Regulating Artificial Intelligence

Joao Guerreiro* Sergio Rebelo† Pedro Teles‡

November 19, 2023

Abstract

We consider an environment in which there is substantial uncertainty about the potential negative external effects of AI algorithms. We find that subjecting algorithm implementation to regulatory approval or alternatively holding developers accountable for adverse external impacts of their algorithms is insufficient to implement the social optimum. When testing costs are low, a combination of mandatory beta testing for external effects and making developers liable for the negative external effects of their algorithms implements the social optimum even when developers have limited liability.

J.E.L. Classification: H21, O33
Keywords: AI, regulation

*UCLA.
†Northwestern University, NBER, and CEPR.
‡Banco de Portugal, Catolica-Lisbon School of Business & Economics, and CEPR.

1 Introduction

In 1950, Isaac Asimov published *I, Robot*, a collection of short stories about the dilemmas of a world where robots powered by artificial intelligence (AI) interact with humans. Recent advances in AI have brought these dilemmas from the realm of science fiction to the pages of newspapers and the halls of parliaments. In this paper, we discuss the efficacy of different approaches to AI regulation.

Over the past decade, the declining costs of computing power and the availability of vast data sets allowed neural networks and other forms of AI to accomplish remarkable feats. Reinforcement learning algorithms beat humans at games of perfect information, like chess and go (Silver et al., 2017, Silver et al., 2018). AI algorithms outperformed humans in games of imperfect information, such as poker (Brown and Sandholm, 2019). Convolutional neural networks achieved remarkable accuracy in image recognition tasks (Langlotz et al., 2019). Natural language processing models like Generative Pre-trained Transformers have made significant strides in language understanding, translation, and content generation (Eloundou, Manning, Mishkin, and Rock, 2023). AI algorithms in general have improved prediction accuracy in many domains relevant to business applications (Agrawal, Gans, and Goldfarb, 2022).

These and other breakthroughs hold the promise of delivering significant benefits to society. However, they also carry the risk of imposing considerable societal costs. These costs include negative externalities, such as fueling political polarization, facilitating fraud, disseminating false information, jeopardizing financial stability, and weakening democracies (Beraja, Kao, Yang, and Yuchtman, 2023). Other costs take the form of "internalities," situations in which individuals are manipulated to act against their self-interest through misinformation or exploitation of self-control and time inconsistency problems.

In May 2023, a consortium of prominent figures in the field of AI signed a statement declaring that "Addressing the existential risks posed by AI should be a global priority, on par with other worldwide challenges like pandemics and nuclear warfare." The leaders of the G7 nations initiated the Hiroshima AI Process to harmonize AI regulation.

Europe and the United States have started to design regulatory frameworks to address the challenges posed by AI (see European Commission, 2020, Benifei and Tudorache, 2023, and Biden, 2023). Ideas proposed so far include mandatory testing of AI algorithms and holding AI developers accountable for the adverse outcomes resulting from the use of their technology. Policymakers are also considering classifying AI technologies into risk tiers (unacceptable, high, limited risk, and minimal), forbidding the development of algorithms that create unacceptable risks (European Commission, 2022).

We assess these ideas using a model designed to capture the key aspect of ongoing developments in AI: there is substantial uncertainty about the resulting societal costs and benefits. Our analysis is normative; we evaluate the impact on social welfare of different regulatory frameworks.

The impact of negative externalities and internalities is broadly similar. To simplify the exposition, we focus our discussion on the potential negative externalities generated by AI algorithms.

We explore two settings. In both settings, an AI developer makes decisions regarding the novelty of their AI algorithm relative to the state of the art. There is ex-ante uncertainty about the negative externalities that this algorithm might cause. This uncertainty grows with the distance between the new algorithm's approach and the status quo.

In the first setting, uncertainty is not resolved until an AI algorithm is fully implemented, and this implementation is irreversible. Potential negative externalities drive a wedge between the social optimum and the unregulated equilibrium. The planner wants to be more cautious than private markets–the optimal level of AI novelty for society is lower than what naturally emerges in an unregulated setting.

In the second setting, uncertainty regarding potential negative externalities can be resolved through experimentation, which we call beta testing. This testing involves making the algorithm available to a small group of households and use the test results to decide whether to make the algorithm available to the population as a whole. Developers regularly engage in beta testing to assess how effective their algorithm is from a user's perspective. The beta testing we emphasize in this paper serves a distinct purpose: to measure an algorithm's external effects.

In the unregulated equilibrium, the developer has weaker incentives for beta testing than the planner. Once again, the planner exhibits greater caution than private markets.

We show three results. First, subjecting algorithm release to regulatory approval is insufficient to implement the social optimum–developers still have an incentive to create algorithms that are too risky. Second, simply holding developers accountable for any adverse external impacts of their algorithms implements the social optimum if developers are not protected by limited liability. However, this policy is also insufficient to implement the social optimum if developers are protected by limited liability. Third, we find that, when testing costs are low, a combination of mandatory beta testing for externalities and making developers liable for the negative external effects of their algorithms implements the social optimum. This regulatory solution implements the social optimum even when there is limited liability. One advantage of this solution is developers do not need to seek regulatory approval before implementing their algorithms.

Our paper is related to four important strands of literature. The first studies the value of experimentation (e.g., Callander, 2011 and Ilut and Valchev, 2023). The second analyses settings that are relevant to the design and execution of clinical trials. These situations feature multiple options and unknown rewards, commonly known as the multi-armed bandit problem (e.g., Thompson, 1933 and Gittins, 1974). The third considers the importance of data as an input into AI algorithms (e.g., Jones and Tonetti, 2020 and Farboodi and Veldkamp, 2021). The fourth researches the impact of AI on the economy (e.g., Burstein, Morales, and Vogel, 2019, Acemoglu and Restrepo, 2022, and Jones, 2023).

In Section 2, we study the model without beta-testing. We introduce beta-testing in Section 3. In Section 4, we evaluate different regulatory proposals. Section 5 concludes.

2 Model without beta testing

This section considers a model in which an AI algorithm cannot be tested before it is released and in which the release is irreversible. We discuss the household problem, the problem of the AI developer, and the unregulated equilibrium. Then, we characterize the social optimum and compare it with the unregulated equilibrium.

2.1 Unregulated equilibrium

Household problem The economy has a continuum of households indexed by $i \in [0, N]$, where N denotes the number of households in the population. Each household has a constant exogenous income level denoted by y. Households decide whether to purchase a license to use an AI algorithm with novelty ℓ at a price p. Their utility, $\mathcal{U}_i$, has a quasi-linear form:

$$\mathcal{U}_i \equiv y + \{u(\ell)\,\mu - p\} \times \mathcal{I}_i - \mathbb{E}[e^2]. \tag{1}$$

The indicator function $\mathcal{I}_i$ takes the value one if household i buys the AI license and zero otherwise. The utility derived from using the AI algorithm is $u(\ell)\,\mu$. To capture positive network externalities, we assume that this utility is proportional to the number of users $\mu = \int \mathcal{I}_i di$.

We assume that the function u is increasing, $u' > 0$, and concave $u'' < 0$ and that the Inada condition $\lim_{\ell \downarrow 0} u'(\ell) = \infty$ holds. We also normalize $u(0) = 0$.

