Nature*, 484 (7392), 31–33.

Azoulay, Pierre, Joshua S Graff Zivin, Danielle Li, and Bhaven N Sampat (2019) "Public R&D Investments and Private-Sector Patenting: Evidence from NIH Funding Rules," *Review of Economic Studies*, 86 (1), 117–152.

Azoulay, Pierre, Joshua S Graff Zivin, and Gustavo Manso (2011) "Incentives and Creativity: Evidence from the Academic Life Sciences," *RAND Journal of Economics*, 42 (3), 527–554.

Babnigg, György and Andrzej Joachimiak (2010) "Predicting Protein Crystallization Propensity from Protein Sequence," *Journal of Structural and Functional Genomics*, 11 (1), 71–80.

Bergemann, Dirk and Juuso Välimäki (1996) "Learning and Strategic Pricing," *Econometrica*, 64 (5), 1125–1149.

Berman, Helen M, Margaret J Gabanyi, Andrei Kouranov, David I Micallef, John Westbrook, and Protein Structure Initiative network of investigators (2017) "Protein Structure Initiative—TargetTrack 2000-2017—all data files," https://zenodo.org/record/821654, accessed June 25, 2019.

Bolton, Patrick and Christopher Harris (1999) "Strategic Experimentation," *Econometrica*, 67 (2), 349–374.

Boroush, Mark A (2021) "New Data on U.S. R&D: Summary Statistics from the 2019–20 Edition of National Patterns of R&D Resources," *National Center for Science and Engineering Statistics (NCSES)*, https://ncses.nsf.gov/pubs/nsf22314.

Brezzi, Monica and Tze Leung Lai (2002) "Optimal Learning and Experimentation in Bandit Problems," *Journal of Economic Dynamics and Control*, 27 (1), 87–108.

Bubeck, Sébastien and Nicolo Cesa-Bianchi (2012) "Regret Analysis of Stochastic and Nonstochastic Multi-Armed Bandit Problems," *arXiv preprint arXiv:1204.5721*.

Buchfink, Benjamin, Klaus Reuter, and Hajk-Georg Drost (2021) "Sensitive Protein Alignments at Tree-of-Life Scale Using DIAMOND," *Nature Methods*, 18 (4), 366–368.

Buchfink, Benjamin, Chao Xie, and Daniel H Huson (2015) "Fast and Sensitive Protein Alignment Using DIAMOND," *Nature Methods*, 12 (1), 59–60.

Cantoni, Davide and Noam Yuchtman (2014) "Medieval Universities, Legal Institutions, and the Commercial Revolution," *Quarterly Journal of Economics*, 129 (2), 823–887.

Cappé, Olivier, Aurélien Garivier, Odalric-Ambrym Maillard, Rémi Munos, Gilles Stoltz et al. (2013) "Kullback–Leibler Upper Confidence Bounds for Optimal Sequential Allocation," *Annals of Statistics*, 41 (3), 1516–1541.

Caria, Stefano, Grant Gordon, Maximilian Kasy, Simon Quinn, Soha Shami, and Alex Teytelboym (2020) "An Adaptive Targeted Field Experiment: Job Search Assistance for Refugees in Jordan," CESifo Working Paper.

Chen, Wei, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu (2016) "Combinatorial Multi-Armed Bandit with General Reward Functions," *Advances in Neural Information Processing Systems*, 29.

Chen, Wei, Liwei Wang, Haoyu Zhao, and Kai Zheng (2021) "Combinatorial Semi-Bandit in the Non-Stationary Environment," in *Uncertainty in Artificial Intelligence*, 865–875, Proceedings of Machine Learning Research.

Chen, Xiaohong, Timothy M Christensen, and Elie Tamer (2018) "Monte Carlo Confidence Sets for Identified Sets," *Econometrica*, 86 (6), 1965–2018.

Chernozhukov, Victor and Han Hong (2003) "An MCMC Approach to Classical Estimation," *Journal of Econometrics*, 115 (2), 293–346.

Chruszcz, Maksymilian, Alexander Wlodawer, and Wladek Minor (2008) "Determination of Protein Structures–A Series of Fortunate Events," *Biophysical Journal*, 95 (1), 1–9.

Cohen, Wesley M and Daniel A Levinthal (1989) "Innovation and Learning: the Two Faces of R&D," *Economic Journal*, 99 (397), 569–596.

Crawford, Gregory S and Matthew Shum (2005) "Uncertainty and Learning in Pharmaceutical Demand," *Econometrica*, 73 (4), 1137–1173.

DeepMind (2020) "AlphaFold: Using AI for Scientific Discovery," https://deepmind.com/blog/article/AlphaFold-Using-AI-for-scientific-discovery.

Dickstein, Michael J (2018) "Efficient Provision of Experience Goods: Evidence from Antidepressant Choice," Working Paper.

Ederer, Florian and Gustavo Manso (2013) "Is Pay for Performance Detrimental to Innovation?" *Management Science*, 59 (7), 1496–1513.

EMBL-EBI (2021) "EMBL-EBI Pfam to PDB Mapping," http://ftp.ebi.ac.uk/pub/databases/Pfam/mappings/.

Erdem, Tülin and Michael P Keane (1996) "Decision-Making under Uncertainty: Capturing Dynamic Brand Choice Processes in Turbulent Consumer Goods Markets," *Marketing Science*, 15 (1), 1–20.

Furman, Jeffrey L and Scott Stern (2011) "Climbing Atop the Shoulders of Giants: The Impact of Institutions on Cumulative Research," *American Economic Review*, 101 (5), 1933–63.

Ganglmair, Bernhard, Timothy Simcoe, and Emanuele Tarantino (2019) "Learning When to Quit: An Empirical Model of Experimentation in Standards Development," Working Paper.

Garivier, Aurélien and Eric Moulines (2011) "On Upper-Confidence Bound Policies for Switching Bandit Problems," in *International Conference on Algorithmic Learning Theory*, 174–188.

Gittins, John C (1979) "Bandit Processes and Dynamic Allocation Indices," *Journal of the Royal Statistical Society: Series B (Methodological)*, 41 (2), 148–164.

Gittins, John C and David M Jones (1979) "A Dynamic Allocation Index for the Discounted Multiarmed Bandit Problem," *Biometrika*, 66 (3), 561–565.

Guan, Melody and Heinrich Jiang, "Nonparametric Stochastic Contextual Bandits," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 32 (1).

Guan, R., E. Marcos, P. O'Connell et al. (2014) "Crystal Structure of an engineered protein with denovo beta sheet design, Northeast Structural Genomics Consortium (NESG) Target OR486," https://www.rcsb.org/structure/4R80.

Henderson, Rebecca and Iain Cockburn (1996) "Scale, Scope, and Spillovers: The Determinants of Research Productivity in Drug Discovery," *RAND Journal of Economics*, 27 (1), 32–59.

Hill, Ryan and Carolyn Stein (2020) "Scooped! Estimating Rewards for Priority in Science," Working Paper.

——— (2021) "Race to the Bottom: Competition and Quality in Science," Working Paper.

Hoelzemann, Johannes, Gustavo Manso, Abhishek Nagaraj, and Matteo Tranchero (2022) "The Streetlight Effect in Data-Driven Innovation," Working Paper.

Hotz, V Joseph and Robert A Miller (1993) "Conditional Choice Probabilities and the Estimation of Dynamic Models," *The Review of Economic Studies*, 60 (3), 497–529.

Huang, Hongzhan, Peter B McGarvey, Baris E Suzek, Raja Mazumder, Jian Zhang, Yongxing Chen, and Cathy H Wu (2011) "A Comprehensive Protein-Centric ID Mapping Service for Molecular Data Integration," *Bioinformatics*, 27 (8), 1190–1191.

Jaffe, Adam B (2002) "Building Programme Evaluation into the Design of Public Research-Support Programmes," *Oxford Review of Economic Policy*, 18 (1), 22–34.

Jahandideh, Samad, Lukasz Jaroszewski, and Adam Godzik (2014) "Improving the Chances of Successful Protein Structure Determination with a Random Forest Classifier," *Acta Crystallographica Section D: Biological Crystallography*, 70 (3), 627–635.

Jaroszewski, Lukasz, Lukasz Slabinski, John Wooley, Ashley M Deacon, Scott A Lesley, Ian A Wilson, and Adam Godzik (2008) "Genome Pool Strategy for Structural Coverage of Protein Families," *Structure*, 16 (11), 1659–1667.

Katehakis, Michael N and Herbert Robbins (1995) "Sequential Choice from Several Populations.," *Proceedings of the National Academy of Sciences*, 92 (19), 8584–8585.

Kawashima, Shuichi, Piotr Pokarowski, Maria Pokarowska, Andrzej Kolinski, Toshiaki Katayama, and Minoru Kanehisa (2007) "AAindex: Amino Acid Index Database, Progress Report 2008," *Nucleic Acids Research*, 36 (suppl_1), D202–D205.

Keller, Godfrey, Sven Rady, and Martin Cripps (2005) "Strategic Experimentation with Exponential Bandits," *Econometrica*, 73 (1), 39–68.

Khmelnitskaya, Ekaterina (2022) "Competition and Attrition in Drug Development," Working Paper.

Klausen, Michael Schantz, Martin Closter Jespersen, Henrik Nielsen et al. (2019) "NetSurfP-2.0: Improved Prediction of Protein Structural Features by Integrated Deep Learning," *Proteins: Structure, Function, and Bioinformatics*, 87 (6), 520–527.

Krieger, Joshua L (2021) "Trials and Terminations: Learning from Competitors' R&D Failures," *Management Science*, 67 (9), 5525–5548.

Kveton, Branislav, Zheng Wen, Azin Ashkan, and Csaba Szepesvari (2015) "Tight Regret Bounds for Stochastic Combinatorial Semi-Bandits," in *Artificial Intelligence and Statistics*, 535–543, Proceedings of Machine Learning Research.

Lai, Tze Leung (1987) "Adaptive Treatment Allocation and the Multi-Armed Bandit Problem," *Annals of Statistics*, 15 (1), 1091–1114.

Lai, Tze Leung and Herbert Robbins (1985) "Asymptotically Efficient Adaptive Allocation Rules," *Advances in Applied Mathematics*, 6 (1), 4–22.

Lane, J.N., C. Ayoubi, K. Boudreau, E. Guinan, and Lakhani K. (2022) "Generating Innovation in the Lab: Experimental Evidence from the Life Sciences," Working Paper.

Lane, Julia I, Jason Owen-Smith, Rebecca F Rosen, and Bruce A Weinberg (2015) "New Linked Data on Research Investments: Scientific Workforce, Productivity, and Public Value," *Research Policy*, 44 (9), 1659–1671.

Langford, John and Tong Zhang (2007) "The Epoch-Greedy Algorithm for Contextual Multi-Armed Bandits," *Advances in Neural Information Processing Systems*, 20 (1), 96–1.

Lattimore, Tor and Csaba Szepesvári (2020) *Bandit Algorithms*, Cambridge: Cambridge University Press.

Li, Danielle, Lindsey R Raymond, and Peter Bergman (2020) "Hiring as Exploration," National Bureau of Economic Research Working Paper.

Miller, Robert A (1984) "Job Matching and Occupational Choice," *Journal of Political Economy*, 92 (6), 1086–1120.

Miller, Susan, Joel Janin, Arthur M Lesk, and Cyrus Chothia (1987) "Interior and Surface of Monomeric Proteins," *Journal of Molecular Biology*, 196 (3), 641–656.

Moore, Peter B (2007) "Let's Call the Whole Thing Off: Some Thoughts on the Protein Structure Initiative," *Structure*, 15 (11), 1350–1352.

Myers, Kyle (2020) "The Elasticity of Science," *American Economic Journal: Applied Economics*, 12 (4), 103–34.

NIGMS (2007a) "Frequently Asked Questions," http://www.nigms.nih.gov/Initiatives/PSI/Background/FAQs.htm.

——— (2007b) "Frequently Asked Questions for PSI-2," http://www.nigms.nih.gov/Initiatives/PSI/Background/FAQsPSI-2.htm.

——— (2007c) "Request for Information (RFI): To Solicit Input for the Assessment of the Protein Structure Initiative," http://grants.nih.gov/grants/guide/notice-files/NOT-GM-07-108.html.

——— (2008a) "Protein Structure Initiative (Pilot Phase) Fact Sheet," http://www.nigms.nih.gov/Initiatives/PSI/Background/PilotFacts.htm.

——— (2008b) "Report of the Future Structural Genomics Initiatives Meeting," http://www.nigms.nih.gov/News/Reports/FutureSGMeeting102008.htm.

——— (2009a) "Concept Clearance: High-Throughput Structural Biology," https://www.nigms.nih.gov/News/Reports/council_concept_clearance_2009.

——— (2009b) "Protein Structure Initiative Advisory Committee (PSIAC): Recommendations for the Future of the PSI," http://www.nigms.nih.gov/News/Reports/PSIAC_Future_2009.htm.

——— (2011a) "PSI Production Phase Fact Sheet," http://www.nigms.nih.gov/Research/FeaturedPrograms/PSI/Background/PSI2FactSheet.htm.

——— (2011b) "PSI:Biology FOA Frequently Asked Questions," https://www.nigms.nih.gov/research/specificareas/PSI/PSI_biology/background/pages/PSIbiology_faqs.aspx.

NIH (2019) "NIGMS Funding Opportunities Search," https://www.nigms.nih.gov/grants/Pages/Funding.aspx?expired=1.

——— (2021) "NIH RePORT Advanced Projects Search," https://reporter.nih.gov/.

Ortner, Ronald, Daniil Ryabko, Peter Auer, and Rémi Munos (2012) "Regret Bounds for Restless Markov Bandits," in *International Conference on Algorithmic Learning Theory*, 214–228.

Pakes, Ariel (1986) "Patents as Options: Some Estimates of the Value of Holding European Patent Stocks," *Econometrica*, 755–784.

Petsko, Gregory A (2007) "An Idea Whose Time Has Gone," *Genome Biology*, 8 (6), 1–3.

Price, W Nicholson, Yang Chen, Samuel K Handelman et al. (2009a) "Supplementary Information for Understanding the Physical Properties that Control Protein Crystallization by Analysis of Large-Scale Experimental Data," *Nature Biotechnology*, 27 (1), 51–57.

——— (2009b) "Understanding the Physical Properties that Control Protein Crystallization by Analysis of Large-Scale Experimental Data," *Nature Biotechnology*, 27 (1), 51–57.

Roberts, Kevin and Martin L Weitzman (1981) "Funding Criteria for Research, Development, and Exploration Projects," *Econometrica*, 49 (5), 1261–1288.

Russo, Daniel, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen (2017) "A Tutorial on Thompson Sampling," *arXiv preprint arXiv:1707.02038*.

Rust, John (1987) "Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher," *Econometrica: Journal of the Econometric Society*, 999–1033.

——— (1994) "Structural Estimation of Markov Decision Processes," *Handbook of Econometrics*, 4, 3081–3143.

Scale (2021) "Netflix Explains Recommendations and Personalization," https://scale.com/blog/Netflix-Recommendation-Personalization-TransformX-Scale-AI-Insights.

Slabinski, Lukasz, Lukasz Jaroszewski, Ana PC Rodrigues, Leszek Rychlewski, Ian A Wilson, Scott A Lesley, and Adam Godzik (2007a) "The Challenge of Protein Structure Determination–Lessons from Structural Genomics," *Protein Science*, 16 (11), 2472–2482.

Slabinski, Lukasz, Lukasz Jaroszewski, Leszek Rychlewski, Ian A Wilson, Scott A Lesley, and Adam Godzik (2007b) "XtalPred: A Web Server for Prediction of Protein Crystallizability," *Bioinformatics*, 23 (24), 3403–3405.

Timmins, Christopher (2002) "Measuring the Dynamic Efficiency Costs of Regulators' Preferences: Municipal Water Utilities in the Arid West," *Econometrica*, 70 (2), 603–629.

UniProt (2021a) "Programmatic Access—Mapping Database Identifiers," https://www.uniprot.org/help/api_idmapping.

——— (2021b) "Programmatic Access - Retrieving Individual Entries," https://www.uniprot.org/help/api_retrieve_entries.

——— (2021c) "Taxonomy—Eukaryota," https://www.uniprot.org/taxonomy/2759.

——— (2021d) "Taxonomy—Homo sapiens (Human)," https://www.uniprot.org/taxonomy/9606.

UniProt Consortium, The (2021) "UniProt: The Universal Protein Knowledgebase in 2021," *Nucleic Acids Research*, 49 (D1), D480–D489.

Van Montfort, Rob LM and Paul Workman (2017) "Structure-Based Drug Design: Aiming for a Perfect Fit," *Essays in Biochemistry*, 61 (5), 431–437.

Varadi, Mihaly, John Berrisford, Mandar Deshpande et al. (2020) "PDBe-KB: a Community-Driven Resource for Structural and Functional Annotations," *Nucleic Acids Research*, 48 (D1), D344–D353.

Wang, Siwei and Wei Chen (2018) "Thompson Sampling for Combinatorial Semi-Bandits," in *International Conference on Machine Learning*, 5114–5122, Proceedings of Machine Learning Research.

Weitzman, Martin L (1979) "Optimal Search for the Best Alternative," *Econometrica*, 641–654.

Whittle, Peter (1988) "Restless Bandits: Activity Allocation in a Changing World," *Journal of Applied Probability*, 25 (A), 287–298.

Woodroofe, Michael (1979) "A One-Armed Bandit Problem with a Concomitant Variable," *Journal of the American Statistical Association*, 74 (368), 799–806.

Wootton, John C (1994) "Non-Globular Domains in Protein Sequences: Automated Segmentation Using Complexity Measures," *Computers & Chemistry*, 18 (3), 269–285.

