Impacts of Aggregation on Model Diversity and Consumer Utility

Kate Donahue[1,2]
[1]Siebel School of Computing and Data Science, University of Illinois Urbana-Champaign
[2]CSAIL/LIDS, Massachusetts Institute of Technology
kpd@illinois.edu

Manish Raghavan
MIT Sloan School of Management & Department of Electrical Engineering and Computer Science
mragh@mit.edu

Abstract

Consider a marketplace of AI tools, each with slightly different strengths and weaknesses. By picking the right model for the task at hand, a user can do better than simply using the same model for everything. Routers operate under a similar principle, where sophisticated model selection can increase overall performance. However, such aggregation across models is often noisy, reflecting imperfect user choices or routing decisions. This leads to two main questions: first, what does a "healthy marketplace" of models look like for maximizing consumer utility? Secondly, how can we incentivize producers to create such models? We show that winrate, a standard benchmark in LLM evaluation, can incentivize model creators to *homogenize*, reducing consumer welfare. We propose a new mechanism, weighted winrate, which rewards models for answers that are higher quality, and show that it provably improves incentives for producers to specialize and increases consumer welfare. We conclude by exploring the impact of our theoretical results in empirical benchmark datasets and discussing implications for benchmark design.

1 Introduction

Modern AI models exhibit incredible versatility. They are capable of performing a broad range of tasks—coding, translation, writing, and many more. In some settings, AI models exhibit varying abilities based on the task at hand: some write superior code, while others excel at legal reasoning. When this is the case, savvy consumers of models can benefit from *aggregation* of models: strategically relying on the right model for the job, with the goal of outperforming any model in isolation. For example, Arena ranks individual models as well as a router (called Max) that aggregates individual responses across models; the router Max achieves the highest ELO score, outperforming individual models [3, 12]. Anecdotally, consumers are also beginning to act similarly, by strategically relying on certain models for particular tasks and other models for different tasks. In this sense, consumers in our setting derive utility from a *set* of models, not just a single one.

However, the benefits of aggregation rely on models differing from each other: aggregation across identical models brings no added benefits to consumers. Recent theory and evidence suggests that AI models are *homogeneous* (that is, having correlated strengths and weaknesses), failing to demonstrate the heterogeneity that might yield productive aggregation in practice [27, 6, 17, 26]. Why does the market fail to produce heterogeneous models?

While correlated data and architectures likely play a strong role, one additional explanation could be incentives produced by benchmarks. Benchmarks (such as Arena) [3, 12] rank models according

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

to their performance: firms may try to design models that perform well on these benchmarks in the hope of attracting increased publicity, consumers, or funding. However, it's unknown whether benchmarks themselves produce incentives for firms to create heterogeneous models, or whether they create incentives for firms to homogenize.

In this work, we investigate the incentives created by common approaches in benchmarking model performance. We use evaluation as a lever - if we change the design of the benchmarks, are we able to incentivize firms to create models that better align with consumer utility? We focus on the incentive structures that govern the relationship between aggregation and ecosystem-level properties like homogeneity, asking whether alternative mechanisms could better align the objectives of producers with the utility of users. Along the way, we find several counterintuitive properties of aggregation that pose challenges for utility alignment.

Contributions. In this paper, we investigate how the imperfect aggregation of agent outputs—whether through automated routers or strategic human selection—shapes both the realized utility of generative AI and the incentives produced by benchmarks in the LLM marketplace. We focus on two central questions: what does a "healthy marketplace" of models that satisfies consumer demand look like? Second, how can we incentive self-interested producers to create such models? We begin by proposing a formal model of aggregation in Section 2: the world is divided into discrete *tasks*, where agents have different performance on each task.[1] Consumers noisily choose a model for each task according to a choice model—most of our results focus on the Bradley-Terry choice model [8, 35, 34], though we consider alternatives as well[2]. In Section 4, we focus on consumer welfare and demonstrate that the addition of a new model or improvement of an existing model does not necessarily benefit consumers. Intuitively, unless consumer choice is perfect, a worse model can "distract" a user and lead to worse outputs.