AI usage causes a negative externality e that reduces utility by e^2. We assume that the externality is proportional to the measure of users and takes the form:

$$e = \phi\left(\ell\right) \times \mu.$$

For each value of ℓ, $\phi(\ell)$ is a random variable. We assume that the distribution $\phi(\ell)$ satisfies two properties. First, it is symmetric around zero, so that the expected externality is zero:

$$\mathbb{E}[\phi(\ell)] = 0.$$

Second, the uncertainty about potential AI externalities is an increasing function of the novelty level ℓ. We let $\sigma^2(\ell)$ denote the uncertainty about potential AI externalities for an algorithm with novelty level ℓ:

$$\sigma^2\left(\ell\right) \equiv \mathbb{E}\left[\phi(\ell)^2\right].$$

We assume that $\sigma^2(\ell)$ is increasing and convex in ℓ, and $\sigma\left(0\right) = 0$, i.e., there is no uncertainty in the status quo.

Replacing e in equation (1), we obtain,

$$\mathcal{U}_i \equiv y + \{u(\ell)\mu - p\} \times \mathcal{I}_i - \sigma^2(\ell)\mu^2.$$

Households purchase a license to use the AI algorithm whenever private benefits exceed the price,

$$u\left(\ell\right)\mu \geq p.$$

The AI developer's problem We consider a single AI developer who chooses ℓ, the algorithm's novelty, the license price, p, and the number of available licenses, μ.

The cost of developing an algorithm with novelty ℓ is $f(\ell)$. This cost is increasing and convex in ℓ and $f(0) = 0$.

The developer experiences disutility from the externality in the same way that households do. However, the developer does not take into account the external effects endured by the households. The utility of the developer is:

$$\mathcal{V} \equiv \begin{cases} \mu p - \sigma^2(\ell)\,\mu^2 - f(\ell) & \text{if } p \leq u(\ell)\,\mu, \\ 0 - f(\ell) & \text{if } p > u(\ell)\,\mu. \end{cases}$$

If the developer markets the AI algorithm, the optimal license price is $p = u(\ell)\,\mu$. The developer uses its monopoly position to capture the entire consumer surplus. This pricing strategy does not generate deadweight losses; it simply redistributes resources from the households to the monopolists.

The optimal levels of ℓ and μ solve the following problem:

$$\max_{\mu,\ell\geq 0,\mu\leq N} u(\ell)\,\mu^2 - f(\ell) - \sigma^2(\ell)\,\mu^2.$$

We characterize the solution to this problem in two steps. First, taking ℓ as given, we ask how many licences, μ, should be made available at a price p. Second, we consider the optimal choice of ℓ from the developer's standpoint.

Given ℓ, the optimal μ depends on the sign of $u(\ell) - \sigma^2(\ell)$. If this expression is positive, it is optimal to make the algorithm available to the whole population ($\mu = N$). Otherwise, the algorithm is not released ($\mu = 0$).

The utility of the AI developer is:

$$\mathcal{V}(\ell) = \begin{cases} \left\{u(\ell) - \sigma^2(\ell)\right\} N^2 - f(\ell) & \text{if } u(\ell) - \sigma^2(\ell) \geq 0, \\ -f(\ell) & \text{if } u(\ell) - \sigma^2(\ell) < 0. \end{cases}$$

If $[u(\ell) - \sigma^2(\ell)]N^2 < f(\ell)$ for all ℓ, the developer does not produce any algorithm.

When the solution for ℓ is interior, it satisfies the first-order condition:

$$\left\{u'(\ell) - \frac{\partial \sigma^2(\ell)}{\partial \ell}\right\} N^2 - f'(\ell) = 0.$$

Unregulated equilibrium We now describe the characteristics of an equilibrium without regulation in which ℓ has an interior value. The superscript e denotes the values of various variables in this equilibrium. These variables satisfy the following conditions:

$$\mu^e = N,$$

$$\left\{u'\left(\ell^e\right) - \frac{\partial\sigma^2\left(\ell^e\right)}{\partial\ell}\right\}N^2 - f'\left(\ell^e\right) = 0,$$

and ℓ^e is such that the utility of the developer is positive,

$$\mathcal{V}\left(\ell^e\right) = \left\{u\left(\ell^e\right) - \sigma^2\left(\ell^e\right)\right\}N^2 - f\left(\ell^e\right) \geq 0.$$

2.2 The planner's problem

The total household welfare in an economy in which μ households use the AI algorithm is

$$\int_0^N \mathcal{U}_i di = Ny + \{u(\ell)\mu - p\}\,\mu - N\sigma^2(\ell)\mu^2.$$

Social welfare is the sum of the households' and developer's utilities:

$$\mathcal{W} = \int_0^N \mathcal{U}_i di + \mathcal{V} = Ny + \left\{u\left(\ell\right) - \left(N+1\right)\sigma^2\left(\ell\right)\right\}\mu^2 - f\left(\ell\right).$$

With quasi-linear utility, we can think of this social welfare function as maximizing the total surplus in the economy.

The following proposition compares the social optimum with the unregulated equilibrium.

Proposition 1 (Conservatism in algorithm release)**.** *The social planner is more conservative than the developer when it comes to releasing algorithms. The social planner releases an algorithm with novelty ℓ to the entire population ($\mu = N$) if*

$$\left\{u\left(\ell\right) - (N+1)\sigma^2\left(\ell\right)\right\}N^2 - f\left(\ell\right) \geq 0.$$

In contrast, the developer releases an algorithm of novelty ℓ to the whole population ($\mu = N$) if

$$\left\{u(\ell) - \sigma^2(\ell)\right\} N^2 - f(\ell) \geq 0.$$

The novelty level that the planner is willing to release is lower than the novelty that the developer chooses to release.

We now compare the social optimum novelty level, ℓ, with the unregulated equilibrium.

Proposition 2 (Conservatism in algorithm development)**.** *The social optimum is more conservative than the unregulated equilibrium. The socially desirable novelty level, ℓ^*, is lower than the level that emerges in the unregulated equilibrium, ℓ^e.*

In the appendix, we provide a proof of this proposition using monotone comparative statics. Below, we sketch a proof for the case in which the solution is interior. The socially optimal solution satisfies the first-order condition:

$$0 = \left\{u'(\ell^*) - (N+1)\frac{\partial \sigma^2(\ell^*)}{\partial \ell}\right\} N^2 - f'(\ell^*).$$

The optimal condition for the developer evaluated at the socially optimal ℓ^* is

$$0 < \left\{u'(\ell^*) - \frac{\partial \sigma^2(\ell^*)}{\partial \ell}\right\} N^2 - f'(\ell^*),$$

so

$$\left\{u'(\ell^*) - \frac{\partial \sigma^2(\ell^*)}{\partial \ell}\right\} N^2 - f'(\ell^*) > \left\{u'(\ell^e) - \frac{\partial \sigma^2(\ell^e)}{\partial \ell}\right\} N^2 - f'(\ell^e) = 0,$$

and therefore

$$\ell^* < \ell^e.$$

The key driver of this result is that when selecting ℓ the developer disregards the external effects on the rest of society.

Regulating AI

One regulatory approach to align the decisions of AI developers with the societal interests is to impose an upper bound on the degree of novelty, ℓ, that developers can implement. This method resembles the European Commission's proposal of classifying AI algorithms into risk tiers (unacceptable, high, limited, and minimal) and forbiding the development of algorithms with unacceptable risks (European Commission, 2022).

In the model we have been considering, in which there is no beta testing, setting the upper bound for ℓ equal to the socially optimal novelty level is sufficient to implement the first best. As we show in the next section, this result no longer holds in the model with beta testing, because incentives to test and implement algorithms are not aligned by simply placing an upper bound on ℓ.