Wrapp, Daniel, Nianshuang Wang, Kizzmekia S Corbett, Jory A Goldsmith, Ching-Lin Hsieh, Olubukola Abiona, Barney S Graham, and Jason S McLellan (2020) "Cryo-EM Structure of the 2019-nCoV Spike in the Prefusion Conformation," *Science*, 367 (6483), 1260–1263.

wwPDB (2013) "PDB Structure Download Statistics," http://www.wwpdb.org/downloadStats.php.

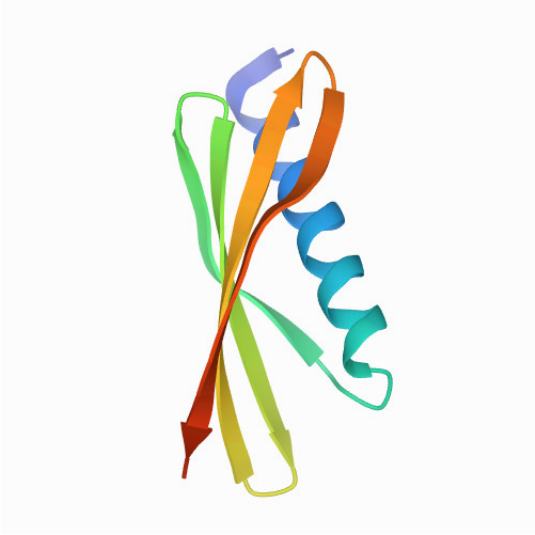
Yang, Jeremy, Dean Eckles, Paramveer Dhillon, and Sinan Aral (2020) "Targeting for Long-Term Outcomes," *arXiv preprint arXiv:2010.15835*.

Zhang, Jason (2021) "Deep Understanding Tesla FSD Part 3: Planning & Control," https://saneryee-studio.medium.com/deep-understanding-tesla-fsd-part-3-planning-control-9a25cc6d04f0.

Zhou, Dongruo, Lihong Li, and Quanquan Gu (2020) "Neural Contextual Bandits with UCB-Based Exploration," in *International Conference on Machine Learning*, 11492–11502, Proceedings of Machine Learning Research.

Figures and Tables

Figure 1: Three-dimensional structure of a protein molecule



Note: The Northeast Structural Genomics Consortium (NESG), one of the large labs in my data, published this structure in 2014 (Guan et al., 2014).

Figure 2: Lab makes decisions about allocating resources across projects in each period

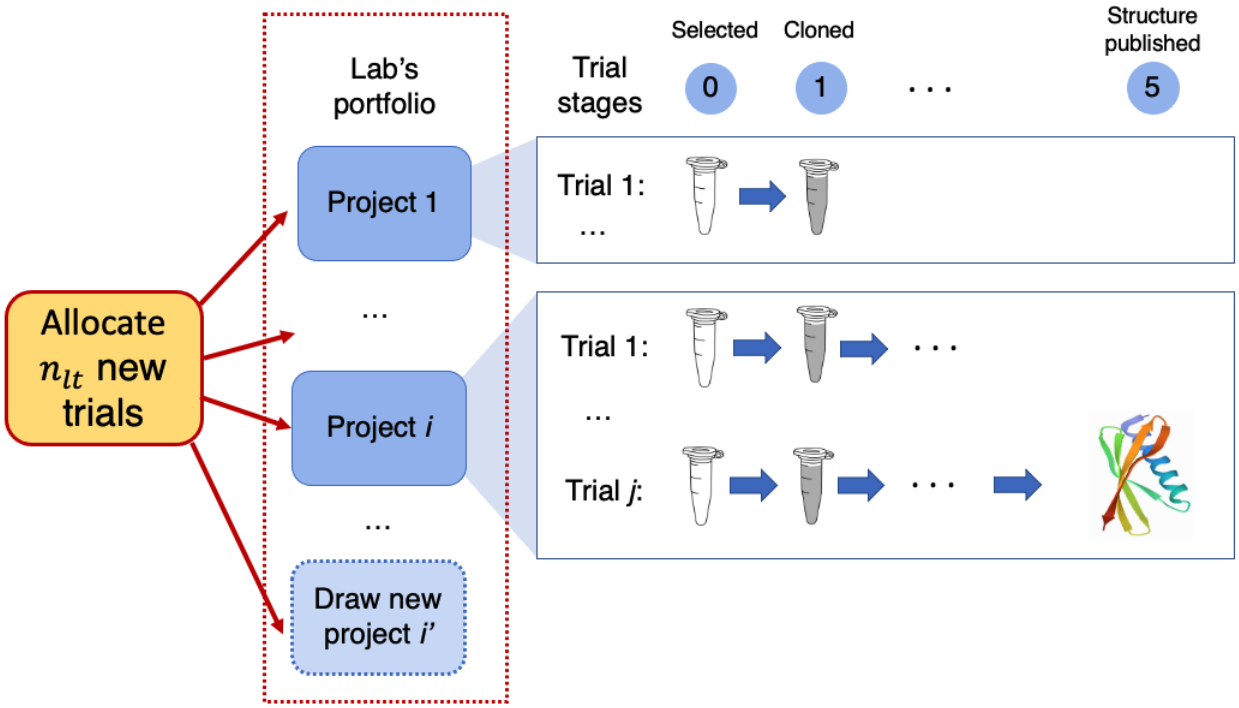


Image credits: The images of PCR tubes are from labicons.net.

Figure 3: Collection and analysis of information guide allocation decisions

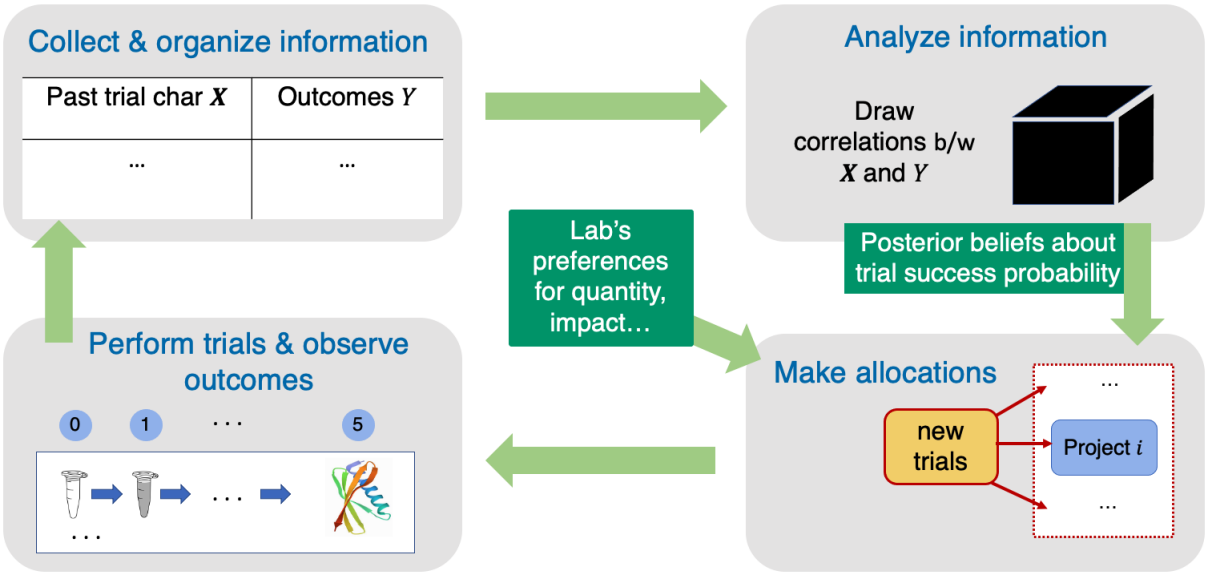
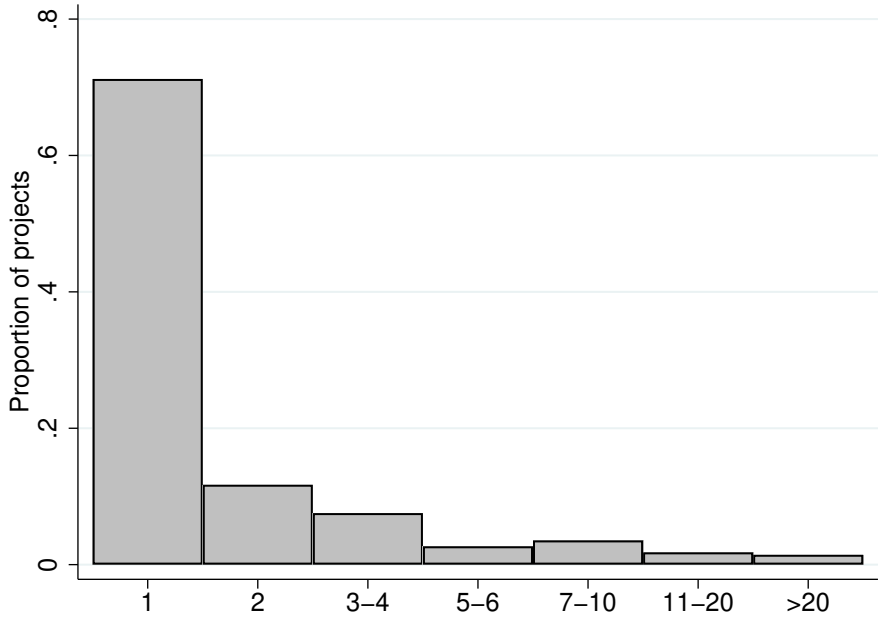
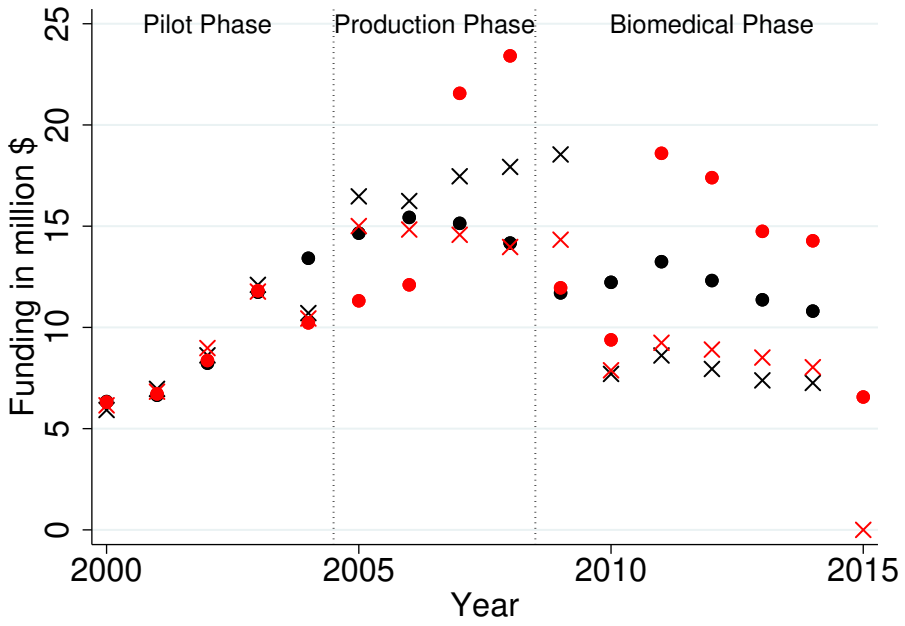


Figure 4: Distribution of input allocation across projects



Note: This figure includes all 335,553 projects ever attempted in my data.

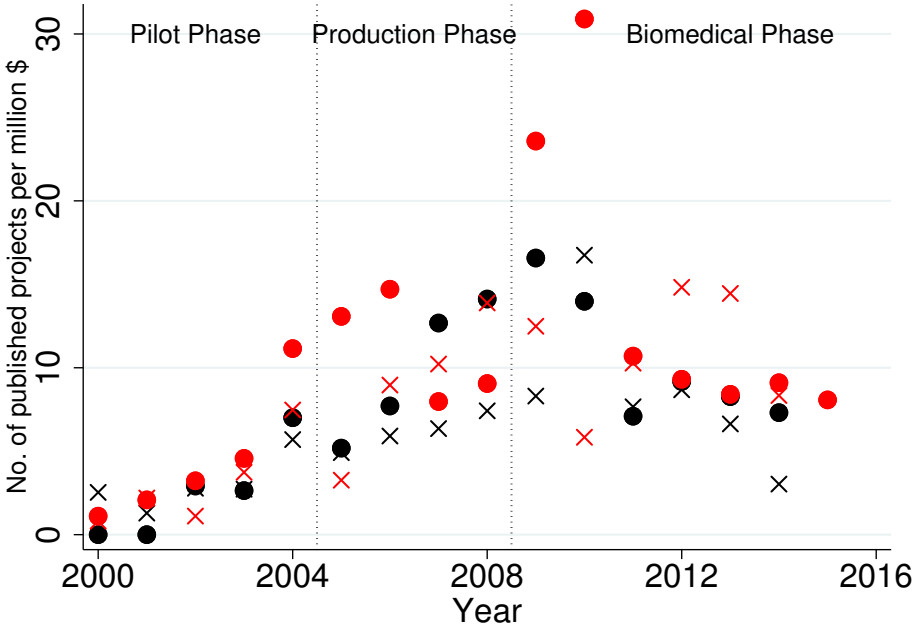
Figure 5: Funding of the four largest labs



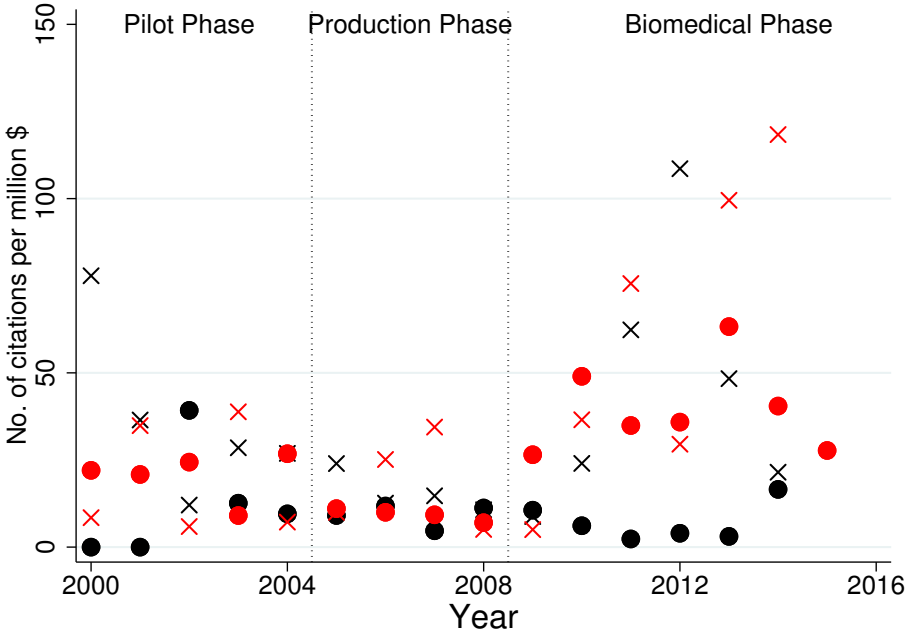
Note: Each series represents one lab.

Figure 6: Observed output of the four largest labs

(a) Number of publications

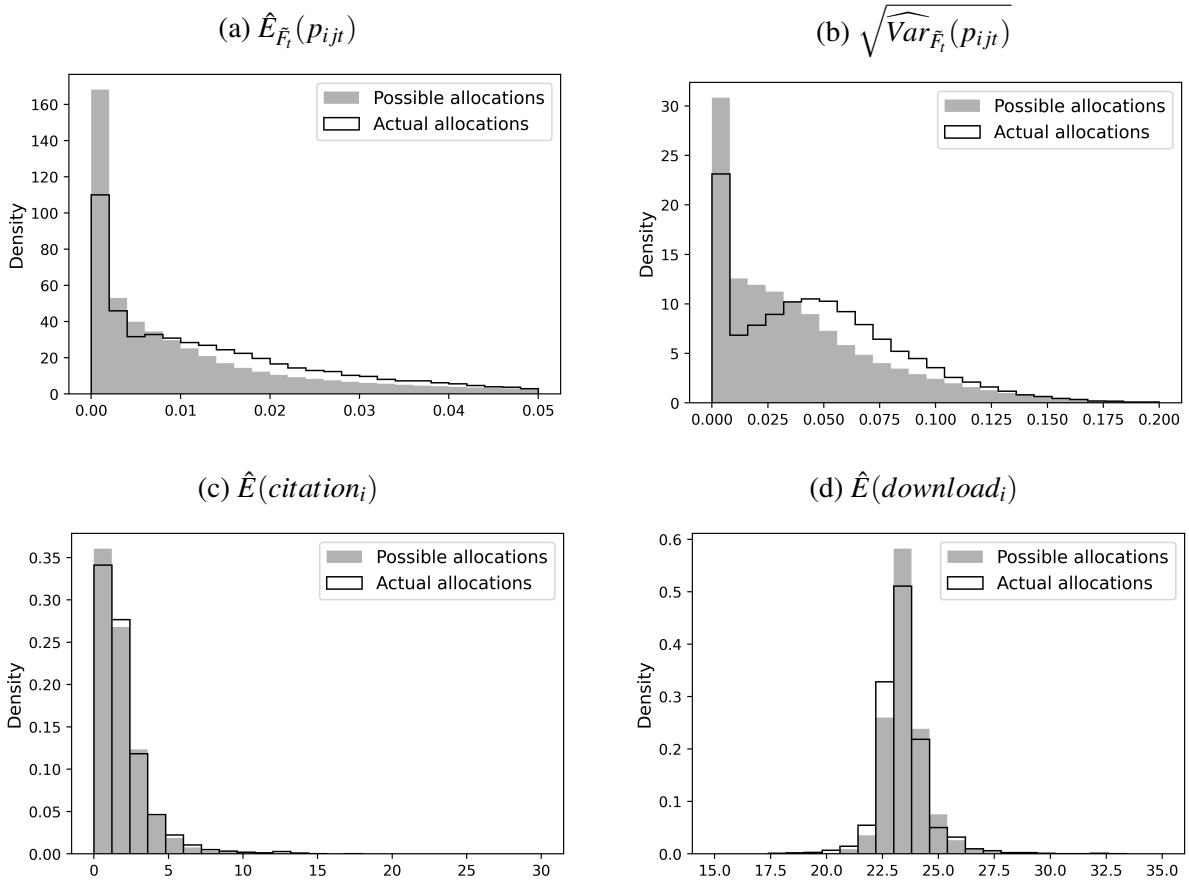


(b) Citations and mentions



Note: Each series represents one lab. Panel 6a shows the number of published structures on unique molecules in a given year divided by the lab's funding in millions in that year. Panel 6b shows the number of five-year citations and mentions the published structures in that year generated, divided by the lab's funding in millions in that year.