In Sections 5 and 6, we study how producer incentives affects consumer utility under aggregation. If producers are motivated by benchmarks using *winrate* (the probability their model is picked), we show that competition leads to homogenization: however, consumer welfare would be maximized by specialization, where different models invest in improvements on different tasks. We propose and analyze a benchmarking mechanism that better aligns producer objectives with consumer welfare. Finally, we include experiments in Section 7 to explore how our results change incentives in real "markets" of models, before concluding.

2 Model

Task structure: We assume that there are $T \geq 2$ different "tasks" or "task types", and a set of $\mathcal{M}$ different models. For each task $t \in [T]$, a given model m has value $v_{m,t} \geq 0$ or $v_{m,t} = \emptyset$ (that is, fails to return a response for this task). We will assume that such value is deterministic or, alternatively, we will consider $v_{m,t}$ to be the expected value when using model m on task t. For a given set of models $\mathcal{M}$, we will use $\vec{v}_t$ to denote the list of values of responses on task t. We will use $\oplus$ to denote adding another value to the list: that is $\vec{v} \oplus v$ is the vector $\vec{v}$ with the value v appended to it. We will often use $N = |\mathcal{M}|$, the number of models in a market. With this task structure, we can express the fact that different models have different strengths and weaknesses. A model's *total value* is given by its sum of values over tasks, $V = \sum_{t \in [T]} v_t$, and its *average value* is its total value, divided by the number of tasks: these quantities will turn out to be crucial for later properties of consumer welfare and producer incentives.

Aggregation: The core phenomenon that we will study is *aggregation*, when a user derives utility across multiple available models. We will focus on *choice-based aggregation*, where for each task a user has some probability $p_m(\vec{v}_t) \in [0, 1]$ of picking a particular model. We require that the total probability of models being picked sums up to 1: $\sum_{m \in [N]} p_m(\vec{v}_t) = 1$. A user's utility for a particular task given choice-based aggregation is given by the expected value of the model she picks: $u(\vec{v}_t) = \sum_{m \in [N]} v_{m,t} \cdot p_m(\vec{v}_t)$. We will define $p_i(\vec{v}) = 0$ if $\vec{v}_i = \emptyset$: any model that fails to return an answer has probability 0 of being picked. A few special cases of choice-based aggregation are: *random* aggregation, where the choice model picks randomly among items, and *optimal* aggregation, which always picks the item with highest value (randomizing in the case of ties).

[1] We will sometimes use "agent" to refer to an AI model to avoid overloading the term "model."

[2] See Appendix B.1 for a detailed analysis

While optimal and random aggregation are helpful limiting cases, neither is a particularly useful model of aggregation in settings we are interested in studying, such as human decision-making or routing between LLMs. Instead, we might expect decisions to be smooth and lie somewhere in between these two extremes: better than random, but worse than optimal, and most likely to "confuse" items of similar value (most likely to incorrectly pick a lower-value response when it is close in value to the best option, for example). More generally, we'd like a flexible choice model that can express a range of routing "accuracy" levels in picking the best model for the task. One choice model satisfying all of these properties is Bradley-Terry-Luce:

Definition 1 (Bradley-Terry-Luce)**.** *Bradley-Terry-Luce (BTL) (alternatively known as Bradley-Terry or Plackett-Luce [8, 35, 34]): with a given temperature $\beta > 0$, the probability of picking a item m from a list of values $\vec{v}_t$ is given by: $p_m(\vec{v}_t) = \frac{\exp(v_{m,t}/\beta)}{\sum_{m' \in \mathcal{M}} \exp(v_{m',t}/\beta)}$.*

BTL aggregation approaches random selection in the limit of high temperature ($\beta \to \infty$) and approaches perfect selection in the limit of low temperature ($\beta \to 0$). The probability of picking an item increases in value, and the probabilities are most close for items that are close in value. Because of these properties, Bradley-Terry is frequently used as a model of human behavior in market competition, e.g. [23, 14, 45]. In Appendix A.2 we include experiments exploring its applicability to modeling errors in routing tasks, and in Appendix B.1 we include experiments generalizing our results to three other choice models reflecting imperfect human decisions.