Another regulatory approach is to hold AI developers liable for the external costs of the AI algorithm. As we discuss in Section 4, without limits to liability, this regulation is sufficient to implement the social optimum in settings with and without beta testing. However, with limited liability, the policy is no longer sufficient to align incentives because AI developers do not fully internalize the external consequences of the AI algorithm when $\phi(\ell)$ is very large. In Section 4, we discuss a combination of limited liability and mandatory beta testing that implements the social optimum.

3 Model with beta testing

This section considers a two-period version of the previous model. Time is discrete and indexed by $t = 1, 2$. The developer can test the algorithm in the first period in a sample of μ_1 users to evaluate the externalities. Based on the outcomes of this test, they can then decide whether to release the algorithm in the second period. For simplicity, we consider the case in which the external effects, $\phi(\ell)$, are perfectly revealed after the AI algorithm is tested in the first period.

People assign weights $1-\beta$ and β to the utility of the first and second period, respectively. The model without beta testing is a particular case of this more general model where $\beta = 0$.

As β converges to one, the weight of the first period's utility falls, and the cost of testing in the first period, which is the opportunity cost of not deploying the algorithm in the first period, becomes negligible.

As in the previous section, we begin by describing the unregulated equilibrium. We then compute the social optimum and compare it to the unregulated equilibrium.

3.1 Unregulated equilibrium

Household's problem The household lives for two periods. Their utility is

$$\mathcal{U}_i = (1-\beta)\left[y + \{u(\ell)\mu_1 - p_1\}\mathcal{I}_{1,i} - \mathbb{E}[e_1^2]\right] + \beta\mathbb{E}\left[y + \{u(\ell)\mu_2 - p_2\}\mathcal{I}_{2,i} - e_2^2\right].$$

The household purchases an AI license in period t as long as the private benefits exceed the price

$$p_t \leq u(\ell)\mu_t.$$

AI developer's problem In period one, the AI developer makes three decisions: which novelty level to develop (ℓ), how many AI licenses to offer for sale (μ_1), and what price to charge for each license (p_1).

If μ_1 equals zero, the developer obtains no information about the external effects of the AI algorithm in period two and the decision-making process resembles that of the model without beta testing.

If μ_1 is greater than zero, the developer obtains information about the external effects of the AI algorithm in period two. Using this information, the developer chooses μ_2, the number of AI licenses to offer for sale in period two, and p_2, the price per license.

The utility of the developer in period two is,

$$\mathcal{V}_2 = \begin{cases} \mu_2 p_2 - \phi(\ell)^2 \mu_2^2, & \text{if } p_2 \leq u(\ell)\mu_2 \quad \text{and} \quad \mu_1 > 0, \\ \mu_2 p_2 - \sigma^2(\ell)\mu_2^2, & \text{if } p_2 \leq u(\ell)\mu_2 \quad \text{and} \quad \mu_1 = 0, \\ 0, & \text{if } p_2 > u(\ell)\mu_2. \end{cases}$$

As before, the price that maximizes the utility of the developer is $p_2 = u(\ell)\mu_2$.

If $\mu_1 > 0$, then $\mu_2 = N$ if $u(\ell) - \phi(\ell)^2 \geq 0$ and $\mu_2 = 0$ otherwise. If $\mu_1 = 0$, then $\mu_2 = N$ if $u(\ell) - \sigma^2(\ell) \geq 0$ and $\mu_2 = 0$ otherwise.

To make the problem interesting, we assume that $\phi(\ell)$ is such that there is a strictly positive probability that both $u(\ell) - \phi(\ell)^2 > 0$ and $u(\ell) - \phi(\ell)^2 < 0$. This assumption means that the probability that the AI algorithm is implemented in period two, given the information obtained in period one, is strictly positive but less than one.

The optimized developer utility in period two, $\mathcal{V}_2^*(\ell, \mu_1)$ is,

$$\mathcal{V}_2^*(\ell, \mu_1) = \max \left\{ u(\ell) - \phi(\ell)^2 \mathcal{I}(\mu_1 > 0) - \sigma^2(\ell)(1 - \mathcal{I}(\mu_1 > 0)), 0 \right\} N^2,$$

where $\mathcal{I}(\mu_1 > 0) = 1$ if $\mu_1 > 0$ and zero otherwise.

Lemma 1. *Expected utility in the second period is higher when there is beta testing in the first period,*

$$\mathbb{E}[\mathcal{V}_2^*(\ell, \mu_1)] > \mathcal{V}_2^*(\ell, 0), \quad if \ \mu_1 > 0.$$

Proof. Let μ_2^0 denote the optimal choice of μ_2 when $\mu_1 = 0$. Note that μ_2^0 is necessarily non-state contingent, so $\mu_2^0 = N$ or $\mu_2^0 = 0$ depending on the degree of uncertainty regarding the externality. Then, if $\mu_1 > 0$:

$$\begin{aligned} \mathbb{E}[\mathcal{V}_2^*(\ell, \mu_1)] &= \mathbb{E}\left[\max\left\{\left(u(\ell) - \phi(\ell)^2\right) N^2, 0\right\}\right] \\ &> \mathbb{E}\left[\left(u(\ell) - \phi(\ell)^2\right) (\mu_2^0)^2\right] = \left(u(\ell) - \sigma^2(\ell)\right) (\mu_2^0)^2 = \mathcal{V}_2^*(\ell, 0). \end{aligned}$$

□

The problem in period one is to choose ℓ, μ_1 and p_1 to maximize

$$\mathcal{V} = (1-\beta)\left(\begin{cases} \mu_1 p_1 \mu_1^2 - \sigma^2(\ell)\mu_1^2, & \text{if } p_1 \leq u(\ell)\mu_1 \\ 0, & \text{if } p_1 > u(\ell)\mu_1 \end{cases}\right) + \beta\mathbb{E}[\mathcal{V}_2^*(\ell,\mu_1)] - f(\ell).$$

The optimal price for the developer is $p = u(\ell)\mu_1$.

From the standpoint of period one, it is still optimal to set $\mu_1 = N$ if $u(\ell) - \sigma^2(\ell) \geq 0$ and $\mu_1 = 0$ if $u(\ell) - \sigma^2(\ell) < 0$. However, experimenting in the first period, $\mu_1 > 0$, creates value by generating information that the developer can use in the second period.

Given the discontinuity in information generation from $\mu_1 = 0$ to $\mu_1 > 0$, the problem may have a supremum but not a maximum. For a given ℓ, if $u(\ell) - \sigma^2(\ell) < 0$ then the static optimal decision would be $\mu_1 = 0$. However, choosing an infinitesimal, positive value of μ_1 yields strictly larger utility than setting μ_1 to zero. Therefore, the optimal number of households trying the technology in period one should be strictly positive but kept as low as possible ($\mu_1 \downarrow 0$). We refer to this setting as the *experimentation solution*: the developer sells AI licenses to an infinitesimal fraction of households to test the product and then decides whether to sell the product given the information revealed in period two.[1]

Proposition 3 (Uncertainty, beta testing, and product release)**.** *In an unregulated equilibrium, the number of user licenses (μ_1^e) offered by the developer in the first period depends on the level of uncertainty as follows:*

1. *The developer does beta testing ($\mu_1^e \downarrow 0$) when the degree of uncertainty is high*

$$\sigma^2(\ell) > u(\ell).$$

[1] We could extend the model to the case in which the information revealed is an increasing function of the number of households involved in beta testing. In this extension, μ_1 would still be positive, but the model is more complex.