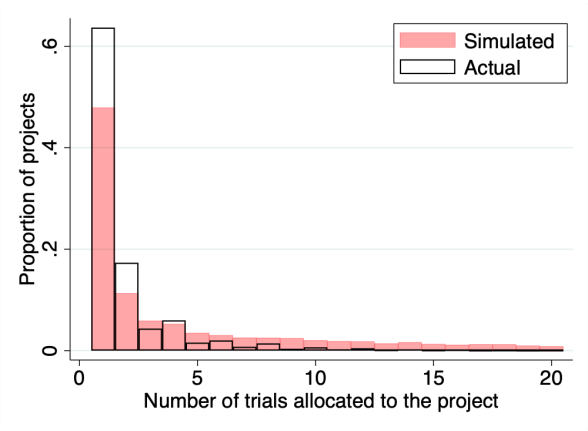
Figure 7: Estimated posterior beliefs about output



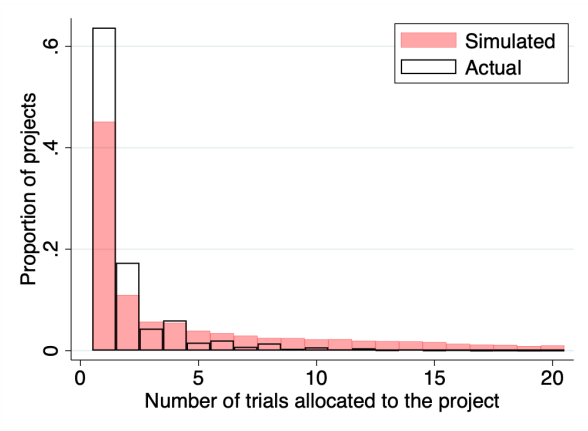
Note: Each panel shows two distributions. The outlined bars show the distribution of the variable for trials actually allocated at the four largest labs. The size of this data is 714,736. The grey unoutlined bars show the distribution of the variable for a random sample of all possible trials in those labs' choice sets. The size of this data is 105,533,539. See Appendix C.2 about sampling from choice sets. Panel 7a shows the distributions of the posterior expected probability of success of a trial. Panel 7b shows the posterior standard deviation of the probability of success of a trial. Panel 7c shows the expected five-year citations and mentions of a trial upon publication. Panel 7d shows the expected five-year downloads (in thousands) of a trial upon publication. An observation from a lab is inversely weighted by the total number of observations from this lab so each lab contributes equally to the aggregate distribution. The number of bins is coarse and equal to 25 to avoid depicting very tall bars near zero. The distributions are truncated to their respective ranges of x-axis values.

Figure 8: Simulated distributions of input allocation across projects

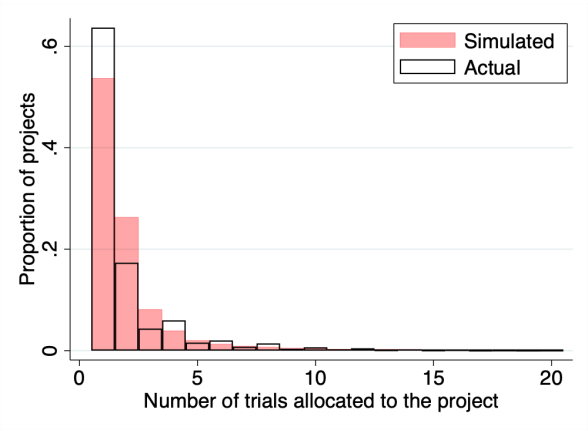
(a) Static



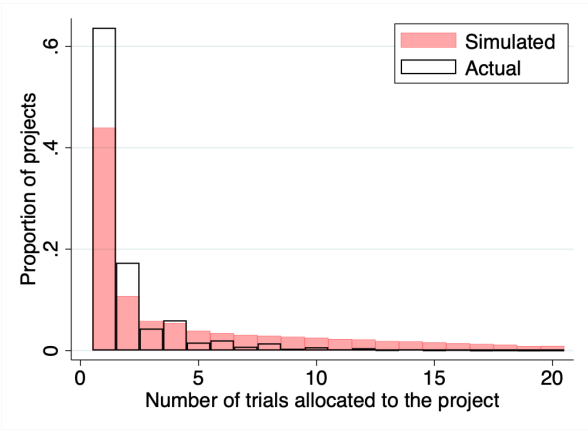
(b) Gittins



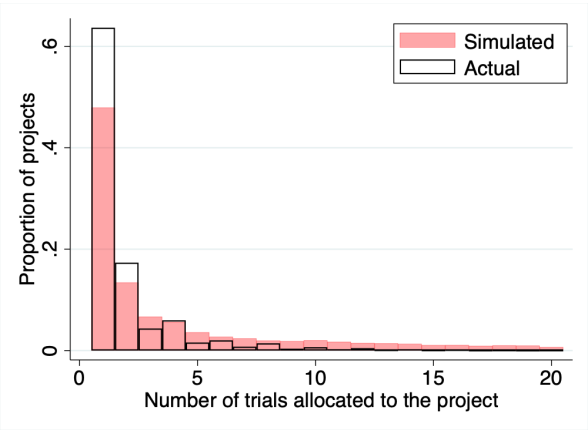
(c) UCB



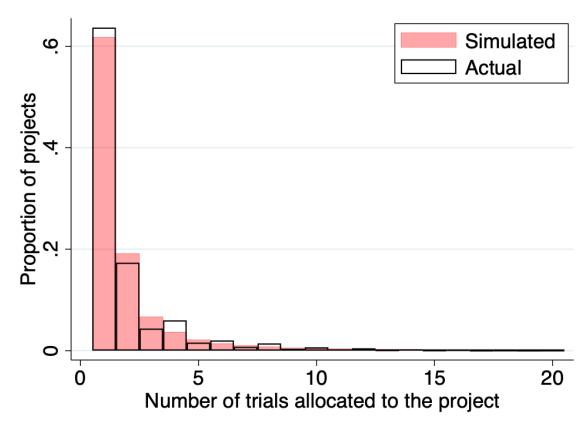
(d) FlexGittins



(e) FlexGittins+D

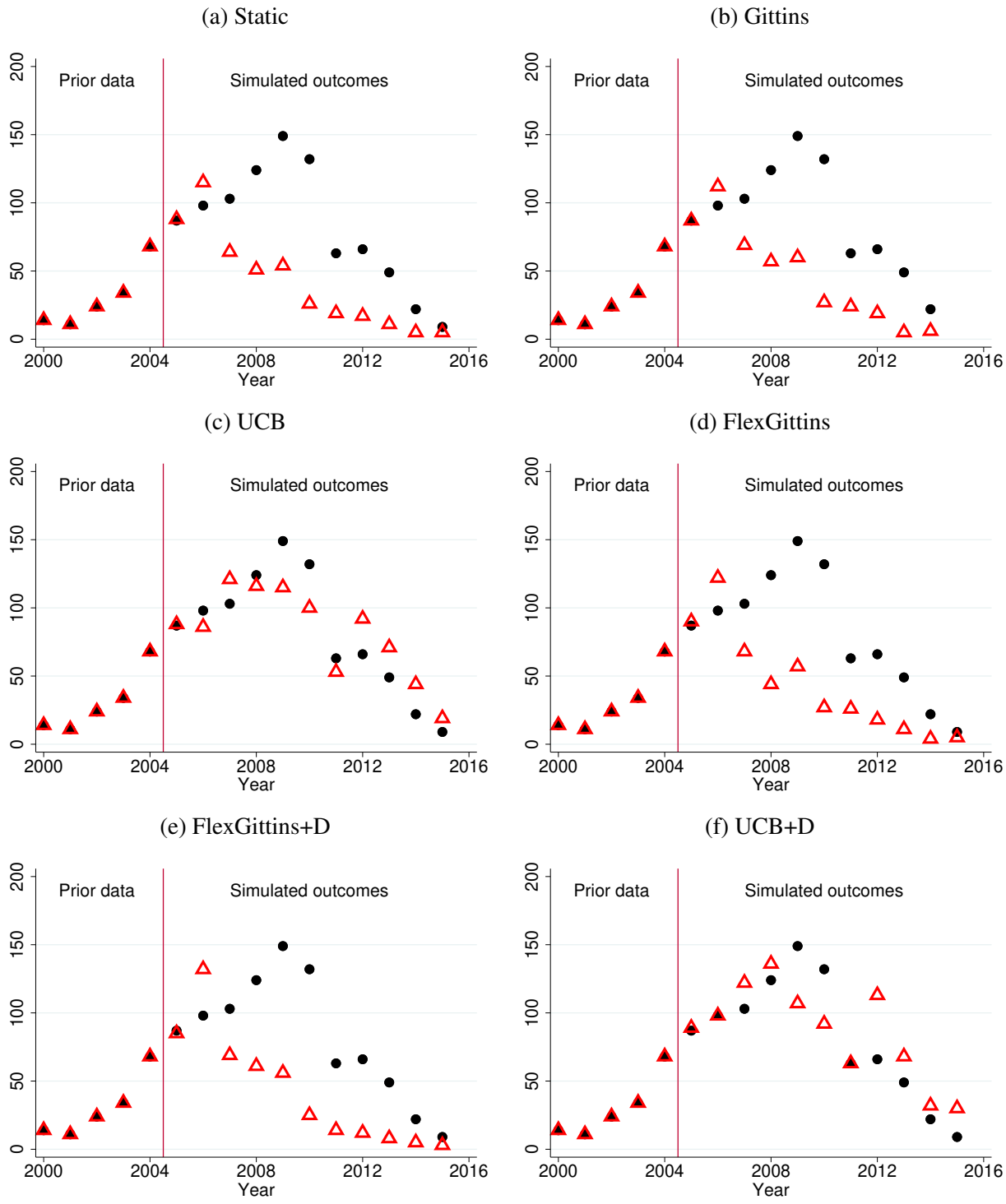


(f) UCB+D



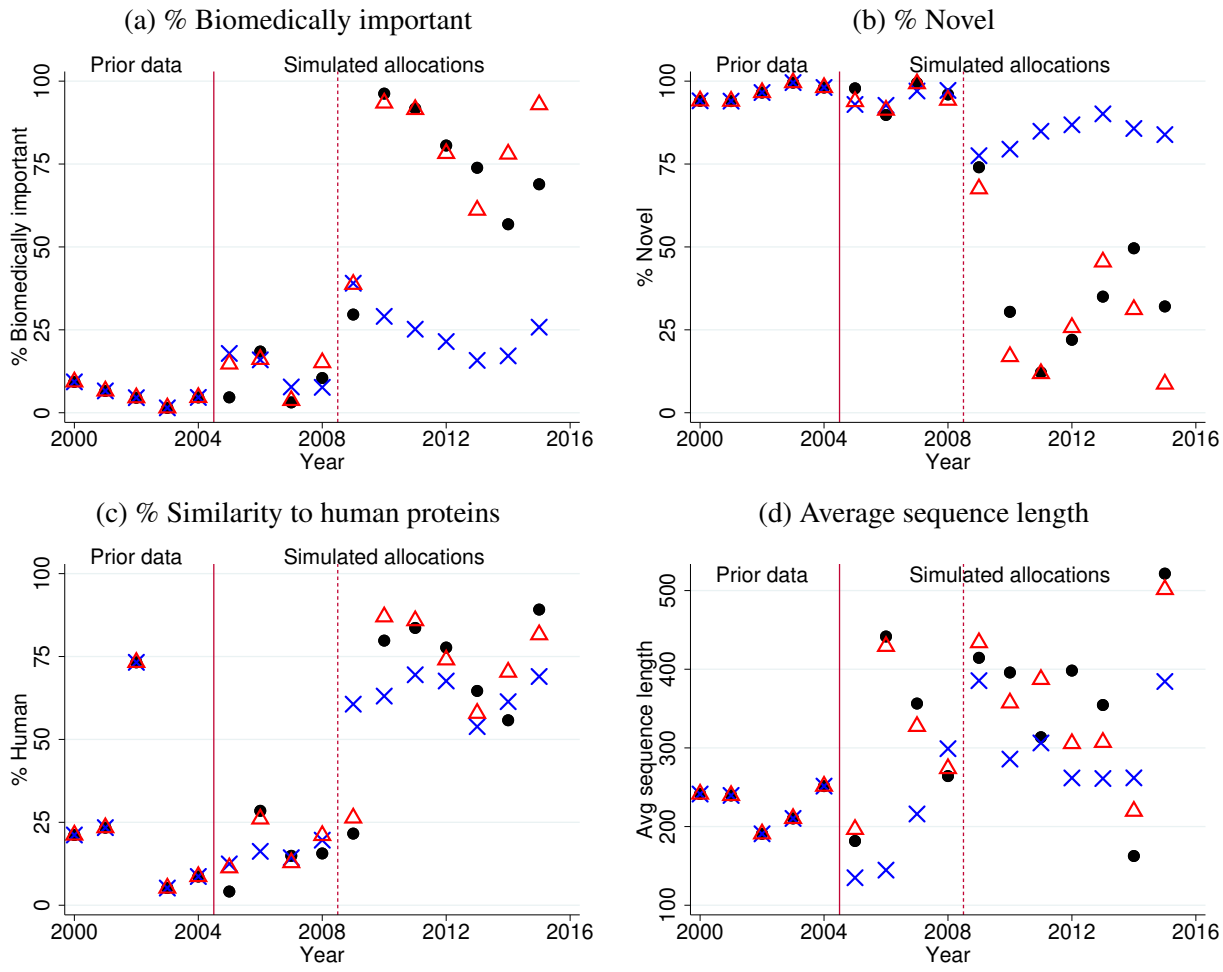
Note: Each panel shows two distributions. The outlined bars show the actual distribution of the number of trials across projects at NESG. The pink unoutlined bars show the simulated distribution under the model. Each distribution is based on a single simulation of the corresponding model.

Figure 9: Simulated number of unique publications under different models



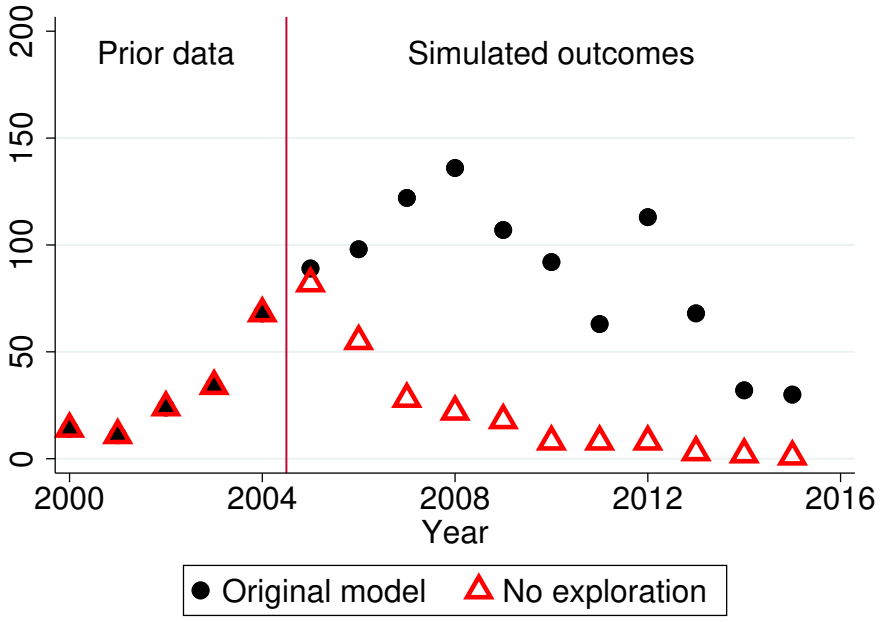
Note: Black dots represent actual numbers of publications on unique molecules at NESG. Red triangles represent simulated numbers of publications on unique molecules. Each series is based on a single simulation of the corresponding model.

Figure 10: Characteristics of trials allocated in simulations, static model versus UCB+D model



Note: Black dots represent characteristics of actually allocated trials at NESG. Blue crosses represent those of trials allocated under the static model. Red triangles represent those of trials allocated under the UCB+D model. Each series is based on a single simulation of the corresponding model.

Figure 11: Simulated number of unique publications, no exploration



Note: Each series is based on a single simulation of the corresponding model for NESG. Black dots represent simulated output under the original UCB+D model. For this simulation, one sets welfare weights $\boldsymbol{\theta}'_{Xl}$ to the estimated welfare weights $\hat{\boldsymbol{\theta}}_{Xl,UCB+D}$ under the UCB+D model, and $\boldsymbol{\theta}'_{Bl}$ to $\hat{\boldsymbol{\theta}}_{Bl,UCB+D}$. This series is identical to the series represented by red triangles in Figure 9f. Red triangles represent simulated output under no exploration. For this simulation, one also sets welfare weights $\boldsymbol{\theta}'_{Xl}$ to the estimated welfare weights $\hat{\boldsymbol{\theta}}_{Xl,UCB+D}$ under the UCB+D model, but sets $\boldsymbol{\theta}'_{Bl}$ to zeros.