Consumer welfare and producer incentives: Throughout, we will assume that consumer welfare is given by the sum of user utility over tasks, assuming aggregation by the BTL choice function: $\sum_{t \in [T]} \sum_{m \in [N]} v_{m,t} \cdot p_m(\vec{v}_t)$. In general, we will *not* assume that producers are directly moving to optimize consumer welfare. Instead, we will assume in the status quo, they are motivated by *winrate*, or the probability that their model is picked by the user: $p_m(\vec{v}_t)$.[3]. One motivation for this is that winrate is used in leaderboard design, as in [12], while other motivations come from the common assumption in modeling market competition that producers are aiming to maximize market share (e.g. [23, 14, 45]).

In Sections 5 and 6, we will show that winrate and consumer welfare are at odds with each other: producers optimizing for winrate could lead to sub-optimal consumer welfare. To address this, we propose a mechanism to align user and producer incentives: weighted winrate (Definition 2). Weighted winrate rewards producers not just for an output that is better than other outputs (reflected by winrate, $p_m(\vec{v}_t)$), but also for a higher value itself (reflected by value $\vec{v}_{m,t}$).

Definition 2 (Weighted winrate (ours))**.** Weighted winrate *is the probability that a model is picked* multiplied by the value it gives*, or $p_m(\vec{v}_t) \cdot \vec{v}_{m,t}$*

We view this mechanism as very natural: it corresponds to producers adopting *outcome-based pricing*, where they charge for value provided instead of proportionally to compute costs. One specific feature of modeling LLMs is that we can intervene through leaderboards to implement mechanisms, while such an intervention might be more challenging in other markets. Weighted winrate could be implemented by directly asking users to indicate not only their most preferred option from a set of responses, but also their value for that response. This could be done with coarsened features (e.g. score from 1 to 10, or a 1-5 star rating): we discuss complexities around eliciting labels in Section 8.

Producer actions (model creation and model replacement): Throughout the paper, we will study two different types of strategic behaviors that producers can take in pursuit of their objectives: *Model creation* is when a producer creates a new model and adds it to the existing market, while *model replacement* occurs when a producer replaces an existing model with a new one (e.g. with local post-training). Model creation mimics a setting where a producer creates an entirely new model, with complete flexibility over its strengths and weaknesses (limited by some cap in *total value*). Model replacement mimics a setting where a model producer can make small (instantaneous) changes to an *existing* model (e.g. post-training), that may slightly change its existing strengths and weaknesses. While model creation and replacement are connected, they are different actions with different implications for consumer welfare. For example, suppose we begin with a market made of two models, $\{A, B\}$. Model creation would result in the new market $\{A, B, C\}$, while model replacement would result in the new market $\{A, B'\}$, with different impacts on consumer welfare. In Appendix B.2 we include experiments on model creation with iterated best response.

[3]Across multiple tasks, we assume they aim to maximize the average winrate.

3 Related work

Aggregation of generative models: The closest area of research to ours is work that studies aggregation with generative AI tools. For example, [1] designs optimal aggregation mechanisms given some fixed set of multiple LLMs: this could be viewed as the inverse of our setting, where we study a fixed aggregation mechanism and study how to incentivize the creation of models that satisfy that it. A related paper on incentives in benchmarks is Ethayarajh and Jurafsky [15]: their focus is on users picking models according to their rank on a benchmark, while we study consumers *aggregating* across models on different tasks. Another paper that studies consumer aggregation is [2], which differs from ours primarily in our focus on mechanisms and our smoother aggregation function (BTL). From an applied angle, Lu et al. [33] study "second opinions" in humans relying on AI tools. Other work studying the performance of agentic teams with specific focus on diversity includes [25, 5, 30, 43, 41].