2. *The developer foregoes beta testing and releases the AI algorithm to the entire population in the first period ($\mu_1^e = N$) when uncertainty is low*

$$\sigma^2(\ell) \leq u(\ell).$$

In both scenarios, the developer learns the external effects of the AI algorithm in the second period and then:

1. *Withdraws the product from the market ($\mu_2^e = 0$) if the personal cost to the developer arising from externalities is substantial,*

$$\phi(\ell)^2 > u(\ell).$$

2. *Makes the product available to the whole population ($\mu_2^e = N$) if the personal cost to the developer arising from externalities is relatively minor*

$$\phi(\ell)^2 \leq u(\ell).$$

Since μ_1 is always positive, then

$$\mathcal{V}_2^*(\ell, \mu_1^e) = \max \left\{ u(\ell) - \phi(\ell)^2, 0 \right\} N^2,$$

and so

$$\mathcal{V} = (1 - \beta) \max \left\{ u(\ell) - \sigma^2(\ell), 0 \right\} N^2 + \beta \mathbb{E}[\mathcal{V}_2^*(\ell, \mu_1^e)] - f(\ell).$$

3.2 The planner's problem

We consider a central planner who can decide, in the first period, both the novelty of the AI algorithm developed and the number of households that can access it. If the AI algorithm is implemented in the first period, the planner learns its external effects. In the second period, the planner decides whether to make the AI algorithm available and how many licenses to offer.

As in the model without beta testing, we compute the allocations that maximize the social surplus $\int_0^1 \mathcal{U}_i di + \mathcal{V}$. With quasi-linear utility, this problem is equivalent to maximizing efficiency. Any distribution of utilities can be achieved using lump-sum transfers.

We begin by describing the solution to the second-period problem, contingent upon the choices made in the first period about ℓ and μ_1.

Social problem, second period Development costs are incurred in period one. If $\mu_1 > 0$, then the planner learns the external effects of the AI algorithm, $\phi(\ell)$. If $\mu_1 = 0$, then the planner faces the same uncertainty about the AI algorithm's potential externalities, $\mathbb{E}[\phi(\ell)^2] = \sigma^2(\ell)$ as in the model without beta testing.

The expected social welfare in the second period, considering the available information, is given by:

$$\mathcal{W}_2 = \begin{cases} Ny + \left\{u(\ell) - (N+1)\,\phi(\ell)^2\right\}\mu_2^2 & \text{if } \mu_1 > 0, \\ Ny + \left\{u(\ell) - (N+1)\,\sigma^2(\ell)\right\}\mu_2^2 & \text{if } \mu_1 = 0. \end{cases}$$

Let's determine the optimal μ_2. If $\mu_1 > 0$, it's optimal to make the algorithm available to the entire population, $\mu_2 = N$, if $u(\ell) - (N+1)\phi(\ell)^2 \geq 0$ and to not release the AI algorithm otherwise ($\mu_2 = 0$). If $\mu_1 = 0$, then $\mu_2 = N$ if $u(\ell) - (N+1)\sigma^2(\ell) \geq 0$ and $\mu_2 = 0$ otherwise.

The planner only releases AI algorithms that are socially beneficial, taking into account the external effects on the entire population, $(N+1)\phi(\ell)^2$. In contrast, the developer considers only its own loss of utility due to external effects, $\phi(\ell)^2$. This difference implies that the developer is willing to commercialize AI algorithms that are detrimental to society.

Proposition 4 (Optimal restrictions on product release)**.** *The central planner does not release AI algorithms that would be commercialized in an unregulated equilibrium under two circumstances:*

1. *If $\mu_1 > 0$, the external effects on the population are larger than the social benefits of implementing the AI algorithm, but the private benefits to the developer of implementing the algorithm are positive:*

$$\frac{u(\ell)}{N+1} < \phi(\ell)^2 \leq u(\ell).$$

2. *If $\mu_1 = 0$, the social expected benefits of implementing the AI algorithm are negative, while the private expected benefits to the developer of implementing the algorithm are positive:*

$$\frac{u(\ell)}{N+1} < \sigma^2(\ell) \leq u(\ell).$$

The resulting social welfare in period two is given by:

$$\mathcal{W}_2^*(\ell, \mu_1) \equiv \max\left\{u(\ell) - (N+1)\left[\phi(\ell)^2\mathcal{I}(\mu_1 > 0) + \sigma^2(\ell)(1 - \mathcal{I}(\mu_1 > 0))\right], 0\right\} N^2.$$

As before, we assume that $\phi(\ell)$ is such that there is a strictly positive probability that both $u(\ell) - (N+1)\phi(\ell)^2 > 0$ and $u(\ell) - (N+1)\phi(\ell)^2 < 0$. This assumption means that the probability that the AI algorithm is implemented in period two, given the information obtained in period one, is strictly positive but less than one.

Lemma 2 (Benefits of beta testing in period one)**.** *Expected social welfare is higher in the second period when there is beta testing in the first period:*

$$\mathbb{E}[\mathcal{W}_2^*(\ell, \mu_1)] > \mathcal{W}_2^*(\ell, 0), \text{ if } \mu_1 > 0.$$

Proof. Let μ_2^0 denote the optimal choice of μ_2 when $\mu_1 = 0$. Note that μ_2^0 is not state-contingent, it's either $\mu_2^0 = N$ or $\mu_2^0 = 0$, depending on the uncertainty regarding external effects. Then, if $\mu_1 > 0$:

$$\begin{aligned}
\mathbb{E}[\mathcal{W}_2^*(\ell, \mu_1)] &= \mathbb{E}\left[\max\left\{\left(u(\ell) - (N+1)\phi(\ell)^2\right) N^2, 0\right\}\right] \\
&> \mathbb{E}\left[\left(u(\ell) - (N+1)\phi(\ell)^2\right)(\mu_2^0)^2\right] = \left(u(\ell) - (N+1)\sigma^2(\ell)\right)(\mu_2^0)^2 \\
&= \mathcal{W}_2^*(\ell, 0).
\end{aligned}$$

□

Social problem, first period The overall expected social welfare is given by

$$\mathcal{W} \equiv (1-\beta)\left[Ny + \left\{u(\ell) - (N+1)\sigma^2(\ell)\right\}\mu_1^2\right] + \beta\mathbb{E}[\mathcal{W}_2^*(\ell,\mu_1)] - f(\ell).$$

From the standpoint of period one, it is optimal to set $\mu_1 = N$ if $u(\ell) - (N+1)\sigma^2(\ell) \geq 0$ and $\mu_1 = 0$ if $u(\ell) - (N+1)\sigma^2(\ell) < 0$. However, beta testing in the first period generates information that is valuable in the second period, i.e., if $\mu_1 > 0$

$$\mathbb{E}[\mathcal{W}_2^*(\ell,\mu_1)] > \mathcal{W}_2^*(\ell,0).$$

Just like in the unregulated equilibrium, the discontinuity in information generation from $\mu_1 = 0$ to $\mu_1 > 0$, implies that the problem may have a supremum but not a maximum. As before, we consider an *experimentation solution*: the planner makes AI licenses available to an infinitesimal fraction of households to test the product and then decides whether to release the product given the information revealed in period two.