Table 1: Key variables and data sources

Variable	Variable Description	Source
Input allocation		
i	Unique identifier of a project or molecule	Berman et al. (2017)
j_i	Unique identifier of the jth trial of project i	Berman et al. (2017)
n_{lt}	Number of trials to allocate at lab l on day t	Berman et al. (2017)
$novel_i$	Binary $= 1$ if project i was novel	Berman et al. (2017)
$prevStruct_{iy}$	Number of published structures in the same protein family as i by year y	EMBL-EBI (2021)
$biomed_i$	Binary $= 1$ if project i was biomedically important	Berman et al. (2017)
$prevPub_{iy}$	Number of publications (not limited to structures) on molecule i by year y	UniProt Consortium (2021)
$human_i$	Max percentage identity of i to any human molecule	UniProt Consortium (2021)
$eukaryote_i$	Max percentage identity of i to any eukaryotic molecule	UniProt Consortium (2021)
$membrane_i$	Binary $= 1$ if molecule i is a membrane protein	UniProt Consortium (2021)
$funding_{ly}$	Funding in dollars lab l received in year y	NIH (2019), NIH (2021)
Observed output		
Y_{ijt}	Binary $= 1$ if trial j_i on day t succeeded	Berman et al. (2017)
Y_{ijkt}	Binary $= 1$ if stage k of trial j_i on day t succeeded	Berman et al. (2017)
$citation_{iy}$	five-year citations and mentions of project i published in year y	Varadi et al. (2020)
$download_{im}$	Number of downloads of project i in month m (between Aug 2007 and Nov 2013)	wwPDB (2013)
Posterior beliefs about output		
$\hat{E}_{\tilde{F}_t}(p_{ijt})$	Best-effort replication of the labs' posterior expectation of the probability of success of trial j_i on day t	Appendix B.1
$\widehat{Var}_{\tilde{F}_t}(p_{ijt})$	Posterior variance of the probability of success of trial j_i on day t	Appendix B.1
$\hat{E}(citation_{iy})$	Expectation of five-year citations and mentions of project i published in year y	Appendix B.2
$\hat{E}(download_{iy})$	Expectation of five-year downloads of project i published in year y	Appendix B.3

Note: This table only summarizes key variables and data sources. For a full description of all data sources and variables, please see Appendix A.

Table 2: Comparison of likelihoods of different models

Model	Free parameters in $B_{ijt}(\cdot)$	Log likelihood	Avg $\hat{P}(a^o_{ijt}=1;\boldsymbol{\theta}_l)$ actual allocation	Avg $\hat{P}(a^o_{ijt}=0;\boldsymbol{\theta}_l)$ actual nonallocation
Static	0	-613,630	0.578	0.993
Gittins	0	-430,746	0.608	0.996
UCB	1	-204,228	0.763	0.998
FlexGittins	1	-412,112	0.614	0.996
FlexGittins+D	2	-230,048	0.710	0.998
UCB+D	2	-119,861	0.837	0.999

Note: I separately estimate each model during 2005–2008 and during 2009–2015 due to the change in the NIH's preferences in 2009. Estimates are in Appendix Table D5. The total log likelihood of each model adds up the log likelihoods from the two periods. Average $\hat{P}(a^o_{ijt}=1;\boldsymbol{\theta}_l)$ for actual allocation sums over the predicted likelihoods of all actually allocated trials in the two periods and divides the sum by the number of actually allocated trials. Average $\hat{P}(a^o_{ijt}=0;\boldsymbol{\theta}_l)$ for actual non-allocation sums over the predicted likelihoods of all actually not allocated trials in the choice sets and divides the sum by the number of actually not allocated trials. Estimation uses data between 2005 and 2015 from NESG, one of the four large labs. The number of trials actually allocated was 109,738. I reduced the sizes of the choice sets with random sampling (see Appendix C.2). The number of trials in the choice sets after random sampling is 32,410,947. Estimation results from the three other large labs are qualitatively similar (see Appendix Tables D1–D3).

Table 3: Estimates of main parameters of interest in UCB+D model

	2005–2008	2009–2015
Parameter	(1)	(2)
θ_{B1}	158.28 [156.35, 160.13]	119.52 [118.15, 121.41]
θ_{B2}	-2.28 [-2.23, -2.30]	-4.71 [-4.66, -4.73]
θ_{biomed}	21.50 [21.32, 21.72]	52.86 [52.69, 52.93]

Note: Table displays the estimates of the main parameters of interest in the UCB+D model for NESG, one of the four large labs. See Appendix Table D5 for the full estimates in different models for the lab. See Appendix Table D6 for the estimates of the main parameters of interest in the UCB+D model for other labs. The column with the header "2005–2008" uses data between 2005 and 2008. The number of trials actually allocated was 59,261. After random sampling (see Appendix C.2), the number of trials in the choice sets is 5,628,158. The columns with the header "2009–2015" uses data between 2009 and 2015. The number of trials actually allocated was 50,477. After random sampling, the number of trials in the choice sets is 26,782,789. 95% confidence intervals are computed using the MCMC approach in Chernozhukov & Hong (2003) and are shown in brackets. These confidence intervals are almost identical to those computed using Procedure 1 of Chen et al. (2018).

Table 4: Comparison of simulated outcomes of different models

Model	Projects attempted	Unique publications	Citations	Downloads (millions)
Static	14,175	597	1,787	14.1
Gittins	14,980	631	1,898	14.9
UCB	59,164	1,052	3,236	24.5
FlexGittins	15,005	621	1,830	14.6
FlexGittins+D	17,382	638	1,892	15.0
UCB+D	59,947	1,097	3,376	25.6
Actual	59,953	1,053	3,502	24.5[†]

Note: Each simulation uses $\hat{\boldsymbol{\theta}}_l$ from parameter estimates of the corresponding model. See Table D5 for those estimates. Results are averaged from three simulations of each model. Actual outcomes are shown in the last row. Simulation results are for NESG, one of the four large labs. Simulation results from the three other large labs are qualitatively similar (see Appendix Tables D7). [†]The download data on actually published projects are between 2007 and 2013, so actual five-year downloads may not be available for some projects. I predict five-year downloads for the actually published projects using the model described in Appendix B.3.

Table 5: Counterfactual outcomes, no exploration

Counterfactual model	Projects attempted	Unique publications	Citations	Downloads (millions)
Static	12,421	394	1,179	9.2
	(−79%)	(−64%)	(−65%)	(−64%)
Baseline model	59,947	1,097	3,376	25.6

Note: Each simulation uses $\hat{\boldsymbol{\theta}}_{Xl}$ from parameter estimates of the UCB+D model in columns (1) and (2) of Table D5. Results are averaged from three simulations of the model. Output from the baseline model are identical to those in the second last row of Table 4. Parentheses show percentage differences as compared to the baseline model. Table shows results from NESG, one of the four large labs. See Appendix Table D9 for results from other labs.

Exploit or Explore?
An Empirical Study of Resource Allocation in Scientific Labs

Appendices

A Data and Variable Construction

A.1 Project Rationale

I construct two binary variables, $novel_i$ and $biomedical_i$, based on textual descriptions in the TargetTrack information system. This section describes the variable construction process.

First, I use keywords to identify projects that were novel and/or biomedically important. TargetTrack contains a variable called *targetCategoryList* where labs give projects categorical labels such as "biomedical," "structural coverage,"[29] and so on. It also contains a text field called *targetRationale* where labs give textual descriptions of projects' rationales. Whenever *targetCategoryList* and *targetRationale* contain the following keywords, I set $novel_i$ equal to 1:

big,[30] coverage of protein universe, diversity, first structure of class, low sequence identity, mega,[31] metagenomic, new fold, no structural information, no structure, numer of homologs,[32] pfam, remote homologs, structural coverage, structural template for unsolved, structure coverage, unsolved families, without any solved structures, without structure.

Whenever *targetCategoryList* and *targetRationale* contain the following keywords, I set $biomedical_i$ equal to 1:

activator, adhension, antibiotic, binding, biochemistry, biological interest, biomedical, cascade, catalyze, cell development, community nominated, communit-nominated,[33] community-nominated, community request, conserved, disease, coronavirus, drug, drug development, drug target, effector, enzyme, essential, function, functional studies, functional, gpcr, high value, higvalue,[34] hiv, homeostasis, host, immune, immunity, infection, infectious, inhibitor, interaction, interact, legionella, medical school, metabolism, mitochondria, model system, operon, parkinsons, partnership, pathogen, pathology, pathway, phosphatase, pneumonia, protein family of high biological importance, reagent, receptor, resistance, resistant, salmonella, school of medicine, secret, sensor, shen lab, shen_lab, shen_selection, stem cell, substrate, syndrome, synthesis, t-cell, t cell, therapeutic, thorson lab, toxoplasma, transcription, transport, tuberculosis, tumor, university, vaccine, vibrio, virulence, virulent.

Second, I use labs' selection protocols of projects for additional information. TargetTrack

[29] "Structural coverage" means the project is in part of the structure space with no or few published structures.

[30] BIG and MEGA domain families were defined by the PSI-2 Target Selection Committee as having high value for extensive coverage. These familes contained hundreds to tens of thousands of members and many subfamilies which could not be modeled well due to a lack of structural coverage.

[31] Same as above.

[32] This typo occurs in the raw data.

[33] Same as above.

[34] Same as above.

contains a field where labs describe the protocols they used to conduct each stage of the trials. One type of protocol is the selection protocol. For example, 15 projects were selected because of the protocol "TSel_101," which states "These proteins are important for cell development." I read the descriptions associated with each selection protocol and manually classified whether each protocol was "novel" and/or "biomedical."[35] Then I set $novel_i$ equal to 1 if the project was selected due to a "novel" protocol. I set $biomedical_i$ equal to 1 if the project was selected due to a "biomedical" protocol.

Lastly, TargetTrack has a field that contains a list of reference IDs of each molecule in large-scale bioinformatics databases.[36] These reference ids may yield additional information. Whenever the list of reference ids contains BIG and MEGA reference ids,[37] I set $novel_i$ equal to 1.

A.2 Lab Funding

The NIGMS released the following FOAs directly tied to the PSI.

Table A1: Funding opportunity announcements (FOA) tied to PSI

Id	Title	Year
RFA-GM-99-009	PILOT PROJECTS FOR THE PROTEIN STRUCTURE INITIATIVE (STRUCTURAL GENOMICS)	1999
PA-99-116	PROTEIN STRUCTURE INITIATIVE (STRUCTURAL GENOMICS)	1999
PA-99-117	PROTEIN STRUCTURE INITIATIVE (STRUCTURAL GENOMICS) – SBIR/STTR	1999
RFA-GM-00-006	PILOT PROJECTS FOR THE PROTEIN STRUCTURE INITIATIVE (STRUCTURAL GENOMICS)	2000
RFA-GM-05-001	LARGE-SCALE CENTERS FOR THE PROTEIN STRUCTURE INITIATIVE	2004
RFA-GM-05-002	SPECIALIZED CENTERS FOR THE PROTEIN STRUCTURE INITIATIVE	2004
RFA-GM-06-004	Structural Genomics Knowledgebase (U01)	2006
RFA-GM-10-004	PSI:Biology Knowledgebase (U01)	2009
RFA-GM-10-005	Centers for High-Throughput Structure Determination (U54)	2009
RFA-GM-10-006	Centers for Membrane Protein Structure Determination (U54)	2009
RFA-GM-10-007	Consortia for High-Throughput-Enabled Structural Biology Partnerships (U01)	2009
PAR-10-214	High-Throughput-Enabled Structural Biology Research (U01)	2010
PAR-11-176	High-Throughput-Enabled Structural Biology Partnerships (U01)	2011

[35] The manual classification is available upon request.

[36] These reference ids include, but are not limited to, the molecule's id in the Protein Data Bank (PDB), UniProt, and the National Center for Biotechnology Information (NCBI) database.

[37] See footnote 30.

These FOAs allowed me to search directly all grants associated with them in the NIH RePORT database. In addition to the FOAs, I performed a direct search of the labs' names and abbreviations using RePORT's advanced search functionality to obtain data on each labs' supplementary funding. The search term I used was (quotation marks included):

"[lab full name]" OR "[lab abbreviation]"

I then aggregate each lab's sum of research grants by year from the search results.

A.3 Matching Projects to UniProt Molecule Information

As a preliminary to using the UniProt data, I match projects from the TargetTrack information system to their molecule information on UniProt through two methods.

TargetTrack has a field containing a list of reference ids of each molecule i in large-scale bioinformatics databases. These reference ids include, but are not limited to, the molecule's id in the Protein Data Bank (PDB), UniProt, and the National Center for Biotechnology Information (NCBI) database. When the UniProt id of the molecule is available in this field, the mapping is direct. I also use the following id types, which easily convert into UniProt molecule id through UniProt's ID Mapping service (Huang et al., 2011; UniProt, 2021a):

- *PDB_ID*: a molecule's id in the Protein Data Bank (PDB), a database for 3D structures.
- *P_REFSEQ_AC*: a molecule's id in NCBI's RefSeq protein database.
- *EMBL*: a molecule's corresponding gene's id in European Molecular Biology Laboratory (EMBL)/GenBank/DNA Data Bank of Japan (DDBJ) CDS database.
- *P_ENTREZGENEID*: a molecule's corresponding gene's id in GeneID (Entrez Gene) database.
- *P_GI*: a molecule's GI number assigned by NCBI.

When the first method fails to find a match (usually due to an entirely missing reference id field or obsolete records in the relevant databases), I use a second method: directly searching the molecule's sequence of amino acids against all protein sequences in UniProt.[38] I perform this search using DIAMOND (Buchfink et al., 2015, 2021), a very fast algorithm for searching similar sequences. The diamond command I used was:

```
diamond blastp -d [database name] -q [input sequences in .fasta]
-o [output in .csv] -f 6 qseqid qlen sseqid slen evalue bitscore pident length
-b4.0 --top 5
```

It produces search results with the following variables:

- *qseqid*: query sequence's identifier (the full sequence in this case).
- *qlen*: query sequence's length.
- *sseqid*: search result's UniProt id.

[38] Downloadable in .fasta format at https://www.uniprot.org/downloads.

- *slen*: search result's length.
- *evalue*: the number of expected hits of similar quality that could be found just by chance in a random database of the same size. E-value is a commonly used measure for the degree of similarity between the query sequence and the search result.
- *bitscore*: the required size of a sequence database in which the current match could be found just by chance. Bit score does not depend on the size of the database and is a common alternative measure for the degree of similarity between the query sequence and the search result.
- *pident*: percentage of identical matches between the query sequence and the search result over the alignment length.
- *length*: the alignment length between the query sequence and the search result.

If the query sequence's best match search result, determined by the e-value, a standard metric for assessing sequence similarity, has at least 95% *pident* and the alignment length *length* is at least 67% of both *qlen* and *slen*, I map the query sequence to the result sequence's UniProt id.

I was able to match 262,984 (78.4%) of the 335,553 projects to their UniProt entries through the id mapping method and match an additional 58,593 (17.5%) projects through the direct search. Overall, I was able to map 321,577 (95.8%) projects to their UniProt entries. I then used UniProt's programmatic access for individual entries (UniProt (2021b)) to pull each molecule's information from UniProt. I successfully pulled this information for 319,986 (95.4%) projects.

A.4 Data Glossary

This paper uses hundreds of project characteristics extracted from a variety of sources. This data glossary offers a comprehensive view of these variables.

* Variable is included in the characteristics $\boldsymbol{X}_{ijt}$ in training $\tilde{F}_t(\Omega_t)$.
† Variable is included in the characteristics $\boldsymbol{X}_{ijt}$ in training $F^*(\Omega_T)$.
‡ Variable is included in the characteristics $\boldsymbol{X}_{ijt}$ in training $ridge((\boldsymbol{X}, citation)_T)$.
§ Variable is included in the characteristics $\boldsymbol{X}_{ijt}$ in training $ridge((\boldsymbol{X}, \Delta download)_T)$.
Please see Appendix B for these models.

Table A2: Data glossary

Variables	Description
$[\text{4 cap letters then 6 digits}]_i$*†	Amino acid attributes from the AAindex database (Kawashima et al. (2007)). Each attribute had an identifier that had four capital letters followed by six digits. I started with the 567 attributes in AAindex1, and then normalized and clustered them to a set of around 30 attribute classes as in Babnigg & Joachimiak (2010). I used scikit-learn's implementation of affinity propagation clustering, which automatically picked 34 clusters. I then kept the cluster center of each class. For each cluster center attribute, I calculated the local average value, the local minimum, and the local maximum of the sum of the attribute in a seven-amino acid sliding window for molecule i as in Babnigg & Joachimiak (2010). This resulted in 102 variables.
$[\text{consortium abbreviation}]_{ijt}$*†‡ §	Binary variable = 1 if trial j_i was conducted by the given consortium at time t. Only consortia with more than 70 observations of projects in the TargetTrack database have their corresponding variables. 36 variables in total.
$[\text{gene}]_i$†‡ §	Binary variable = 1 if molecule i is coded for by the given gene. From UniProt. 48,548 variables in total. Most variables are very sparse. For † I only include genes that have occurred more than 200 times in my data. For ‡ and § I only include genes that are associated with at least one molecule whose structure was successfully published in my sample.
$[\text{keyword}]_i$†‡ §	Binary variable = 1 if molecule i is associated with the given keyword in UniProt. Examples of keywords include "Alzheimer disease," "Antioxidant," "RNA-binding," "Viral envelope protein." 1,053 variables in total. Most variables are very sparse. For † I only include keywords that have occurred more than 200 times in my data and remove the keyword "3D-structure" because this is the outcome. For ‡ and § I only include keywords that are associated with at least one molecule whose structure was successfully published in my sample.
$[\text{superkingdom-phylum}]_i$*†‡ §	Binary variable = 1 if molecule i comes from an organisim in the specific superkingdom and phylum. From UniProt. Due to the large number of species molecules in TargetTrack represent, I do not go down the UniProt taxonomy below phylum. 81 variables in total.
$aminoAcid\_[X]_i$*†	Counts the number of times amino acid "X" is in molecule i. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Contents of certain amino acids are linked to more successes of trials (Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).