A complementary line of work applies insights from voting theory to settings with multiple benchmarks, where different benchmarks may reflect varying performance across tasks, for example. Some work aims to design multi-task benchmarks that incorporate rankings of the same models across multiple distinct tasks: this work differs from ours in that the goal is typically to produce a single benchmark, while we study how producers compete over multiple tasks [46, 18, 13, 39]. Other work strategically selects subsets of representative benchmarks [36]. Finally, a related line of literature studies the performance of teams [22], as well as specifically studying the utility of voting or other aggregation mechanisms with diverse teams (e.g. [24]).

Strategic competition: One related line of work studies model-providers competing over users, as in [23, 14, 45, 44]. There are strong connections, such as the use of modeling user choice through BTL functions and characterizing producers' best response. However, one key difference of our paper from this work is our focus on the mechanism design aspect: our goal is not only to understand the behavior of producers, but to change their incentives with the goal of better aligning with consumer welfare. Other related works (e.g. [19, 20, 32]) have studied whether it is possible to improve consumer utility by designing mechanisms that agents strategically best-respond to. This line of work tends to differ from ours in its motivating application area (where ours focuses more specifically on generated AI. A few papers have specifically looked at mechanism design for leaderboards (e.g. [11, 21]): these tend to focus more on score inflation, as compared to our goal of managing homogenization.

More generally, there has been a rich and long literature studying agents competing over items, such as the Colonel Blotto game, with a pair of Colonels competing over a series of "battlefields" [7], and Tullock contests where agents aim to allocate value in the hope of winning a probabilistic *contest* [42]: see Appendix A.1 for a deeper discussion. A few key differences are the actions available to players differ substantially ("model creation," "model replacement", which we view as being more natural for LLM creation) and our work's focus on consumer welfare.

4 Aggregation and consumer welfare

First, in this section we begin by studying how consumer welfare given by aggregation over a set is influenced by model creation or model replacement. Our primary results for this section will illustrate that optimizing consumer welfare is surprisingly non-trivial, even leaving aside questions of producer incentives. All proofs are deferred to Appendix C.

Model creation: As a warm-up, note that model creation with perfect aggregation or random aggregation always has straightforward impacts on consumer welfare. With model creation, adding a new model always weakly improves consumer welfare, while with random aggregation, it helps only when the new model has higher average value.

Theorem 1 shows that BTL aggregation adds complexity to this story. Specifically, given an existing set of models $\mathcal{M}$ and a new model B, there are scenarios where B can be *weaker* than every model in $\mathcal{M}$ and still the creation of B can help consumer welfare, and also be scenarios where B is *better* than every model in $\mathcal{M}$ and the addition can *hurt* consumer welfare.

Theorem 1. *Given BTL aggregation, it is possible to have a set of existing models $\mathcal{M}$ and a new model B such that either 1) B has higher average value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$ decreases consumer welfare, or 2) B has* lower *average value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$* increases *consumer welfare. Moreover, this effect is dependent on temperature (β).*

The implication of Theorem 1 is that when users aggregate across models, optimizing model creation for consumer welfare can be non-intuitive, an insight which will be useful in our later analysis of mechanisms to improve consumer welfare.

Model replacement: We next turn to analyzing model replacement: when one model is replaced with a different model with different values on each task. Again, perfect aggregation and random aggregation provide useful limiting cases for when model replacement improves consumer welfare: Note that with *perfect aggregation*, model replacement weakly improves consumer welfare if and only if it weakly improves value on some task. With *random aggregation*, model replacement improves consumer welfare if and only if it improves the average value of the model. One natural property we might want is the following: weak task-wise improvement in models would only lead to weak increases in utility. That is, replacing a model with one that dominates the previous model on every task should only *improve* consumer welfare. Monotonicity (Definition 3) makes this property formal.