Proposition 5 (Optimal implementation of an AI algorithm in periods one and two)**.** *Given ℓ, the social optimum number of users in period one (μ_1^*) involves:*

1. *Beta testing ($\mu_1^* \downarrow 0$) if uncertainty is sufficiently large*

$$\sigma^2(\ell) > \frac{u(\ell)}{N+1}.$$

2. *Immediate implementation ($\mu_1^* = N$) if uncertainty is sufficiently small*

$$\sigma^2(\ell) \leq \frac{u(\ell)}{N+1}.$$

In both cases, upon learning the externality consequences of the AI, the planner:

1. *Does not implement the AI algorithm ($\mu_2^* = 0$) if the externalities are sufficiently large*

$$\phi(\ell)^2 > \frac{u(\ell)}{N+1}.$$

2. *Implements the AI algorithm ($\mu_2^* = N$) if the externalities are sufficiently small*

$$\phi(\ell)^2 \leq \frac{u(\ell)}{N+1}.$$

The planner either implements beta testing or releases the algorithm to the population in period one. However, the planner always adopts a more cautious stance than the developer when deciding whether to beta test rather than make the algorithm available to the whole population. There are AI novelty levels for which the developer prefers an immediate release to the general public, while the planner opts for beta testing.

Upon learning in period two the external effects of the AI algorithm, there are algorithms that the developer would find privately beneficial to continue commercializing in the second period that the planner withdraws from the market. Both of these observations stem from the fact that the planner considers the externalities affecting the entire population, while the developer is only concerned with the impact of the external effect on its own utility.

In summary, because the planner considers the impact of externalities on the entire population, it takes a more cautious approach to beta testing in the first period and commercializing the AI algorithm in the second period. We summarize these findings in the following corollary.

Proposition 6 (Caution in testing and implementation)**.** *Fix ℓ. In period one:*

1. *If uncertainty is substantial, $\sigma^2(\ell) \geq u(\ell)$, both the planner and the developer agree to beta test.*

2. *If uncertainty is moderate, $\frac{u(\ell)}{N+1} < \sigma^2(\ell) < u(\ell)$, the planner and the developer disagree. It is optimal for the planner to do beta testing but the developer finds full-scale implementation without testing privately optimal.*

3. *If uncertainty is low, $\sigma^2(\ell) \leq \frac{u(\ell)}{N+1}$, both the planner and the developer agree to release the algorithm to the entire population without beta testing.*

In period two:

1. *If externalities are substantial,* $\phi(\ell)^2 \geq u(\ell)$*, both the planner and the developer agree to withdraw the algorithm from the market.*

2. *If externalities are moderate,* $\frac{u(\ell)}{N+1} < \phi(\ell)^2 < u(\ell)$*, the planner wants to withdraw the product from the market but the developer does not.*

3. *If externalities are low,* $\phi(\ell)^2 \leq \frac{u(\ell)}{N+1}$*, both the planner and the developer agree to release the algorithm to the entire population.*

This proposition means that the planner is more cautious than the developer in the sense that it implements beta testing for externalities more often than the developer. The planner is also more conservative in implementing the algorithm in the second period.

Surprisingly, in contrast with the model without beta testing, the first best can feature a higher novelty level, ℓ, than the unregulated equilibrium. In the model with beta testing, the planner can be cautious in two ways. The first is by choosing a lower, less risky, novelty level ℓ. The second is by doing beta testing and withdrawing the product when the net social benefits are negative. The planner withdraws products from the market more often than the developer. It is possible that because it exercises caution in testing and implementing, the planner prefers a higher novelty level. In the Appendix B, we show an example of this possibility.

4 Regulating AI

In this section, we use the model with beta testing to study the implications of two forms of AI regulation.[2] The first is mandatory beta testing of external effects and approval by the regulator conditional on the test results. The second is making developers liable for the external effects of their algorithms.

[2]Recall that the model without beta testing is a particular case of the general model with $\beta = 0$.

4.1 Beta testing with conditional approval

Suppose the regulator mandates either beta testing or immediate release in period one and approves the implementation of the technology only if

$$\phi^2 \leq \frac{u(\ell)}{N+1}.$$

For a given ℓ, conditional approval in period two generates lower ex-ante uncertainty about the effects of the externality

$$\varsigma^2(\ell) \equiv \int_{-\sqrt{\frac{u(\ell)}{N+1}}}^{\sqrt{\frac{u(\ell)}{N+1}}} \phi^2 dG_\ell(\phi) \leq \sigma^2(\ell),$$

where G_ℓ denotes the CDF of $\phi(\ell)$. The ex-ante uncertainty about the externality at time two is given by the variance of the externality conditional on approval multiplied by the probability of approval in the second period.

The next proposition shows that this popular policy proposal does not implement the social optimum.

Proposition 7 (Conservatism in algorithm development)**.** *Suppose that the regulator controls whether an algorithm is implemented or not in both periods. Suppose, furthermore, that ex-ante uncertainty $\varsigma^2(\ell)$ is increasing in ℓ. Then, the developer chooses a level of novelty that is higher than the social optimum.*

To streamline the exposition, we relegate the proof to the appendix. The intuition for this proposition is that, for a given ℓ that is worthwile for the developer to implement, the utility of the developer is higher than social welfare, and this difference increases in ℓ. This difference in objectives occurs because the developer cares relatively less about the externality than the regulator.

4.2 Developers are liable for externalities

Suppose that the regulator allows the developer to freely choose the novelty level and whether to implement the AI algorithm but makes the developer liable for any negative externalities. This policy means that the developer is forced to pay:

$$\tau_t(\phi(\ell), \mu_t) = N\phi(\ell)^2\mu_t^2,$$

where μ_t denotes the number of households to whom the developer sells licenses.

In this case, the utility of the developer is given by

$$\mathcal{V} = (1-\beta)\left(\begin{cases}\mu_1 p_1 - \sigma^2(\ell)\mu_1^2 - \mathbb{E}[\tau_t(\phi(\ell), \mu_1)], & \text{if } p_1 \leq u(\ell)\mu_1 \\ 0, & \text{if } p_1 > u(\ell)\mu_1\end{cases}\right) + \beta\mathbb{E}[\mathcal{V}_2] - f(\ell),$$

where

$$\mathcal{V}_2 \equiv \begin{cases}\mu_2 p_2 - \phi(\ell)^2\mu_2^2 - \tau_t(\phi(\ell), \mu_2), & \text{if } p_2 \leq u(\ell)\mu_2 \quad \text{and} \quad \mu_1 > 0, \\ \mu_2 p_2 - \sigma^2(\ell)\mu_2^2 - \mathbb{E}[\tau_t(\phi(\ell), \mu_2)], & \text{if } p_2 \leq u(\ell)\mu_2 \quad \text{and} \quad \mu_1 = 0. \\ 0, & \text{if } p_2 > u(\ell)\mu_2.\end{cases}$$

It is still optimal for the AI developer to set $p_t = u(\ell)\mu_t$. Replacing this price and the liability payments, we see that the utility of the developer coincides with the objective function of the social planner when choosing the novelty level and making implementation decisions:

$$\mathcal{V} = (1-\beta)[u(\ell) - (N+1)\sigma^2(\ell)]\mu_1^2 + \beta\mathbb{E}[\mathcal{V}_2] - f(\ell),$$

where

$$\mathcal{V}_2 \equiv \begin{cases}[u(\ell) - (N+1)\phi(\ell)^2]\mu_2^2, & \text{if } \mu_1 > 0, \\ [u(\ell) - (N+1)\sigma^2(\ell)]\mu_2^2, & \text{if } \mu_1 = 0.\end{cases}$$

When AI developers are liable for external effects, private and social incentives become aligned. It follows that the privately optimal decisions coincide with the social optimum. We summarize these results in the following proposition.