$aminoAcidPercent\_[X]_i$*†	Calculate the amino acid "X" content in molecule i in percentages. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Contents of certain amino acids are linked to more successes of trials (Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).
$biomedical_i$*†‡ §	Binary variable $= 1$ if project i was biomedically important. See Appendix A.1 for variable construction.
$citation_{iy}$	five-year citations and mentions of project i published in year y, from PDBe (Varadi et al. (2020)). When multiple structures were published on molecule i, I take the mean values of the five-year citations and year of publication.
$download_{im}$	Number of downloads of published project i in month m across the three major structure databases in the world, from wwPDB (wwPDB (2013)). Available for Aug 2007–Nov 2013.
$\hat{E}(citation_{iy})$	Expected five-year citations and mentions of project i published in year y. See Appendix B.2 for construction.
$\hat{E}(download_{iy})$	Expected five-year downloads of project i published in year y. See Appendix B.3 for construction.
$\hat{E}_{\tilde{F}_t}(p_{ijt})$	Best-effort replication of the labs' posterior expectation of the probability of success of trial j_i on day t. See Appendix B.1 for construction.
$eukaryote_i$*†‡ §	Maximal percentage identity of molecule i to any eukaryotic molecule. To construct this variable, I search each molecule i against all UniProt protein sequences in the Eukaryota superkingdom (UniProt (2021c)). From the search results, I take the maximal percentage identity of i to any eukaryotic molecule as the variable $eukaryote_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. $evalue$ is a well-understood metric for search quality in this field. If there are no search results meeting the cutoff, I let $eukaryote_i = 0$.
$exposedAminoAcid\_[X]_i$*†	Counts the number of times amino acid "X" is on the predicted exposed surface of molecule i. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Contents of certain amino acids on the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).

$exposedAminoAcidPercent\_[X]_i$*†	Calculate the amino acid "X" content on the predicted exposed surface of molecule i in percentages. 20 variables for each of amino acids A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Contents of certain amino acids on the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).
$extinctCoeffReduced_i$*†	Molar extinction coefficient of molecule i with reduced cysteines. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Slabinski et al. (2007b) used the extinction coefficient as a feature to predict project success.
$extinctCoeffOxidized_i$*†	Molar extinction coefficient of molecule i with disulfid bridges. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Slabinski et al. (2007b) used the extinction coefficient as a feature to predict project success.
$funding_{ly}$	Total sum of research grants consortium l received from NIH in year y. See Appendix A.2 for variable construction.
$gaps_i$*†	The average number of insertions in molecule i's alignment compared to homologs in UniProt protein sequences. Computed by searching sequence i against UniProt protein sequences using DIAMOND (Buchfink et al. (2015, 2021)). The output variable *gaps* captures this value. Insertions were included as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Jahandideh et al. (2014).
$gapOpen_i$*†	The average number of insertion openings in the alignment compared to homologs in UniProt protein sequences. Computed by searching sequence i against UniProt protein sequences using DIAMOND (Buchfink et al. (2015, 2021)). The output variable *gapOpen* captures this value. Insertions were included as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Jahandideh et al. (2014).
$gravyIndex_i$*†	Grand average of hydropathicity index (GRAVY) of molecule i, used to represent the hydrophobicity value of a molecule. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Hydrophobicity is a key determinant of success of trials (Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).
$hasPrevSuccess_{ijkt}$*	Binary variable $= 1$ if at least one previous trial on molecule i successfully completed stage k before date t.
$hasPrevFailure_{ijkt}$*	Binary variable $= 1$ if at least one previous trial on molecule i failed at stage k before date t.

$human_i$*†‡ §	Maximal percentage identity of molecule i to any human molecule. To construct this variable, I search each molecule i against all UniProt protein sequences in the Homo sapiens (human) species (UniProt (2021d)). From the search results, I take the maximal percentage identity of i to any human molecule as the variable $human_i$. Due to potentially large number of search results, the search algorithm DIAMOND (Buchfink et al. (2015, 2021)) by default cuts off results at $evalue = 0.001$. $evalue$ is a well-understood metric for search quality in this field. If there are no search results meeting the cutoff, I let $human_i = 0$.
$instabilityIndex_i$*†	Instability index of molecule i, which is an estimate of the stability of the protein in a test tube. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Instability Index was included as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Jahandideh et al. (2014).
$isoelectricPoint_i$*†	Isoelectric point of molecule i. Calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module. Isoelectric point is a key determinant of success of trials (Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).
$membrane_i$*†‡ §	Binary variable $= 1$ if project i's UniProt information contains the word "membrane."
$molecularWeight_i$*†‡ §	Molecular weight of molecule i, calculated using Biopython's ProteinAnalysis function from Bio.SeqUtils.ProtParam module.
$novel_i$*†‡ §	Binary variable $= 1$ if project i was novel. See Appendix A.1 for variable construction.
$p^*_{ij,k-1,t_{k-1}}$†	The predicted probability of success of stage $k-1$ of project-trial j_i that started in period t_{k-1}. If $k = 0$, this variable is set to 1.
$percentCoil_i$*†	Predicted percentage of coil secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Jahandideh et al. (2014).
$percentCoiledCoil_i$*†	Percentage of coiled-coil regions in molecule i from UniProt. Coiled-coil regions were used in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014).
$percentDisordered_i$*†	Predicted percentage of disordered region in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Disordered region was used as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014).

$percentDisorderedUniprot_i$*†	Percentage of disordered region in molecule i from UniProt. Disordered region was used as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014).
$percentExposed_i$*†	Predicted percentage of amino acids on the exposed surface of molecule i. Exposed surface was predicted using the NetSurfP (Klausen et al. (2019)) program with the cutoff of relative solvent accessibility (rsa) > 0.25. Extent of the exposed surface of the molecule are linked to more successes of trials (Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014)).
$percentHelix_i$*†	Predicted percentage of helix secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014).
$percentLowComplexity_i$*†	Predicted percent low-complexity regions in molecule i. Computed using the SEG program (Wootton (1994)). Low-complexity regions were used as features in Slabinski et al. (2007a,b); Jaroszewski et al. (2008).
$percentSignalPeptide_i$*†	Percentage of signal peptide in molecule i. From UniProt. Slabinski et al. (2007a,b); Price et al. (2009a); Babnigg & Joachimiak (2010); Jahandideh et al. (2014) state molecules containing signal peptides have very low chances of success.
$percentStrand_i$*†	Predicted percentage of strand secondary structure in molecule i. Predicted using the NetSurfP (Klausen et al. (2019)) program. Secondary structure features were used in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014).
$percentTransmembraneHelices_i$*†	Percentage of transmembrane helices in molecule i. From UniProt. Transmembrane helices were used as a feature in Slabinski et al. (2007a,b); Jaroszewski et al. (2008); Price et al. (2009a,b); Babnigg & Joachimiak (2010); Jahandideh et al. (2014) .
$pfam_i$	A list of protein families associated with molecule i, from UniProt (UniProt (2021b)).
$phase1_{ijkt}$†	Binary variable $= 1$ if stage k of project-trial j_i started in phase 1 of PSI (pilot phase). I let this variable be 1 if the stage start year is before or in 2005. Phase 1 ended in 2004. However, based on Figures D1 and 6, one can clearly see that 2004 and 2005 are transition periods: output quantity jumped up in 2004. I therefore let 2004 and 2005 be part of both Phase 1 and Phase 2.

$phase2_{ijkt}$†	Binary variable = 1 if stage k of project-trial j_i started in phase 2 of PSI (production phase). I let this variable be 1 if the stage start year is between 2004 and 2010. Phase 2 is between 2005 and 2008. However, based on Figures D1 and 6, one can clearly see that 2009 and 2010 are transition periods: output quantity stayed high but citations reversed the trend. I therefore let 2009 and 2010 be part of both Phase 2 and Phase 3.
$phase3_{ijkt}$†	Binary variable = 1 if stage k of project-trial j_i started in phase 3 of PSI (biomedical phase). I let this variable be 1 if the stage start year is 2009 and beyond.
$prevPub_{iy}$*†‡ §	Number of publications on molecule i by the start of year y, from UniProt (UniProt (2021b)).
$prevStruct_{iy}$*†‡ §	Number of already published structures in the same protein families associated with molecule i by the start of year y. To construct this variable, I first pull from UniProt the list of protein families $pfam_i$ associated with molecule i. I then obtain a mapping of each protein family to its associated structures from EMBL-EBI (2021) and the structures' publication dates (I take the structure's deposition date to the PDB as the publication date) from Varadi et al. (2020). Merging the datasets results in $prevStruct_{iy}$. If i is associated with multiple protein families, I take the average of the number of already published structures in each protein family associated with i.
$prevSuccesses_{ijkt}$*	Number of previous trials on molecule i that have successfully completed stage k before date t.
$prevTrials_{ijkt}$*	Number of previous trials on molecule i that have reached stage k before date t.
$refId_i$	A list of reference ids of molecule i in TargetTrack, used to map i to its information in UniProt.
seq_i	Sequence representation of molecule i's amino acids, unique identifier of project i.
$seqLength_i$*†‡ §	The number of amino acids in molecule i.
$simPrevProj_{it}$	The maximal degree of similarity between project i and all previously attempted projects at time t, measured by the bit score (see Appendix A.3 for the definition of bit score). Computed by searching sequence i against all sequences attempted before time t using DIAMOND (Buchfink et al. (2015, 2021)). The maximum of the output variable $bitscore$ among research results was used as $simPrevProj_{it}$.

$surfaceRuggedness_i$*†	Surface ruggedness of molecule i, defined by the total accessible surface of molecule i divided by the accessible surface predicted based on molecular mass. The total accessible surface of the molecule i is calculated by summing the predicted absolute solvent accessibility of each amino acid from NetSurfP (Klausen et al. (2019)). The accessible surface predicted based on molecular mass is calculated using the formula $6.3(molecularMass)^{0.73}$ (Miller et al. (1987)). Jahandideh et al. (2014) used this variable as a feature.
$trialId_{ij}$	Trial id of project-trial j_i, unique identifier of trial j_i in TargetTrack.
$\widehat{Var}_{\tilde{F}_t}(p_{ijt})$	Best-effort replication of the posterior variance of the labs' beliefs about the probability of success of trial j_i on day t. See Appendix B.1 for construction.
Y_{ijt}	Binary variable $= 1$ if trial j_i on date t was successful.
Y_{ijkt}	Binary variable $= 1$ if intermediate stage k of trial j_i on date t was successful. $Y_{ij0t} = 1$ if DNA was successfully cloned. Y_{ij1t} is only defined when $Y_{ij0t} = 1$ and is equal to 1 if protein was successfully expressed. Y_{ij2t} is only defined when $Y_{ij0t} = 1$ and $Y_{ij1t} = 1$ and is equal to 1 if protein was successfully purified. Y_{ij3t} is only defined when $Y_{ij0t}, Y_{ij1t}, Y_{ij2t} = 1$ and is equal to 1 if protein was successfully crystalized for X-ray crystallography or prepared for NMR or cryo-EM. Y_{ij4t} is only defined when all previous stages were successful and is equal to 1 if the structure was successfully produced and deposited to the Protein Data Bank (PDB) for publication.

B Constructing Posteriors

B.1 Random Forest for Trial Success Probabilities

There are two kinds of models of trial success probability in this paper. The first one is a model that captures how labs formed posterior beliefs. It does not have to produce an unbiased estimate of the true probability of success of a trial. However, it has to produce an unbiased estimate of the labs' perceived posterior beliefs about the probability of success. My implementation of $\tilde{F}_t$ closely follows the machine learning approach the labs described in published journal articles.

The second one is a model of the true data generating process of trial success probability F^*, which is used in simulating counterfactual outcomes. Estimating F^* is different from estimating the posterior using $\tilde{F}_t$ because F^* needs to produce an unbiased estimate of the true probability of success of a trial. As such, my implementation of F^* deviates from $\tilde{F}_t$ in several ways to correct the potential bias of and improve upon the machine learning systems the labs described.

In this appendix, I first explain my implementation of $\tilde{F}_t$; then I move on to discuss how my implementation of F^* deviates from that of $\tilde{F}_t$.

B.1.1 Implementation of $\tilde{F}_t$

My implementation of $\tilde{F}_t$ fits stage-specific models to account for information embodied in outcomes of intermediate stages. Recall that each trial j_i has multiple sequential stages and the overall probability of success of j_i is equal to the product of the probabilities of success for all sequential stages $p_{ijt} = \prod_{k=0}^{4} p_{ijkt}$. The intermediate outcomes Y_{ijk} for stages $k = 0, 1, ...$ up to the point when the overall trial failed/succeeded provides information for future trials' potential.

For a given quarter $q(t)$ and each of the stages $k = 0, 1, 2, 3, 4$, I let the information set $\Omega_{k,q(t)}$ consist of project-trial outcomes realized before quarter $q(t)$ at stage k and these project-trials' characteristics. Following when the labs started to use machine learning to form posterior, I let $q(t)$ to be between 2005 and 2015. For $q(t) = 2005Q1$, I use the trial outcomes realized before 2005 and these trials' characteristics as the initial information set Ω_{2005Q1}. Project-trial characteristics $\boldsymbol{X}_{ijkt}$ I use for training $\tilde{F}_t$ and prediction of labs' posterior beliefs fall under three categories:[39]

- Physicochemical properties of molecule i based on scientific reasoning. These variables were identified by the series of journal articles the labs published and were quite similar across labs and time.
- Other characteristics of project i, for examples, novelty, biomedical importance, and the number of prior publications on molecule i.
- Past successes and failures of project i at stage k.

Then, for the given quarter $q(t)$ and each of the stages $k = 0, 1, 2, 3, 4$, I fit a random forest model $\tilde{F}_{k,q(t)}(\Omega_{k,q(t)})$ using `RandomForestClassifier` from python package `scikit-learn`. Random forest is an *ensemble*[40] machine learning method. The algorithm constructs a large number of decision trees at training time. Each decision tree is a learning model that aims to find the project-trial characteristics predictive of success/failure in the training set. When it comes to prediction, the trained random forest classifier $\tilde{F}_{k,q(t)}(\Omega_{k,q(t)})$ would pool individual trees and average predicted values of $\{\hat{p}_{ijkt}^{ntree}\}$ from individual trees as the final output. As Jahandideh et al. (2014) did, I set the number of trees in the random forest equal to 1000.

Decision trees and random forests are known for often overfitting without regularization. To avoid overfitting, I regularize by restricting the hyperparameters *max_depth*,[41] *min_samples_leaf*,[42] *max_features*,[43] and *min_samples_split*.[44] I perform model selection with a grid search of the

[39] Please see Appendix A.4 for the full list of variables used.

[40] Ensemble methods use multiple learning models to obtain better predictive performance than could be obtained from any of the constituent learning models alone.

[41] This hyperparameter determines the maximum depth of each decision tree.

[42] This hyperparameter determines the minimum number of observations a node in the decision tree must have before it can be split.

[43] This hyperparameter determines the maximum number of features to consider when looking for the best split.

[44] This hyperparameter determines the minimum number of observations required to split a node.

combinations of the four hyperparameters.[45] For each hyperparameter combination, I evaluate the model with five-fold cross validation using `scikit-learn`'s `cross_validate` function. In each iteration of the cross-validation, the function fits a random forest on four out of five cross-validation folds and then computes the cross-validation score by comparing the model's predictions with the actual data from the remaining fold. I use the average log likelihood (`log_loss` scoring in `scikit-learn`) as the cross-validation scoring method. I choose the hyperparameter combination that maximizes the average log likelihood in cross-validation.

After training the models $\tilde{F}_{k,q(t)}$ for $k = 0,1,2,3,4$ for a given $q(t)$, I predict $\widehat{E}_{\tilde{F}_{q(t)}}(p_{ijt})$ and $\widehat{Var}_{\tilde{F}_{q(t)}}(p_{ijt})$ for each project-trial in the choice set C_{lt} at decision time t as follows. I first collect the predictions $\{\hat{p}^{ntree}_{ijkt}\}$ from the 1000 individual decision trees in $\tilde{F}_{k,q(t)}(\Omega_{k,q(t)})$, and then compute $\hat{p}^{ntree}_{ijt} = \prod_{k=0}^{4} \hat{p}^{ntree}_{ijkt}$. There are 1000 values in the set $\{\hat{p}^{ntree}_{ijt}\}$. I let

$$\widehat{E}_{\tilde{F}_{q(t)}}(p_{ijt}) = \bar{p}^{ntree}_{success,ijt}, \tag{14}$$

$$\widehat{Var}_{\tilde{F}_{q(t)}}(p_{ijt}) = s^2(p^{ntree}_{success,ijt}) \tag{15}$$

Although I extensively reference the labs' implementations of machine learning systems when I implement $\tilde{F}_t$, my estimate of the posterior is not a perfect replica of the labs' posteriors. I note why replicating perfectly the labs' posterior beliefs would be difficult and where my implementation corresponds to and deviates from the labs' learning and updating process below:

- I include in $\boldsymbol{X}_{ijt}$ the set of physicochemical properties of molecules identified in the labs' published journal articles (Slabinski et al., 2007a,b; Jaroszewski et al., 2008; Price et al., 2009a,b; Babnigg & Joachimiak, 2010; Jahandideh et al., 2014). This set is the union of the sets of such properties in different articles (to minimize the risk of selection on unobservables) and is fixed for all labs and time periods in my implementation. In contrast, though similar across lab and time, the set of physicochemical properties the labs used in training and prediction still varied. It is impossible to capture all of these potential variations during the labs' long operational history (some may not have been recorded by the published articles).