Definition 3 (Monotone)**.** *A choice function is monotone if task-wise increase in values always leads to increased consumer welfare: for all $\vec{v}_m, \vec{v}'_m$ with $\vec{v}_{m,t} \leq \vec{v}'_{m,t}$ for all $t \in [T]$, $u(\vec{v}) \leq u(\vec{v}')$.*

While monotonicity is a natural property to desire, unfortunately it is not guaranteed. Lemma 1 gives an example where monotonicity *fails* with BTL aggregation. For intuition, the proof is given by example: consider a set with two values $\{x, 5\}$, for $x = 1, 4$. When the value of x increases from 1 to 4, the value of the first item increased, but so does the probability that element x would be picked. For BTL aggregation with noise parameter $\beta = 1$, it turns out that the value of set $\{1, 5\}$ is *higher* than the value of set $\{4, 5\}$.[4] At a high level, the intuition behind non-monotonicity is that a mediocre answer can be more distracting than a clearly terrible answer: for this reason, model improvement can sometimes be harmful.

Lemma 1. *BTL aggregation is non-monotone: e.g. a model could weakly* increase *the value of its responses on every task, and consumer welfare might* decrease*.*

One natural question might be whether it is BTL specifically that leads to non-monotonicity. Surprisingly, the answer turns out to be no—in fact, any *separable* choice function (Definition 4) also cannot be monotone, as shown in Theorem 2. Intuitively, a function is separable if the probability of an item being picked is proportional to some function of its value. Thus, Theorem 2 implies non-monotonicity holds more broadly.

Definition 4 (Separable)**.** *A choice function $p(\cdot)$ is separable if there is some function $f(v) \geq 0$ such that $p_i(\vec{v}) \propto f(\vec{v}_i)$: we also require $f(\cdot)$ be differentiable, monotonic, and non-constant.*

Theorem 2. *A separable choice function cannot be monotone.*

However, we also show that non-monotonicity is not an inherent trait of choice functions: in the proof of Lemma 2 we give an example of a choice function $p(\cdot)$ constructed so that it satisfies monotonicity. At a high level, the aggregation function in Lemma 2 is based on the *gap* between values between items, rather than the values themselves, which side-steps the impossibility condition in Theorem 2.

Lemma 2. *There exists an aggregation function that is monotone in value.*

Finally, Lemma 3 shows that BTL aggregation satisfies monotonicity under certain conditions. Specifically, we will say that a choice function is in the "monotone region" whenever, for a specific set of values $\vec{v}$, the derivative of consumer utility with respect to $\vec{v}_i$ is positive. [5] Lemma 3 shows that the BTL function is in the monotone region under certain conditions on the values and level of noise (β). In particular, it is in this region when $\beta = 0$ (perfect aggregation), and when β is sufficiently large (sufficiently noisy), where "sufficiently large" is a function of the values $\vec{v}$.

Lemma 3. *BTL aggregation function is in the monotone region when 1) $\beta = 0$, corresponding to perfect aggregation, or 2) when β is sufficiently high: that is, for a given set of values $\vec{v}$ and a model i, BTL aggregation is guaranteed to be in the monotone region whenever $\beta > \max_{i,j} \vec{v}_j - \vec{v}_i$.*

Taken together, results in this section show that model improvement can fail to improve consumer welfare for a surprisingly broad set of scenarios. We will use insights from this section in designing incentives for model improvement in the following sections.

[4] Related phenomena have been seen in auctions, voting, and social choice: see Appendix A.1 for a deeper discussion.

[5] A function is monotone if for all values, it is always in a monotone region.

5 Incentivizing model creation

In Section 4 we studied how model creation and model replacement affect consumer welfare. However, in current marketplaces, models are created by specific producers, who may be incentivized by objectives that are different from purely optimizing consumer welfare. Specifically, we will assume that producers are currently motivated by *winrate*: the probability that their model is selected. In this section and in Section 6, we will show how winrate and consumer welfare may be at odds with each other: producers optimizing for winrate could lead to sub-optimal consumer welfare. Finally, we will discuss our proposed mechanism to better align user and producer incentives (weighted winrate). In this section, we will explore how to incentivize *model creation*: the addition of an entirely new model. All proofs are deferred to Appendix D.