Proposition 8 (Optimality of regulated equilibrium with full liability)**.** *Suppose that the developer is liable for the algorithm's external effects, then private and social incentives are aligned. This alignment implies that both the testing, implementation, and novelty level ℓ chosen by the developer are the same as in the first best.*

4.2.1 Limited Liability

The previous policy may require the developer to pay large sums that potentially exceed their profits. Suppose there is limited liability, in the sense that the liability payment cannot exceed the developer's profits

$$\tau_t(\phi(\ell), \mu_t) \leq p_t \mu_t.$$

We assume that, in this case, the regulator imposes the maximum payment that is consistent with non-negative profits for the AI developer,

$$\tau_t(\phi(\ell), \mu_t) = \max\{N\phi(\ell)^2 \mu_t^2, p_t \mu_t\}.$$

In this regulatory environment, it is still optimal for the developer to charge the maximum price $p_t = u(\ell)\mu_t$.

As before, it is always optimal to either release or beta test the AI algorithm. So, we only need to consider the case with full information in period two. This result implies that

$$\mathcal{V}_2^*(\ell) = [u(\ell) - \phi(\ell)^2]\mu_2^2 - \max\{N\phi(\ell)^2 \mu_2^2, u(\ell)\mu_2^2\}.$$

Suppose that $N\phi(\ell)^2 < u(\ell)$, then the developer decides to withdraw the AI algorithm from the market if $\phi(\ell)^2 > u(\ell)/(N+1)$ and to sell AI licenses to the whole population if $\phi(\ell)^2 \leq u(\ell)/(N+1)$. Instead, if $N\phi(\ell)^2 \geq u(\ell)$, then the developer makes no profits from selling AI licenses, but still suffers the external consequences of the AI algorithm.

Suppose that $N\phi(\ell)^2 \geq u(\ell)$, then the developer withdraws the AI algorithm from the market. Importantly, note that if $N\phi(\ell)^2 \geq u(\ell)$, then $\phi(\ell)^2 > u(\ell)/(N+1)$. It follows that the social planner agrees to withdraw the AI algorithm from the market.

In sum, even in the presence of limited liability, making AI developers liable for the external costs of their algorithms is sufficient to align incentives *in the second period*. We summarize these results in the following proposition.

Proposition 9 (No restrictions on product release with limited liability)**.** *Suppose that the regulator makes AI developers liable for the external consequences of the AI algorithm subject to limited liability. Then, the developer and the regulator agree on the implementation strategy in period two, i.e.,*

1. *If externalities are substantial, $\phi(\ell)^2 > u(\ell)/(N+1)$, both the regulator and the AI developer agree to withdraw the algorithm from the market.*
2. *If externalities are low, $\phi(\ell)^2 \leq u(\ell)/(N+1)$, both the regulator and the AI developer agree to release the algorithm to the entire population.*

It follows from Proposition 9 that, with limited liability, $\mathcal{V}_2^*(\ell, \mu_1) = \mathcal{W}^*(\ell, \mu_1) - y$. However, note that because $\mathbb{E}[\tau_1(\phi(\ell), N)] < N\sigma^2(\ell)$, then incentives in the first period are not aligned.

In the presence of limited liability, the AI developer chooses a higher level of ℓ than the planner. The AI developer sells licenses to the whole population if

$$u(\ell) > \sigma^2(\ell) + \mathbb{E}[\max\{N\phi(\ell)^2, u(\ell)\}]$$

and beta tests the AI algorithm if

$$u(\ell) \leq \sigma^2(\ell) + \mathbb{E}[\max\{N\phi(\ell)^2, u(\ell)\}].$$

The utility of the AI developer is given by

$$\mathcal{V} = (1-\beta)\max\{u(\ell) - \sigma^2(\ell) - \mathbb{E}[\max\{N\phi(\ell)^2, u(\ell)\}]\}N^2 + \beta[\mathcal{W}^*(\ell) - y] - f(\ell).$$

So, in general, making AI developers liable for external effects is insufficient to implement the social optimum in the presence of limited liability.

4.2.2 Limited liability with beta testing

Consider now an environment with limited liability in which the regulator has the ability to mandate beta testing for externalities. This policy implements the first best.

The regulator mandates beta testing when it is socially optimal and this mandate is sufficient to align private and social incentives. It is simpler for the regulator to implement mandatory, unconditional beta testing, but this policy is not optimal.

Proposition 10 (Limited liability and beta testing)**.** *Suppose that there is limited liability and the regulator always mandates beta testing. This regulatory environment does not implement the social optimum when beta testing is not optimal. When beta testing is socially optimal, private and social incentives are aligned: it is optimal for the AI developer to set the novelty level of their algorithms equal to the socially optimal novelty level. The AI developer beta tests the product in period one and releases the algorithm to the entire population if and only if externalities are low $\phi^2 \leq u(\ell)/(N+1)$. Furthermore, the developer's novelty choice is equal to the social optimum.*

The cost of beta testing is missing the opportunity to release the algorithm to the entire population in period one. As beta converges to one, this opportunity cost converges to zero. In this limit, the social optimum can be implemented by making the developers responsible for any external effects with limited liability and requiring mandatory beta testing.

Given that the beta testing phase typically represents only a small fraction of the AI algorithm's usage period, enforcing mandatory beta testing is nearly optimal in real-world applications.

5 Conclusion

In this paper, we study an environment with substantial uncertainty about the social costs and benefits of new AI technologies. We use this environment to assess different regulatory proposals put forth by policymakers in Europe and the United States.

We discuss optimal AI regulation in a world with a single country. In a world with multiple countries, international coordination might be necessary to achieve a global optimum.

Suppose external effects are local, that is, using an AI algorithm in one country does not impose externalities on other countries. If there is no limited liability, national regulators can achieve a global optimum by holding developers accountable for local external effects. With limited liability, local regulators must enforce optimal testing beta policies in addition to making developers liable for external effects. These implementations do not require global coordination.

International cooperation is generally required when there is tax competition or global externalities, that is when using an AI algorithm on one country imposes externalities on other countries. This cooperation is harder to implement when some governments pursue objectives that are different from social welfare (see Beraja, Kao, Yang, and Yuchtman, 2023 for empirical evidence along these lines).

Cooperation aside, the basic regulatory principle that emerges from our normative analysis is that implementing a social optimum requires both mandating beta testing for externalities and making developers liable for the external effects caused by their algorithms. This combination of regulatory measures achieves the social optimum even in the presence of limited liability.

Working out all the practical details and implementing this basic regulatory principle is an arduous task that requires substantial investment in expertise and computational resources by regulatory bodies. But as Isaac Asimov writes in his Foundation trilogy, ”It has been my philosophy of life that difficulties vanish when faced boldly.”

References

ACEMOGLU, D. AND P. RESTREPO (2022): "Tasks, Automation, and the Rise in U.S. Wage Inequality," *Econometrica*, 90, 1973–2016.

AGRAWAL, A., J. GANS, AND A. GOLDFARB (2022): *Prediction Machines: The Simple Economics of Artificial Intelligence*, Harvard Business Press.

BENIFEI, B. AND I.-D. TUDORACHE (2023): "Proposal for a Regulation of the European Parliament and of the Council on Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act)," Tech. rep., Committee on the Internal Market and Consumer Protection, European Union.

BERAJA, M., A. KAO, D. Y. YANG, AND N. YUCHTMAN (2023): "Exporting the Surveillance State via Trade in AI," Working paper, Brookings Center on Regulation and Markets.

BIDEN, J. R. (2023): "Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence," Tech. rep., The White House.

BROWN, N. AND T. SANDHOLM (2019): "Superhuman AI for Multiplayer Poker," *Science*, 365, 885–890.