- The construction of some variables in $\boldsymbol{X}_{ijt}$ requires using software packages that are constantly being updated or have become obsolete. I make my best effort to construct variables using methods as close to the labs' original approach as possible (see Appendix A.4).

- The $\boldsymbol{X}_{ijt}$ in my implementation includes past trial outcomes of projects while the labs' implementations did not explicitly include those characteristics. Still, it is reasonable to believe that researchers working on a project would update their beliefs on the potential of the project upon seeing a trial success/failure.

[45] To reduce computational burden, I do not perform model selection for all $\tilde{F}_{k,q(t)}$. Rather, for each $k = 0,...,4$, I construct $\Omega_{k,T}$ using all outcomes at stage k and only perform model selection for $\tilde{F}_{k,T}$ on this full training set. I then use the selected hyperparameters to train the models $\tilde{F}_{k,q(t)}$ where $q(t) = 2005Q1, 2005Q2, ..., 2015Q4$. The set of *max_depth* used in grid search is $[int(log(sample\_size,2)), 2 \cdot int(log(sample\_size,2)), 3 \cdot int(log(sample\_size,2)), 4 \cdot int(log(sample\_size,2))]$. The set of *min_samples_leaf* used in grid search is $[1,2,4]$. The set of *max_features* used in grid search is $[0.1, 0.2, 0.3, 0.4]$ of the total number of features. The set of *min_samples_split* used in grid search is $[8, 16, 32, 64, 128]$.

- I use random forest as the model of posterior updating for all labs and time periods. In contrast, the machine learning models the labs used in training and prediction varied across lab and across time. It is impossible to capture all of these potential variations during the labs' long operational history (some may not have been recorded by the published articles).

- I set the frequency of "updating" and refitting models at the quarterly interval. In contrast, the labs' actual belief updating frequency is not clearly documented. I use the quarterly interval because training models at a finer interval, such as at the daily frequency, places large computational and storage burden. The day-to-day change of the information set Ω was also relatively small. Therefore, to improve computational tractability, I coarsen the frequency of refitting new models to quarterly.

- My model predicts the overall potential of success of a trial while the labs' implementations focused on predicting the potential of success of bottleneck stages of a trial. That is, for stages where success rates were usually reasonable (for example cloning the DNA), the labs were often not explicitly reliant on something as rigorous as supervised machine learning systems to form and update beliefs, while they were explicitly reliant on such systems for predicting the potential of success in crystallizing a molecule and studying its structure through X-ray crystallography.

- The output the labs' systems produced may not exactly be $\widehat{E}_{\tilde{F}_{q(t)}}(p_{ijt})$ and $\widehat{Var}_{\tilde{F}_{q(t)}}(p_{ijt})$. For example, the model in Slabinski et al. (2007b) predicted the probability of success as an intermediate outcome. The final output was an integer score between 1 and 5, where 1 represents "optimal" and 5 represents "very difficult." The labs' systems did not always predict $\widehat{Var}_{\tilde{F}_{q(t)}}(p_{ijt})$. When they did, the measure took the form of comparing predictions from multiple models side by side (Slabinski et al., 2007a,b; Babnigg & Joachimiak, 2010; Jahandideh et al., 2014). It is reasonable to believe that labs had some understanding that predictions from different models (or submodels of an ensemble model) differed, and looking at how those predictions varied was valuable, though they did not percolate the idea down to form an additional metric just to measure that variation. This seems consistent with the notation that the labs used heuristics to guide their exploration of high-variance projects.

B.1.2 Implementation of F^*

The implementation of F^* is almost identical to that of $\tilde{F}_t$ except for a few deviations. First of all, a new model $\tilde{F}_{k,q(t)}$ (for stages $k = 0,..,4$) is trained for every quarter $q(t)$ between 2005Q1 and 2015Q4, incorporating new trial outcomes realized in each quarter. In contrast, F^*_k (for stages $k = 0,...,4$) is trained only on the full information set Ω_T. Ω_T covers the characteristics and outcomes of all trials in my trial allocations and outcomes dataset in the entire sample period.

Second, F^* uses additional covariates to correct the potential bias of $\tilde{F}_t$ in predicting trial success probabilities. The model $\tilde{F}_t$ may be biased in predicting trial success probabilities because it does not account for the propensity of observing a specific stage of a trial. To see this, think about the probability of success of stage 1 of a trial. We observe stage 1 of a trial only if stage 0 of the trial was successful. If the probabilities of success of stages 0 and 1 are positively correlated, then we are more likely to observe stage 1 of trials that are more likely to succeed in stage 1. Therefore,

models trained with the observed data on stage 1 would produce prediction results that are positively biased. Correcting this bias is simple if we assume that the selection into observing a given stage is only based on observable characteristics of trials: we can use the predicted probability of success of the previous stage as the propensity score of observing the given stage. As such, I include $p^*_{ij,k-1,t_{k-1}}$, the predicted probability of success of stage $k-1$ of trial j_i that started in period t_{k-1}, as a covariate when I train F^*_k. For stage $k=0$, I set this variable to 1. The labs' published articles offer no discussion about this source of bias, so I do not include this variable in training $\tilde{F}_t$.

Another difference between F^* and $\tilde{F}_t$ is that F^* does not include variables on previous trial outcomes as covariates. In simulations, all previous trial outcomes of a project are simulated and should not shift the project's true probability of success; therefore, the simulated counterfactual outcome of a trial should not be based on the simulated previous outcomes.

To further improve the predictive power of F^*, I include in F^* project-trial characteristics the labs did not use in their machine learning systems. I include keywords and genes associated with the molecule i. I also include three phase-specific dummy variables to capture the effects of different phases of the grant program on the probabilities of success, as I learned during my conversations with NIH program officers that the labs underwent retooling corresponding to the changes of phases.

B.2 Ridge Regression for Citations

Let the model be $ridge(\boldsymbol{X}, citation)_T)$, where the training set $(\boldsymbol{X}, citation)_T$ represents the characteristics and $citation_{iy}$ of all published projects in my data. The goal of this model is to predict $E(citation_{iy}|\boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, citation)_T))$, the expected number of five-year citations a project i published in year y would generate conditional on the project's characteristics $\boldsymbol{X}_{iy}$.

The number of characteristics that could potentially predict higher citations is very large. Characteristics ranging from the organism the molecule i is from to the gene that expresses molecule i could all contribute to the biomedical significance and research interests on molecule i. The number of characteristics is on the order of hundreds, most of which are very sparse, while I only have 10,424 observations.[46] This calls for regularization to avoid overfitting.

I use a ridge regression from the python package `scikit-learn`. The project characteristics $\boldsymbol{X}_{iy}$ included for model fitting are shown in Appendix A.4. I choose the regularization hyperparameters using cross-validation with the `RidgeCV` function provided by the `scikit-learn` package.

I standardize the outcome variable $citation_{iy}$ by subtracting away the lab mean value of this variable and then dividing by the lab standard deviation as publications from different labs had large variations in citation numbers. When the model makes a prediction, I multiply the predicted value with the lab standard deviation and add the lab mean to get the predicted citations.

I assess model fit by comparing the actual citations with their out-of-sample predicted citations under five-fold cross validation. Figure B1 shows the distribution of the predicted citations correctly captures a high proportion of zero values in the actual data. Figure B2 shows a scatterplot of predicted citations against the actual citations. A linear regression of the predicted citations on the actual citations without constant shows an $R^2 = 0.580$.

The predicted citations $\hat{E}(citation_{iy}|\boldsymbol{X}_{iy}, Ridge((\boldsymbol{X}, citation)_T))$ has a y subscript because some

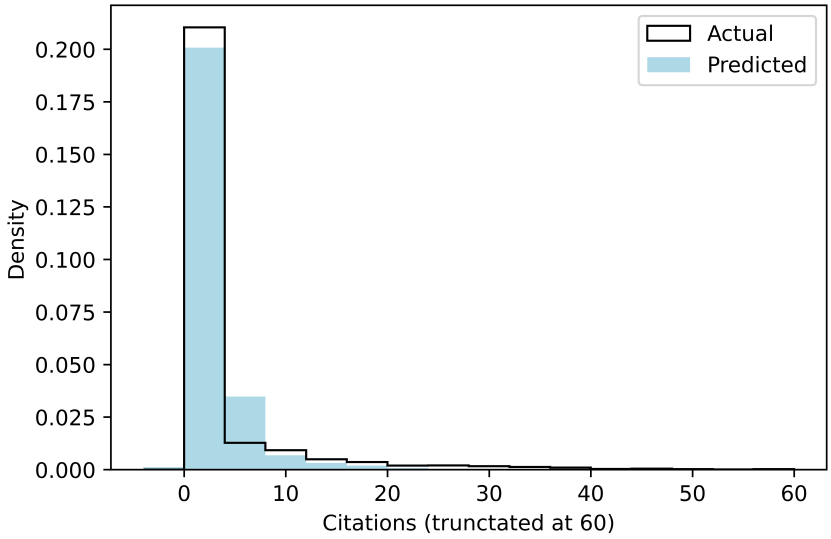
[46] The total number of published projects in my data is 10,501. 77 published projects in my data did not give the PDB ids of their publications so I was not able to map their citation information.

of the important characteristics vary with time, for example, the number of publications on molecule i prior to the year of publication of the structure. In Section 3.3, since we do not know which exact year the structure of each trial in the choice set would have been published if the trial was allocated, I remove the y subscript by averaging the predicted citations for each molecule across years so that

$$\hat{E}(citation_i) = \frac{1}{16} \sum_{y=2000}^{2015} \hat{E}(citation_{iy} | \boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, citation)_T)). \tag{16}$$

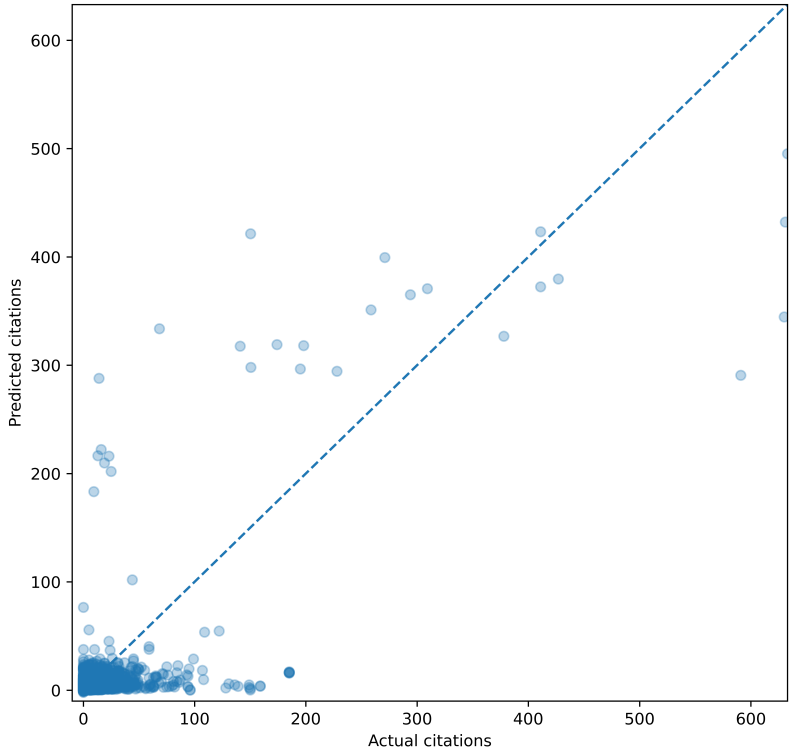
In simulations, I simulate both the outcome (success or failure) of an allocated trial and the date that outcome is realized. In that case, the y subscript is preserved for the predicted citations of the publication.

Figure B1: Distributions of actual citations and predicted citations



Note: Distributions are truncated at 60. Bin width is 4.

Figure B2: Scatterplot of predicted citations against actual citations



B.3 Ridge Regression for Downloads

Before we start predicting downloads, the raw download data needs to be transformed because the number of downloads shows a strong time trend over the lifecycle of a publication. The raw download data I obtain consists of monthly downloads on the structure level between Aug 2007 and Nov 2013 for all publicly available structures human beings know of. The number of structure-month observations is 5,484,800. As some structures in a given month observed were published a long time ago while some just got published, comparing these structures' raw download counts in a month would be misleading. As shown in Figure B3, downloads peak within a month since the publication of a structure and then sharply decline over the following months until reaching some steady level in approximately two years.

I perform a transformation of the raw download data to detrend it as follows. Let $pubAge(i,m)$ be the age of publication i (in months) in month m. To detrend, I first compute $\overline{download}_{pubAge(i,m)}$, the mean downloads of structures that have been published for $pubAge(i,m)$ months.[47] I then compute how much the number of downloads structure i had in month m deviates from this mean, $\widetilde{download}_{im} = download_{im} - \overline{download}_{pubAge(i,m)}$. I then define $\Delta download_i$, the average deviation of structure i's monthly downloads from the mean download trend, by the average of $\widetilde{download}_{im}$, in other words $\Delta download_i = \overline{\widetilde{download}_{im}}$. I treat the variable $\Delta download_i$ as the outcome variable. If there are multiple structures on the same project i, I take the mean of their

[47] I pool observations with $pubAge > 25$ months in computing this mean as the mean number of downloads flattens by 25 months since publication.

average deviations as $\Delta download_i$. I then match the download data with the project-trial characteristics of completed projects in my data.

Let the model be $ridge((\boldsymbol{X}, \Delta download)_T)$, where the training set $(\boldsymbol{X}, \Delta download)_T$ represents the characteristics and $\Delta download_i$ of all published projects in my data. The goal of this model is to predict $E(\Delta download_{iy} | \boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, \Delta download)_T))$, the expected average deviation of the structure's monthly downloads from the mean download trend for a project i published in year y, conditional on the project's characteristics $\boldsymbol{X}_{iy}$. There is a y subscript because some of the important characteristics vary with time, for example, the number of publications on molecule i prior to the year of publication of the structure.

The number of characteristics that could potentially predict higher downloads is very large. Characteristics ranging from those of molecule i's organism to those of molecule i's gene could all contribute to the level of interest on molecule i. The number of characteristics is on the order of hundreds, most of which are very sparse, while I only have 10,424 observations. This calls for regularization to avoid overfitting.

I use a ridge regression from the python package `scikit-learn`. The project characteristics $\boldsymbol{X}_{iy}$ included for model fitting are shown in Appendix A.4. I choose the regularization hyperparameters using cross-validation with the `RidgeCV` function provided by the `scikit-learn` package.

I assess model fit by comparing the actual $\Delta download_{iy}$ with their out-of-sample predicted values under five-fold cross validation. Figure B4 shows a comparison of the distributions. As ridge regression shrinks all regression coefficients towards zero, the distribution of the predicted values is narrower. Still, the predicted values capture the rank order of the actual data well. Figure B5 shows a scatterplot. The plot shows a relationship quite close to the line $y = x$. Figure B6 shows a binned scatterplot.

In Section 3.3 where we show descriptives, I perform additional transformations on the predicted value $\hat{E}(\Delta download_{iy} | \boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, \Delta download)_T))$. First, the predicted value has a y subscript because some of the important characteristics vary with time, for example, the number of publications on molecule i prior to the year of publication of the structure. Since we do not know which exact year the structure of each trial in the choice set would have been published if the trial was allocated, I remove the y subscript by averaging the predicted values for each molecule across years so that

$$\hat{E}(\Delta download_i) = \frac{1}{16} \sum_{y=2000}^{2015} \hat{E}(\Delta download_{iy} | \boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, \Delta download)_T)). \tag{17}$$

Second, the variable $\hat{E}(\Delta download_i)$ predicts the average deviation of monthly downloads from a trend and is difficult to interpret. I therefore transform this variable to a prediction of five-year downloads $\hat{E}(download_i)$ by using the following formula:

$$\hat{E}(download_i) = \sum_{pubAge=0}^{59} \overline{download_{pubAge}} + 60 \times \hat{E}(\Delta download_i), \tag{18}$$

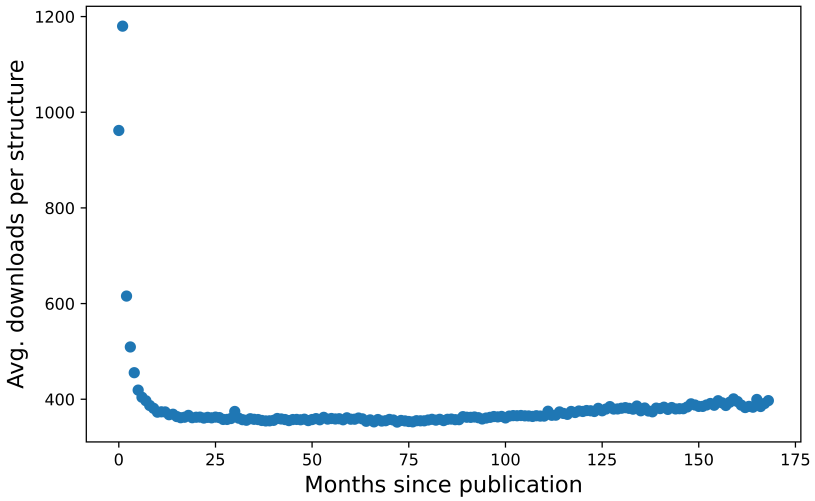
where $\sum_{pubAge=0}^{59} \overline{download_{pubAge}} = 23960.11$ is computed based on the full download data with

5,484,800 structure-month observations.