BURSTEIN, A., E. MORALES, AND J. VOGEL (2019): "Changes in Between-Group Inequality: Computers, Occupations, and International Trade," *American Economic Journal: Macroeconomics*, 11, 348–400.

CALLANDER, S. (2011): "Searching and Learning by Trial and Error," *American Economic Review*, 101, 2277–2308.

ELOUNDOU, T., S. MANNING, P. MISHKIN, AND D. ROCK (2023): "Gpts are Gpts: An Early Look at the Labor Market Impact Potential of Large Language Models," *arXiv preprint arXiv:2303.10130*.

EUROPEAN COMMISSION (2020): "Report on the Safety and Liability Implications of Artificial Intelligence, the Internet of Things and Robotics," Tech. rep., European Commission.

——— (2022): "Regulatory Framework Proposal on Artificial Intelligence," Tech. rep., European Commission.

FARBOODI, M. AND L. VELDKAMP (2021): "A Model of the Data Economy," Tech. rep., National Bureau of Economic Research.

GITTINS, J. (1974): "A Dynamic Allocation Index for the Sequential Design of Experiments," *Progress in statistics*, 241–266.

ILUT, C. AND R. VALCHEV (2023): "Economic Agents as Imperfect Problem Solvers," *The Quarterly Journal of Economics*, 138, 313–362.

JONES, C. I. (2023): "The AI Dilemma: Growth Versus Existential Risk," Tech. rep., Technical Report, Stanford GSB. Mimeo.

JONES, C. I. AND C. TONETTI (2020): "Nonrivalry and the Economics of Data," *American Economic Review*, 110, 2819–2858.

LANGLOTZ, C. P., B. ALLEN, B. J. ERICKSON, J. KALPATHY-CRAMER, K. BIGELOW, T. S. COOK, A. E. FLANDERS, M. P. LUNGREN, D. S. MENDELSON, J. D. RUDIE, ET AL. (2019): "A Roadmap for Foundational Research on Artificial Intelligence in Medical Imaging: From the 2018 NIH/RSNA/ACR/The Academy Workshop," *Radiology*, 291, 781–791.

MILGROM, P. AND C. SHANNON (1994): "Monotone Comparative Statics," *Econometrica: Journal of the Econometric Society*, 157–180.

SILVER, D., T. HUBERT, J. SCHRITTWIESER, I. ANTONOGLOU, M. LAI, A. GUEZ, M. LANCTOT, L. SIFRE, D. KUMARAN, T. GRAEPEL, ET AL. (2018): "A General

Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go Through Self-play," *Science*, 362, 1140–1144.

SILVER, D., J. SCHRITTWIESER, K. SIMONYAN, I. ANTONOGLOU, A. HUANG, A. GUEZ, T. HUBERT, L. BAKER, M. LAI, A. BOLTON, ET AL. (2017): "Mastering the Game of Go Without Human Knowledge," *Nature*, 550, 354–359.

THOMPSON, W. R. (1933): "On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples," *Biometrika*, 25, 285–294.

A Proof of Proposition 2

Define

$$\mathcal{O}(\ell,d) \equiv \begin{cases} \{u(\ell)-\sigma^2(\ell)\}N^2 - f(\ell) & \text{if } d=1, \\ \{u(\ell)-(N+1)\sigma^2(\ell)\}N^2 - f(\ell) & \text{if } d=0. \end{cases}$$

If $d=1$, then $\mathcal{O}(\ell,1)$ is the objective function of the AI developer, whereas if $d=0$, then $\mathcal{O}(\ell,0)$ is the objective function of the social planner.

Define the maximum admissible level of novelty considered by the developer and the social planner $\overline{\ell}(d)$ for $d=0$ and $d=1$ respectively. Note that $\mathcal{O}(\ell,1) > \mathcal{O}(\ell,0)$. Given our assumptions on u, σ, and f, this condition implies that

$$\overline{\ell}(1) > \overline{\ell}(0).$$

This result shows that the social planner implements lower levels of novelty than the developer. In particular, because the social planner has a higher weight on the externality, it does not allow the implementation of any novelty level $\ell \in (\overline{\ell}(0), \overline{\ell}(1)]$.

Finally, let

$$\ell^*(d) \equiv \arg\max_{\ell} \mathcal{O}(\ell,d)$$

be the optimal novelty level for the developer if $d=1$ and the social planner if $d=0$. If $\ell^*(1) \notin [0,\overline{\ell}(0)]$, then it immediately follows that $\ell^*(0) < \ell^*(1)$ since $\ell^*(0) \in [0,\overline{\ell}(0)]$.

Then, suppose that $\ell^*(1) \in [0,\overline{\ell}(0)]$. We first show that the function $\mathcal{O}(\ell,d)$ satisfies strict single crossing in (ℓ,d). Then, using the monotone comparative statics results in Milgrom and Shannon (1994), we find that $\ell^*(1) > \ell^*(0)$.

Take $\ell' > \ell$, we show that

$$\mathcal{O}(\ell',0) \geq \mathcal{O}(\ell,0) \Rightarrow \mathcal{O}(\ell',1) > \mathcal{O}(\ell,1).$$

Note that

$$
\begin{aligned}
&\mathcal{O}(\ell',0) \geq \mathcal{O}(\ell,0) \\
\Leftrightarrow &\{u(\ell') - (N+1)\sigma^2(\ell')\}N^2 - f(\ell') \geq \{u(\ell) - (N+1)\sigma^2(\ell)\}N^2 + -f(\ell) \\
\Leftrightarrow &\{u(\ell') - \sigma^2(\ell')\}N^2 - f(\ell') - N(\sigma^2(\ell') - \sigma^2(\ell))N^2 \geq \{u(\ell) - \sigma^2(\ell)\}N^2 - f(\ell) \\
\Leftrightarrow &\mathcal{O}(\ell',1) - N(\sigma^2(\ell') - \sigma^2(\ell))N^2 \geq \mathcal{O}(\ell',0).
\end{aligned}
$$

Since $\sigma^2(\ell') > \sigma^2(\ell)$ and $\varsigma^2(\ell') > \varsigma^2(\ell)$, then the previous expression implies that

$$\mathcal{O}(\ell',1) > \mathcal{O}(\ell',0).$$

Since $\mathcal{O}(\ell,d)$ satisfies the single-crossing property, the results in Milgrom and Shannon (1994) imply that $\ell^*(d)$ is increasing in d. In other words,

$$\ell^*(1) \geq \ell^*(0),$$

i.e., the developer chooses a higher novelty level than the social planner. We prove this result by contradiction. Suppose that $\ell^*(1) < \ell^*(0)$. Since $\ell^*(0)$ is optimal at $d = 0$, it must be that

$$\mathcal{O}(\ell^*(0),0) \geq \mathcal{O}(\ell^*(1),0).$$

Since $\mathcal{O}$ satisfies the single-crossing property, then it follows that

$$\mathcal{O}(\ell^*(0),1) > \mathcal{O}(\ell^*(1),1),$$

which is contradicts the fact that $\ell^*(1)$ is optimal at $d = 1$.

B Example where social optimum has higher novelty than unregulated equilibrium

In this appendix, we provide an example in which, by being more cautious in beta testing and implementation, the social planner opts for a higher level of novelty than the AI developer.

The numerical example is as follows. Suppose that $u(\ell) = 2\sqrt{\ell}$ and that $f(\ell) = \chi\ell^2/2$ with $\chi = 10$. In addition, assume that $\beta = 0.7$ and that $\phi(\ell)$ is such that

$$\phi(\ell) = \begin{cases} \varphi\ell^2, & \text{with prob. } \frac{1-\alpha}{2} \\ 0, & \text{with prob. } \alpha \\ -\varphi\ell^2, & \text{with prob. } \frac{1-\alpha}{2}. \end{cases}$$

We set $\varphi = 1.0079$. In this case,

$$\sigma^2(\ell) = (1-\alpha)\,\overline{\phi}\psi^2\varphi^4.$$

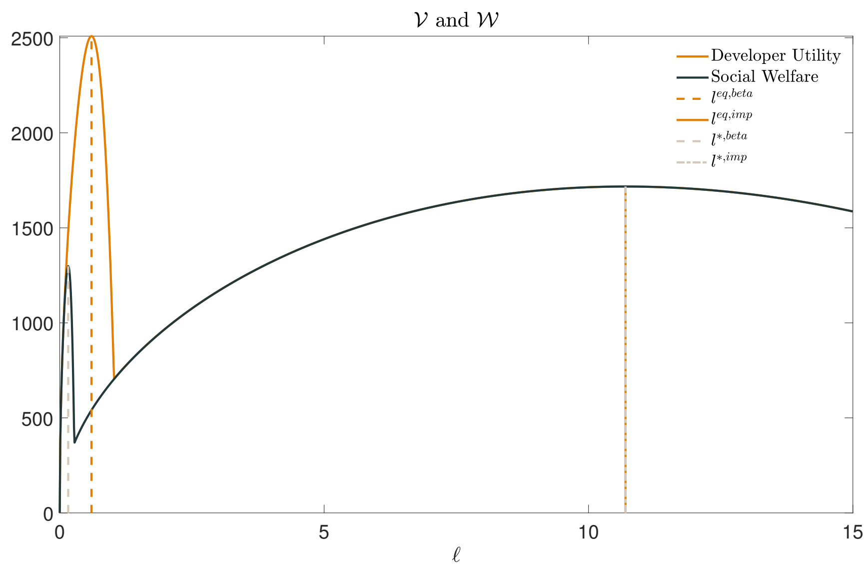

Figure 1: Example where social optimum has higher novelty than developer's optimum

In this case, the developer chooses a novelty level of 0.6 and immediately releases the algorithm to the whole population in period one. If the social planner was forced to release the algorithm to the whole population, it would choose a lower novelty level $\ell = 0.16$. However, by beta testing the product in period one, the planner actually prefers to have a much higher novelty level of 10.7.

If the developer was forced to beta test the algorithm at time 1, we see that they would choose the same novelty level as the planner. This result follows from the fact that, in this example, both the developer and planner agree to commercialize the AI at time 2 only if $\phi(\ell) = 0$. However, we see that beta testing is privately suboptimal since it would cost the developer the time 1 profits.

C Proof of Proposition 7

Define

$$\mathcal{O}(\ell,d) \equiv \begin{cases} (1-\beta)\{u(\ell)-\sigma^2(\ell)\}(\mu_1^*)^2 + \beta \int_{-\sqrt{\frac{u(\ell)}{N+1}}}^{\sqrt{\frac{u(\ell)}{N+1}}} [u(\ell) - hi^2] dG_\ell(\phi) - f(\ell) & \text{if } d = 1, \\ (1-\beta)\{u(\ell)-(N+1)\sigma^2(\ell)\}(\mu_1^*)^2 + \beta \int_{-\sqrt{\frac{u(\ell)}{N+1}}}^{\sqrt{\frac{u(\ell)}{N+1}}} [u(\ell) - (N+1)\phi^2] dG_\ell(\phi) - f(\ell) & \text{if } d = 0. \end{cases}$$

If $d = 1$, then $\mathcal{O}(\ell, 1)$ is the objective function of the AI developer, whereas if $d = 0$ then $\mathcal{O}(\ell, 0)$ is the objective function of the social planner. Let

$$\ell^*(d) \equiv \arg\max_{\ell} \mathcal{O}(\ell, d),$$

be the optimal novelty level for the developer if $d = 1$ and the social planner if $d = 0$.

We first show that the function $\mathcal{O}(\ell, d)$ satisfies strict single crossing in (ℓ, d). Then, using the monotone comparative statics results in Milgrom and Shannon (1994), we find that $\ell^*(1) > \ell^*(0)$.

Take $\ell' > \ell$, we show that

$$\mathcal{O}(\ell', 0) \geq \mathcal{O}(\ell, 0) \Rightarrow \mathcal{O}(\ell', 1) > \mathcal{O}(\ell, 1).$$

Let $\alpha(\ell) = \mathbb{P}\left[\phi(\ell)^2 \leq \frac{u(\ell)}{N+1}\right]$ be the ex-ante probability that the AI algorithm is implemented in period 2. Note that

$$
\begin{aligned}
&\mathcal{O}(\ell',0) \geq \mathcal{O}(\ell,0)\\
\Leftrightarrow &(1-\beta)\{u(\ell') - (N+1)\sigma^2(\ell')\}(\mu_1^*)^2 + \beta\{u(\ell')\alpha(\ell') - (N+1)\varsigma^2(\ell')\} - f(\ell')\\
&\geq (1-\beta)\{u(\ell) - (N+1)\sigma^2(\ell)\}(\mu_1^*)^2 + \beta\{u(\ell)\alpha(\ell) - (N+1)\varsigma^2(\ell)\} - f(\ell)\\
\Leftrightarrow &(1-\beta)\{u(\ell') - \sigma^2(\ell')\}(\mu_1^*)^2 + \beta\{u(\ell')\alpha(\ell') - \varsigma^2(\ell')\} - f(\ell')\\
&- (1-\beta)N(\sigma^2(\ell') - \sigma^2(\ell))(\mu_1^*)^2 - \beta N(\varsigma^2(\ell') - \varsigma^2(\ell))\\
&\geq (1-\beta)\{u(\ell) - \sigma^2(\ell)\}(\mu_1^*)^2 + \beta\{u(\ell)\alpha(\ell) - \varsigma^2(\ell)\} - f(\ell)\\
\Leftrightarrow &\mathcal{O}(\ell',1) - (1-\beta)N(\sigma^2(\ell') - \sigma^2(\ell))(\mu_1^*)^2 - \beta N(\varsigma^2(\ell') - \varsigma^2(\ell)) \geq \mathcal{O}(\ell',0).
\end{aligned}
$$

Since $\sigma^2(\ell') > \sigma^2(\ell)$ and $\varsigma^2(\ell') > \varsigma^2(\ell)$, it follows from the previous formula that

$$\mathcal{O}(\ell',1) > \mathcal{O}(\ell',0).$$

Because $\mathcal{O}(\ell,d)$ satisfies the single-crossing property, then the results in Milgrom and Shannon (1994) imply that $\ell^*(d)$ is increasing in d. In other words,

$$\ell^*(1) \geq \ell^*(0),$$

i.e. the developer opts for a higher novelty level than what the social planner. We prove this result by contradiction. Suppose that $\ell^*(1) < \ell^*(0)$. Since $\ell^*(0)$ is optimal at $d = 0$, it must be that

$$\mathcal{O}(\ell^*(0),0) \geq \mathcal{O}(\ell^*(1),0).$$

Since $\mathcal{O}$ satisfies the single-crossing property, then it follows that

$$\mathcal{O}(\ell^*(0),1) > \mathcal{O}(\ell^*(1),1),$$

which is contradicts the fact that $\ell^*(1)$ is optimal at $d = 1$.