In simulations, I simulate both the outcome (success or failure) of an allocated trial and the date that outcome is realized. In that case, the y subscript is preserved for the predicted average deviation of monthly downloads of the published projects. Therefore the predicted five-year downloads becomes

$$\hat{E}(download_{iy}) = \sum_{pubAge=0}^{59} \overline{download_{pubAge}} + 60 \times \hat{E}(\Delta download_{iy} | \boldsymbol{X}_{iy}, ridge((\boldsymbol{X}, \Delta download)_T)). \tag{19}$$

Figure B3: Average downloads per structure in months since publication



Note: The plot is based on 5,484,800 structure-month observations of download counts between Aug 2007 and Nov 2013. Each blue dot aggregates in this data the average monthly downloads for structures published m months ago.

Figure B4: Distributions of predicted $\hat{E}(\Delta download_{iy})$ against actual $\Delta download_{iy}$

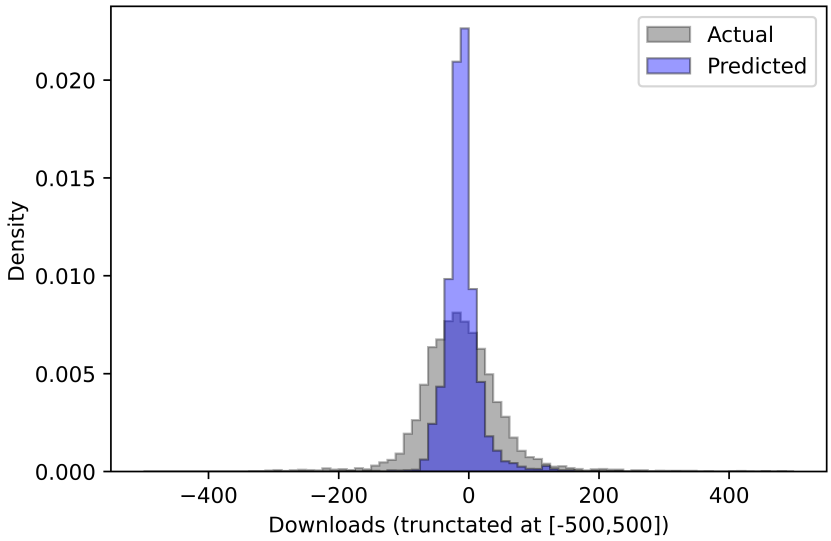


Figure B5: Scatterplot of predicted $\hat{E}(\Delta download_{iy})$ against actual $\Delta download_{iy}$

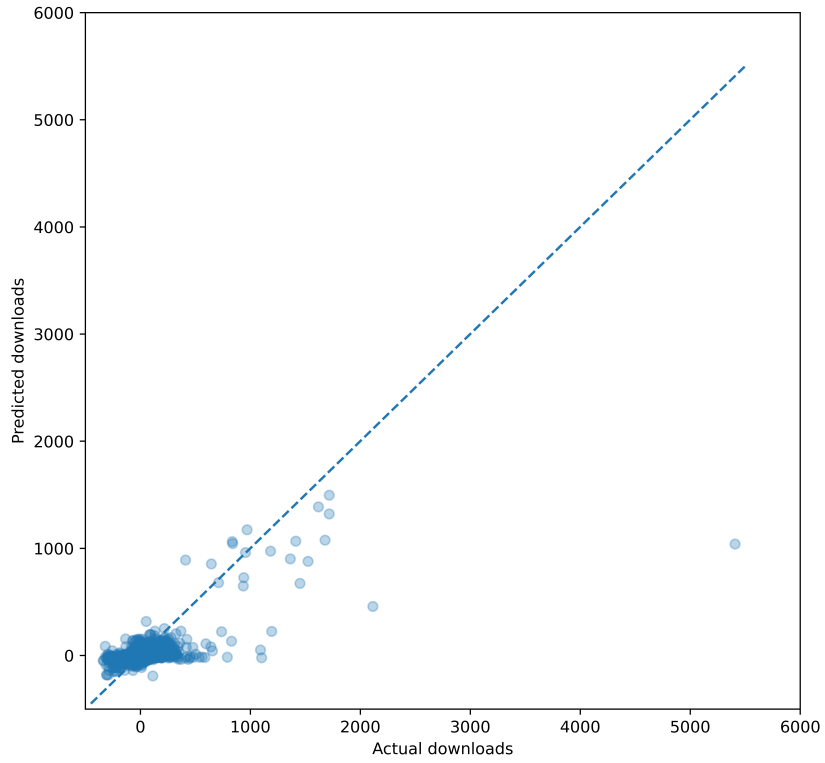
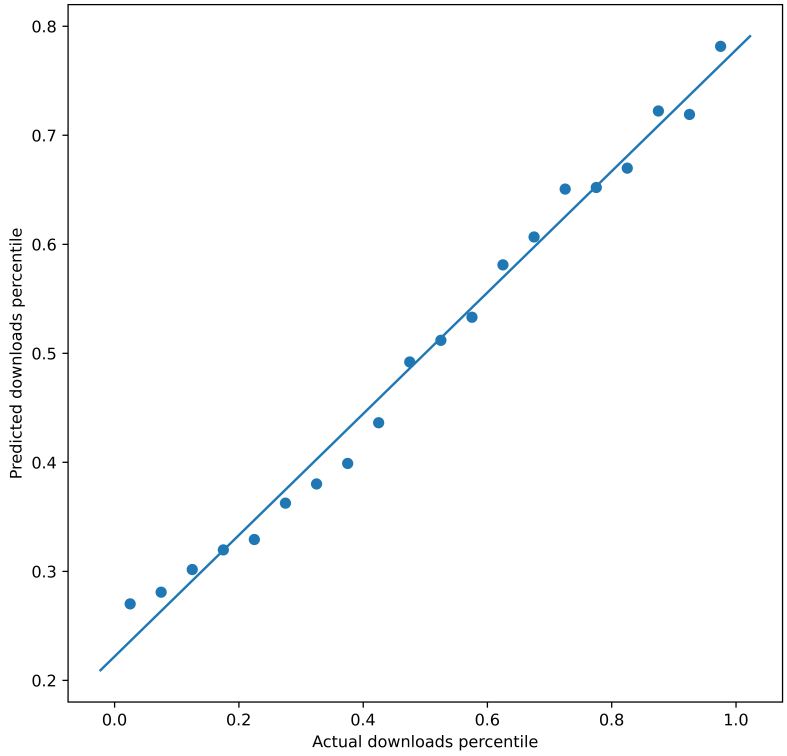


Figure B6: Binned scatterplot of predicted $\hat{E}(\Delta download_{iy})$ against actual $\Delta download_{iy}$



C Additional Details on Modeling and Estimation Procedure

C.1 Functional Form of Payoff Function $\pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})$

To begin, recall that the payoff $\pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})$ of project-trial j_i at t given action $\boldsymbol{a}_{lt}$ has a probability distribution depending on p_{ijt}. The lab or the economist does not perfectly know p_{ijt}, but previous outcomes of trials reveal information about it so one can form a posterior $\tilde{F}_t(p_{ijt}|\Omega_t)$. Integrating $\pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt}; \boldsymbol{\theta}_{Xl})$ over the posterior, we obtain the posterior expected payoff $\pi_{ijt}(\Omega_t, \boldsymbol{a}_{lt}; \boldsymbol{\theta}_{Xl})$.

I define a function $q(\boldsymbol{a}_{lt}, p_{ijt})$ that maps the probability of success to the probability of *payoff* of project-trial j_i at t given actions. When a project-trial is not allocated on day t, it does not pay off even though it may have a nonzero probability of success. Moreover, the labs often simultaneously allocated multiple trials to the same project. A successful trial j_i of project i should only receive payoff if the simultaneous trials $(j-m)_i, (j-m+1)_i, ..., (j-1)_i$ fail, because we have assumed only the first successful trial/publication on a project produces welfare. Each trial j_i is a Bernoulli trial with probability p_{ijt}. We can express $q(\boldsymbol{a}_{lt}, p_{ijt})$ as follows:

$$q(\boldsymbol{a}_{lt}, p_{ijt}) = a_{ijt}(1-p_{ijt})^m p_{ijt}, \tag{20}$$

where trials $(j-m)_i, (j-m+1)_i, ..., (j-1)_i$ are in the choice set C_{lt} and trial $(j-m)_i$ is the

smallest-numbered trial of project i in C_{lt}.[48] When $m = 0$, trial j_i is the smallest-numbered trial of project i in C_{lt} and its probability of payoff is simply $a_{ijt}p_{ijt}$.

I then specify a deterministic reward function, which captures the amount of payoff a lab will get when a trial pays off. I let the reward function $r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl})$ be a function of project i's characteristics $\boldsymbol{X}_{it}$ on day t. $\boldsymbol{\theta}_{Xl}$ are the welfare weights on $\boldsymbol{X}_{it}$ and are to be estimated. To reduce the number of parameters, I restrict $\boldsymbol{X}_{it}$ to correspond to the set of NIH evaluation metrics. $\boldsymbol{X}_{it}$ includes a constant 1 to capture preference for quantity; $novel_i$ and $prevStruct_{iy}$ to capture preference for novelty; $biomed_i$ and $prevPub_{iy}$ to capture preference for biomedical importance; and $human_i$, $eukaryote_i$, and $membrane_i$ to capture preferences for human, eukaryotic, and membrane proteins, respectively. I let $r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl})$ have a simple linear form

$$r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl}) = 1\cdot\theta_{quant,l} + biomed_i\cdot\theta_{biomed,l} + ... + membrane_i\cdot\theta_{membrane,l}. \tag{21}$$

Whenever a trial pays off, the lab receives a baseline amount $\theta_{quant,l}$ plus additional amounts depending on the other characteristics of the project.

We can then break the posterior expected payoff into a few pieces:

$$\begin{aligned}\int \pi_{ijt}(\boldsymbol{a}_{lt}, p_{ijt};\boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t) &= \int r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl})\cdot q(\boldsymbol{a}_{lt}, p_{ijt})\, d\tilde{F}_t(p_{ijt}|\Omega_t)\\ &= a_{ijt}\cdot r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl}) \underbrace{\int \overbrace{[(1-p_{ijt})^m p_{ijt}]}^{\text{let it be } M_{ijt}}\, d\tilde{F}_t(p_{ijt}|\Omega_t)}_{\text{estimated offline}}.\end{aligned} \tag{22}$$

Notice that M_{ijt} only depends on p_{ijt}. Since we have estimated $\tilde{F}_t(p_{ijt}|\Omega_t)$ offline (see Appendix B.1), we can estimate $E_{\tilde{F}_t}(M_{ijt})$ and $Var_{\tilde{F}_t}(M_{ijt})$ offline as well.[49]

Also notice that given the breakdown of the posterior expected payoff in equation (22), V^A_{ijt} does not depend on the full action vector $\boldsymbol{a}_{lt}$. It only depends on the action a_{ijt}. Plugging in equation (22) into V^A_{ijt} in equation (3) and evaluating it at $a_{ijt} = 1$, we obtain:

$$\begin{aligned}V^A_{ijt}(\Omega_t, a_{ijt}=1;\boldsymbol{\theta}_l) &= \int \pi_{ijt}(a_{ijt}=1, p_{ijt};\boldsymbol{\theta}_{Xl})\, d\tilde{F}_t(p_{ijt}|\Omega_t) \;+\; B_{ijt}(\Omega_t, a_{ijt}=1;\boldsymbol{\theta}_{Bl})\\ &= r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl})E_{\tilde{F}_t}(M_{ijt}) + B_{ijt}(\Omega_t, a_{ijt}=1;\boldsymbol{\theta}_{Bl}).\end{aligned} \tag{25}$$

For the main model, $V^A_{ijt}(\Omega_t, a_{ijt}=1;\boldsymbol{\theta}_l) = r(\boldsymbol{X}_{it};\boldsymbol{\theta}_{Xl})E_{\tilde{F}_t}(M_{ijt}) + \sqrt{\frac{\theta_{B1,l}}{j}} + \theta_{B2,l}\cdot(t - t'_{i,t})$. For

[48] $q(\boldsymbol{a}_{lt}, p_{ijt}) = a_{ijt}(1-p_{i,j-m,t})...(1-p_{i,j-1,t})p_{ijt}$. As all trials on day t share the same information set Ω_t, the posteriors for $p_{i,j-m,t}, ..., p_{ijt}$ are the same.

[49] Each trial j_i is a Bernoulli trial with probability of success p_{ijt},

$$E_{\tilde{F}_{q(t)}}(M_{ijt}) = E_{\tilde{F}_{q(t)}}((1-p_{ijt})^m p_{ijt}) = [1 - E_{\tilde{F}_{q(t)}}(p_{ijt})]^m \cdot E_{\tilde{F}_{q(t)}}(p_{ijt}), \tag{23}$$

$$\begin{aligned}Var_{\tilde{F}_{q(t)}}(M_{ijt}) &= Var_{\tilde{F}_{q(t)}}((1-p_{ijt})^m p_{ijt})\\ &= \{Var_{\tilde{F}_{q(t)}}(p_{ijt}) + [E_{\tilde{F}_{q(t)}}(p_{ijt})]^2\} \times \{Var_{\tilde{F}_{q(t)}}(1-p_{ijt}) + [E_{\tilde{F}_{q(t)}}(1-p_{ijt})]^2\}^m\\ &\quad - [E_{\tilde{F}_{q(t)}}(p_{ijt})]^2 \times \{[E_{\tilde{F}_{q(t)}}(1-p_{ijt})]^2\}^m.\end{aligned} \tag{24}$$

Plugging in $\hat{E}_{\tilde{F}_{q(t)}}(p_{ijt})$ and $\widehat{Var}_{\tilde{F}_{q(t)}}(p_{ijt})$ into the above equations, one obtains $\hat{E}_{\tilde{F}_{q(t)}}(M_{ijt})$ and $\widehat{Var}_{\tilde{F}_{q(t)}}(M_{ijt})$.

alternative model 1, where $B_{ijt}(\cdot) = 0$, this reduces to $V^A_{ijt}(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) = r(\boldsymbol{X}_{it}; \boldsymbol{\theta}_{Xl}) E_{\tilde{F}_t}(M_{ijt})$. For alternative model 2, $V^A_{ijt}(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) = r(\boldsymbol{X}_{it}; \boldsymbol{\theta}_{Xl}) E_{\tilde{F}_t}(M_{ijt}) + \psi(\cdot) r(\boldsymbol{X}_{it}; \boldsymbol{\theta}_{Xl}) Var_{\tilde{F}_t}(M_{ijt})$. Evaluating V^A_{ijt} at $a_{ijt} = 0$, we obtain $V^A_{ijt}(\Omega_t, a_{ijt} = 0; \boldsymbol{\theta}_l) = 0$ for all models.

Moreover, with this functional form, the second constraint in equation (4) will always be guaranteed by the solution. Notice that in most models including the main model, for all $j_i < j_i' \in C_{lt}$, $V^A_{ijt}(\Omega_t, a_{ijt} = 1; \boldsymbol{\theta}_l) \geq V^A_{ij't}(\Omega_t, a_{ij't} = 1; \boldsymbol{\theta}_l)$ because the two terms only differ by $E_{\tilde{F}_t}(M_{ijt}) \geq E_{\tilde{F}_t}(M_{ij't})$. For models based on the Gittins index, the two terms also differ by $Var_{\tilde{F}_t}(M_{ijt}) \geq Var_{\tilde{F}_t}(M_{ij't})$ and the constraint continues to be satisfied. In equation (10), we add an ε_{it} to both terms and the constraint continues to be satisfied.

C.2 Specifying Choice Set C_{lt}

As discussed in Section 2, the major labs in my data received new projects through three ways with close NIH involvement: 1) a centralized planning committee periodically assigned families of novel molecules; 2) the biomedical research community nominated projects; and 3) the labs determined projects of their own interest, which they reported to the NIH well in advance. These processes placed limits on the new projects the labs could plausibly consider when they made trial allocations. These limits allow me to considerably reduce C_{lt}.

I restrict C_{lt} to include only the following project-trials. For an older project i that the lab has attempted up until trial j_i in period $t' < t$, I include trials $(j+1)_i, ..., (j+n_t)_i$ in C_{lt}. For a new project i' that the lab has not attempted until t but attempts within the next six months, I include trials $1_{i'}, ..., (n_t)_{i'}$ in C_{lt}. One can also consider using alternative windows for the new projects, such as projects attempted within the next three months or nine months. Doing so changes the magnitudes of the estimates, but all qualitative results are the same as when using six months as the window. Likewise, adding some new projects that the lab could have considered but never actually attempted could change the estimates, but the qualitative results should stay the same.

I further reduce the sizes of the choice sets used in estimation by taking random subsamples of C_{lt}. The reason is related to computation. Labs at times allocate hundreds of project-trials on a day and they usually had tens of thousands of projects in their portfolios. The sizes of some choice sets C_{lt} could be on the order of millions. Given that we have thousands of periods, if we use the full choice sets C_{lt}, we need to compute log likelihood for billions of choices in each iteration of maximum likelihood. This would result in a very large memory burden and slow computation. Moreover, it is not necessary to include every possible choice in C_{lt} to consistently estimate $\boldsymbol{\theta}_l$. A random subsample of the choices on each side of the threshold value would be sufficient. Due to the sheer number of projects in each lab's portfolio, the set of actual trials is much smaller than the set of not-allocated trials. I therefore reduce the sizes of the choice sets for estimation by taking random subsamples of the latter on the level of project and date. Let the reduced choice sets be $C^R_{lt} \subset C_{lt}$. Table C1 shows the project-trials in C^R_{lt} after random sampling from C_{lt}.

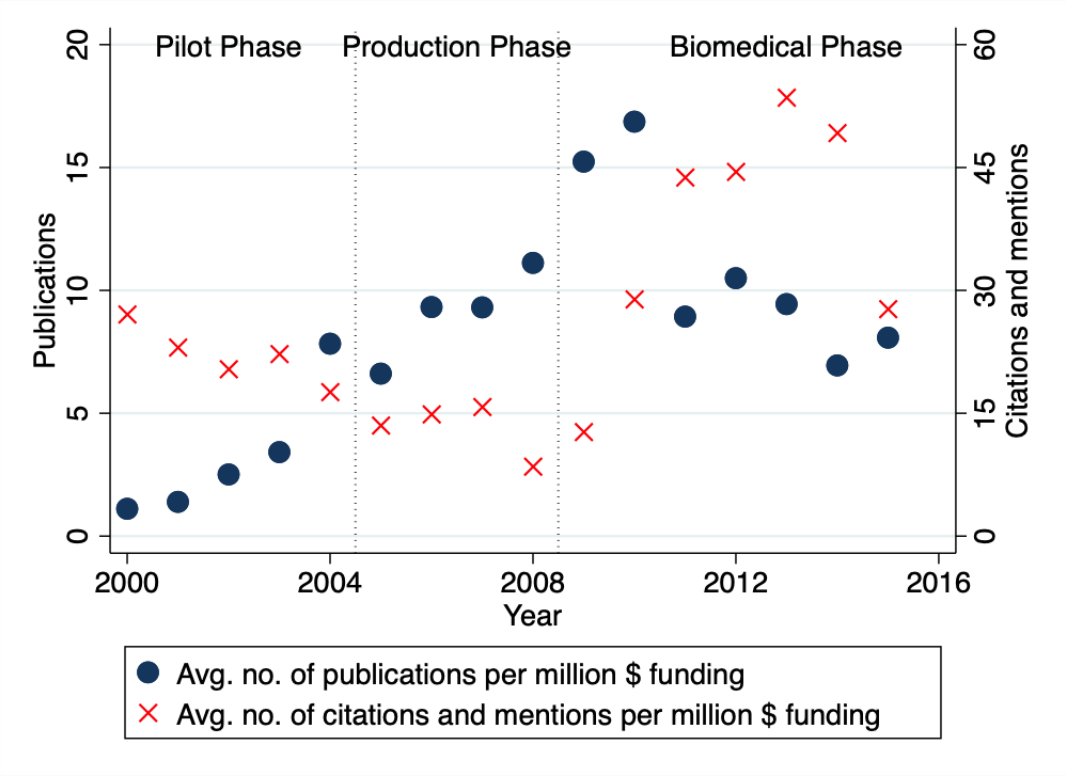
Table C1: Trials included in the reduced choice set for estimation

Project	Trial	Actually allocated on day t?	Notes
i	$(j+1)_i$	Y	Trials $(j+1)_i$ through $(j+m-1)_i$ were actually allocated on day t, include all in C_{lt}^R. When no trial was allocated on day t, m equals 1. Trials $(j+m)_i$ through $(j+n_{lt})_i$ were not actually allocated on day t, include one random trial $(j+r)_i$ in C_{lt}^R.
	$\vdots$	$\vdots$	
	$(j+m-1)_i$	Y	
	$(j+m)_i$	N	
	$\vdots$	$\vdots$	
	$(j+r)_i$	N	
	$\vdots$	$\vdots$	
	$(j+n_{lt})_i$	N	

Notes: For an older project that has been attempted before t, j equals the number of trials allocated to the project before t. For a new project, $j=0$. Trials in black are included in the reduced choice set C_{lt}^R. Trials in grey are in the choice set C_{lt} but are excluded from C_{lt}^R to reduce computational burden.

D Additional Results

Figure D1: Observed output: number of publications and citations



Note: The blue dots show the number of published structures on unique molecules in a given year divided by the lab consortium's funding in millions in that year. The red crosses show the number of 5-year citations and mentions the published structures in that year generated, divided by the lab consortium's funding in millions in that year. Each plot shows the average value across the four large lab in each year. The disaggregated values are in Figure 6.

Table D1: Comparison of likelihoods of different models, JCSG

Model	Free parameters in $B_{ijt}(\cdot)$	Log likelihood	Avg $\hat{P}(a^o_{ijt}=1;\boldsymbol{\theta}_l)$ actual allocation	Avg $\hat{P}(a^o_{ijt}=0;\boldsymbol{\theta}_l)$ actual nonallocation
Static	0	-1,308,021	0.691	0.891
Gittins	0	-1,104,171	0.640	0.910
UCB	1	-544,118	0.854	0.981
FlexGittins	1	-1,056,772	0.640	0.917
FlexGittins+D	2	-349,531	0.644	0.981
UCB+D	2	-206,951	0.909	0.995

Note: Estimation uses data between 2005 and 2015 from JCSG, one of the four large labs in the data. The number of trials actually allocated was 320,295. The number of trials in the choice sets after random sampling is 5,807,902. The rest of the notes of Table 2 apply.

Table D2: Comparison of likelihoods of different models, MCSG

Model	Free parameters in $B_{ijt}(\cdot)$	Log likelihood	Avg $\hat{P}(a^o_{ijt}=1;\boldsymbol{\theta}_l)$ actual allocation	Avg $\hat{P}(a^o_{ijt}=0;\boldsymbol{\theta}_l)$ actual nonallocation
Static	0	-636,089	0.644	0.994
Gittins	0	-473,481	0.669	0.996
UCB	1	-274,230	0.769	0.998
FlexGittins	1	-402,754	0.681	0.997
FlexGittins+D	2	-225,201	0.780	0.998
UCB+D	2	-157,987	0.850	0.999

Note: Estimation uses data between 2005 and 2015 from MCSG, one of the four large labs in the data. The number of trials actually allocated was 141,059. The number of trials in the choice sets after random sampling is 39,136,035. The rest of the notes of Table 2 apply.

Table D3: Comparison of likelihoods of different models, NYSGRC

Model	Free parameters in $B_{ijt}(\cdot)$	Log likelihood	Avg $\hat{P}(a^o_{ijt}=1;\boldsymbol{\theta}_l)$ actual allocation	Avg $\hat{P}(a^o_{ijt}=0;\boldsymbol{\theta}_l)$ actual nonallocation
Static	0	-677,382	0.538	0.990
Gittins	0	-550,827	0.561	0.992
UCB	1	-339,498	0.682	0.996
FlexGittins	1	-529,851	0.565	0.993
FlexGittins+D	2	-400,671	0.626	0.994
UCB+D	2	-288,162	0.700	0.996

Note: Estimation uses data between 2005 and 2015 from NYSGRC, one of the four large labs in the data. The number of trials actually allocated was 139,276. The number of trials in the choice sets after random sampling is 23,883,552. The rest of the notes of Table 2 apply.

Table D4: Out-of-sample fit of UCB+D model

Lab	Sample	Avg Log likelihood	Avg $\hat{P}(a^o_{ijt}=1;\hat{\boldsymbol{\theta}}_l)$ actual allocations	Avg $\hat{P}(a^o_{ijt}=0;\hat{\boldsymbol{\theta}}_l)$ actual nonallocations
JCSG	in	-0.033	0.909	0.994
	out	-0.036	0.914	0.996
MCSG	in	-0.004	0.826	0.999
	out	-0.004	0.871	0.999
NESG	in	-0.004	0.817	0.999
	out	-0.003	0.850	0.999
NYSGRC	in	-0.013	0.656	0.996
	out	-0.011	0.734	0.997

Note: To compute these results, I first estimate the UCB+D model using only observed allocation decisions in odd years. I then compute the in-sample results using the estimates and the odd years' decisions which I used to fit the model. I compute the out-of-sample results using even years' decisions. The rest of the notes of Table 2 apply.

Table D5: Estimates of parameters in different models, NESG

	UCB+D		Static		Gittins		UCB		FlexGittins		FlexGittins+D	
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)				
	2005–2008	2009–2015	2005–2008	2009–2015	2005–2008	2009–2015	2005–2008	2009–2015	2005–2008	2009–2015	2005–2008	2009–2015
θ_{B1}	158.3	119.5	–	–	–	–	210.5	188.8	1.6	1.4	1.4	0.8
	[156.4, 160.1]	[118.1,121.4]					[208.8, 212.5]	[188.5,189.2]	[1.6,1.6]	[1.4,1.4]	[1.4,1.4]	[0.8,0.8]
θ_{B2}	-2.3	-4.7	–	–	–	–	–	–	–	–	-6.5	-8.4
	[-2.3, -2.2]	[-4.7,-4.7]									[-6.4,-6.6]	[-8.4,-8.4]
θ_{quant}	105.6	95.9	288.9	232.6	167.5	162.3	106.1	146.1	150.0	149.5	138.4	143.5
	[105.5, 105.8]	[95.8,95.9]	[286.3, 291.4]	[232.4,232.9]	[167.1, 167.8]	[162.1,162.5]	[106.0, 106.3]	[146.1,146.2]	[149.3,151.0]	[149.4,149.8]	[137.9,139.1]	[143.3,143.7]
θ_{novel}	28.4	-23.3	202.6	20.7	67.6	-8.5	28.7	-39.8	48.4	-14.4	38.7	-7.7
	[27.8, 28.9]	[-23.4,-23.1]	[200.5, 204.3]	[20.5,20.8]	[67.3, 67.8]	[-8.6,-8.4]	[28.6, 28.9]	[-39.8,-39.8]	[47.9,49.1]	[-14.5,-14.3]	[38.1,39.4]	[-7.7,-7.7]
$\theta_{prevStructZ}$	18.1	20.1	9.8	56.5	3.8	7.9	22.3	47.6	3.0	5.2	-0.4	4.9
	[17.6, 18.6]	[20.1,20.3]	[9.4, 10.0]	[56.4,56.6]	[3.8, 3.9]	[7.8,8.0]	[22.1, 22.5]	[47.5,47.7]	[2.7,3.2]	[5.1,5.2]	[-0.7,-0.1]	[4.8,5.0]
θ_{biomed}	21.5	52.9	45.3	158.5	7.3	44.1	21.5	102.5	8.0	38.0	3.4	29.3
	[21.3, 21.7]	[52.7,52.9]	[43.1, 47.0]	[158.3,158.8]	[7.1, 7.5]	[44.0,44.2]	[21.4, 21.7]	[102.4,102.5]	[7.8,8.3]	[37.8,38.4]	[2.8,3.9]	[29.1,29.3]
$\theta_{prevPubZ}$	-9.2	-3.0	-14.5	-1.7	-6.9	-0.5	-9.4	-6.8	-6.9	-4.5	-3.8	-3.1
	[-9.3, -9.0]	[-3.1,-3.0]	[-14.7, -14.2]	[-1.7,-1.7]	[-7.0, -6.8]	[-0.5,-0.4]	[-9.6, -9.2]	[-6.8,-6.7]	[-7.2,-6.7]	[-4.5,-4.4]	[-3.9,-3.6]	[-3.1,-3.1]
θ_{human}	67.1	124.0	105.9	191.0	51.4	90.5	69.8	160.6	40.2	83.0	36.3	74.5
	[66.9, 67.3]	[123.9,124.0]	[104.8, 106.9]	[190.7,191.3]	[51.2, 51.6]	[90.4,90.6]	[69.7, 70.0]	[160.6,160.7]	[40.1,40.5]	[82.9,83.1]	[36.0,36.7]	[74.5,74.6]
$\theta_{eukaryote}$	-29.8	-35.9	67.7	61.6	-8.3	-26.9	-30.0	-0.3	-14.3	-33.7	-11.7	-33.4
	[-29.9, -29.7]	[-36.0,-35.8]	[67.1, 69.1]	[61.5,61.8]	[-8.7, -8.0]	[-27.0,-26.8]	[-30.4, -29.8]	[-0.3,-0.2]	[-14.4,-13.9]	[-33.7,-33.6]	[-12.0,-11.6]	[-33.4,-33.4]
$\theta_{membrane}$	58.3	13.1	96.5	52.3	35.2	28.4	58.1	22.7	31.2	24.8	23.5	22.1
	[58.0, 58.6]	[13.1,13.3]	[96.1, 97.2]	[52.1,52.4]	[35.1, 35.4]	[28.3,28.5]	[57.9, 58.5]	[22.7,22.8]	[31.1,31.4]	[24.8,24.9]	[23.1,23.9]	[22.1,22.2]

Note: Table displays the full estimates of parameters in different models for NESG, one of the four large labs. Results from other labs are available upon request. $prevStructZ_{iy}$ represents the number of standard deviations by which $prevStruct_{iy}$ differs from the yearly mean $\overline{prevStruct}_y$. $prevPubZ_{iy}$ represents the number of standard deviations by which $prevPub_{iy}$ differs from the yearly mean $\overline{prevPub}_y$. The rest of the notes of Table 3 apply.

Table D6: Estimates of main parameters of interest in UCB+D model, other labs

	JCSG		MCSG		NYSGRC	
	2005–2008	2009–2015	2005–2008	2009–2015	2005–2008	2009–2015
Parameter	(1)	(2)	(3)	(4)	(5)	(6)
θ_{B1}	558.2	1001.9	273.3	127.1	61.6	115.6
	[551.1,573.4]	[977.6,1028.0]	[266.2,284.8]	[126.3,127.8]	[61.2,61.9]	[114.2,116.9]
θ_{B2}	-247.9	-104.7	-3.9	-3.8	-2.9	-3.9
	[-247.6, -248.5]	[-102.9, -106.6]	[-3.9,-3.9]	[-3.8,-3.8]	[-2.9,-3.0]	[-3.9,-3.9]
θ_{biomed}	-34.3	105.1	12.7	84.8	33.0	89.4
	[-34.6,-34.1]	[104.7,105.8]	[12.6,12.7]	[84.8,84.8]	[32.9,33.1]	[89.1,89.8]

Note: Table displays the estimates of the main parameters of interest in the UCB+D model for the other three large labs. Full estimates of parameters in different models are available upon request. 95% confidence intervals are computed using the MCMC approach in Chernozhukov & Hong (2003) and are shown in brackets. These confidence intervals are almost identical to those computed using Procedure 1 of Chen et al. (2018).

Table D7: Simulated outcomes of UCB+D model, other labs

Lab	Model	Projects attempted	Unique publications	Citations	Downloads (millions)
JCSG	UCB+D	40,881	1,607	1,495	38.4
	Actual	40,881	1,512	1,463	36.3†
MCSG	UCB+D	77,503	2,040	2,524	46.3
	Actual	78,740	2,276	3,145	50.0†
NYSGRC*	UCB+D	59,734	626	2,579	14.7
	Actual	59,734	617	2,575	14.4†

Note: Each simulation uses $\hat{\boldsymbol{\theta}}_l$ from parameter estimates of the corresponding model. Results are averaged from three simulations of each model. †The download data on actually published projects are between 2007 and 2013, so actual five-year downloads may not be available for some projects. I predict five-year downloads for the actually published projects using the predictive model described in Appendix B.3. *I note data problems related to NYSGRC. For this particular lab, more than half of the trials that produced structures either miss key stage dates or have those dates in wrong orders (for example, publication is at an earlier date than previous stages). As a result, I am not able to correctly simulate the dates and outcomes of different stages of trials. Though this set of simulation results looks quite nice, please take them with a grain of salt.

Table D8: Out-of-sample simulation results for UCB+D model

Lab	Model	Projects attempted	Unique publications	Citations	Downloads (millions)
JCSG	in	40,881	1,607	1,495	38.4
	out	40,881	1,621	1,481	38.6
	actual	40,881	1,512	1,463	36.3†
MCSG	in	77,503	2,040	2,524	46.3
	out	77,504	2,051	2,553	46.4
	actual	78,740	2,276	3,145	50.0†
NESG	in	59,947	1,097	3,376	25.6
	out	59,913	1,085	3,336	25.3
	actual	59,953	1,053	3,502	24.5†
NYSGRC*	in	59,734	626	2,579	14.7
	out	59,734	628	2,594	14.7
	actual	59,734	617	2,575	14.4†

Note: In-sample simulation results are identical to those for the UCB+D model in Tables 4 and D7. I simulate the out-of-sample results as follows. I first fit the UCB+D model for each lab with odd years of observed decisions. Using those estimated parameters, I simulate each lab's full input allocation history and output, in odd and even years. The rest of the notes of Table D7 apply.

Table D9: Counterfactual outcomes, no exploration, other labs

Lab	Counterfactual model	Projects attempted	Unique publications	Citations	Downloads (millions)
JCSG	Static	10,710	1,338	1,270	32.4
		(-74%)	(−17%)	(−15%)	(−16%)
	Baseline model	40,881	1,607	1,495	38.4
MCSG	Static	14,883	668	856	15.6
		(-81%)	(−67%)	(−66%)	(−66%)
	Baseline model	77,503	2,040	2,524	46.3
NYSGRC*	Static	5,203	247	1,026	5.9
		(-91%)	(−61%)	(−60%)	(−60%)
	Baseline model	59,734	626	2,579	14.7

Note: Each simulation uses $\hat{\boldsymbol{\theta}}_{Xl}$ from parameter estimates of the UCB+D model for the corresponding lab. These estimates are available upon request. Results are averaged from three simulations of the model. Output from the baseline model are identical to those for the UCB+D model in Appendix Table D7. Parentheses show percentage differences as compared to the baseline model. Table shows results from the three other large labs. See Table 5 for results from NESG. *I note data problems related to NYSGRC. For this particular lab, more than half of the trials that produced structures either miss key stage dates or have those dates in wrong orders (for example, publication is at an earlier date than previous stages). As a result, I am not able to correctly simulate the dates and outcomes of different stages of trials. Though this set of simulation results looks quite nice, please take them with a grain of salt.