We will assume the following structure: there are T tasks and a set of existing models $\mathcal{M}$, and each task has value $\vec{v}_t$. Then, a producer decides to make a new model of total value V: assume that they can create this model by allocating non-negative value arbitrarily across tasks, and choose to do so according to some objective: winrate, weighted winrate, or consumer welfare. Intuitively, this is connected to how producers might make strategic choices in how to shape their models based on incentives, such as investing in training data or capabilities tailored towards specific tasks[6]. We will explore whether producers choose to specialize, placing most value on a few tasks, or homogenize, spreading value across tasks. We will show that winrate leads producers to homogenize, which leads to lower utility, and our proposed mechanism (weighted winrate) leads producers to specialize, leading to improved consumer welfare.

Model producer actions optimizing different objectives: We will begin by analyzing the optimal response to each objective. First, Lemma 4 shows that the optimal allocation for consumer welfare is to either maximally specialize, placing all value on a single task, or to abstain (if the total amount of value V for the new model is very low). This result will be the benchmark by which we measure how well producers satisfy consumer welfare.

Lemma 4. *If a new model with total value V aims to maximize consumer welfare, its best action is: if V has higher value than the minimum existing values on tasks ($V \geq \min_{i\in[T]} u(\vec{v}_i)$), specialize by allocating all value on a single task i which maximizes $p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i))$. If V is not sufficiently high, then its best action is to abstain (fail to create a model).*

Next, Theorem 3 studies best response for winrate, showing that if the new model is sufficiently powerful (has sufficiently high V), producers are incentivized to *homogenize*, equalizing their winrate across tasks. Intuitively, this occurs because producers optimizing for winrate face decreasing marginal benefit by increased value on a single task. Note the lower bound on V in Theorem 3 depends on the strength of aggregation β: the producer's incentives are influenced by consumer noise[7].

Theorem 3. *Consider a set of tasks with existing values $\{\vec{v}_i\}$ across tasks. Then, an incoming model with total value V optimizing for* total winrate *must equalize winrate across tasks whenever: $V > 2 \cdot T \cdot \beta \cdot \max_{j\in[T]} \log\left(\sum_{i\in[N]} \exp(\vec{v}_{ji}/\beta)\right)$*

Finally, Lemma 5 analyzes the incentives produced by our new mechanism, weighted winrate (Definition 2): weighted winrate incentivizes producers to maximally specialize by placing all value on a single task and returning a value of 0 elsewhere.

Lemma 5. *Suppose there is a set of existing models and tasks with values $\{\vec{v}_t\}$. Then, an incoming model with total value V optimizing for weighted winrate would best respond by putting all value on a single task (the task that maximizes $p_{N+1}(\vec{v}_i \oplus V)$), and value 0 elsewhere.*

The results of Lemmas 4 and 5 show that in order to maximize weighted winrate and consumer welfare, the optimal allocation has all value on a single task, and either value 0 or $\emptyset$ (abstain) on all other tasks. This outcome could reflect settings where a producer strategically chooses to not participate in a type of task—for example, Nano Banana specializes in image generation, while Gemini Deep Research specializes in academic research. However, it may not be realistic to have models with such performance, either because the benchmarks themselves may become saturated at a

[6] In Appendix B.2 we include experiments on model creation with iterated best response.

[7] Theorem 3 gives us a lower bound on V, where any producer above this must be incentivized to homogenize. For a complementary result, Lemma 15 in Appendix D gives an *upper* bound on V, below which any producer must specialize.

particular maximum value, or because scaling laws suggest that achieving increased performance becomes increasingly hard.

However, the main story of our results generalizes to such settings. Here, we assume that every task has some maximum value v_i^* above which value cannot be achieved: this could reflect the maximum score on a benchmark, or the maximum utility that a user can obtain for a response. Then, our results for weighted winrate and consumer welfare generalize as follows:

Lemma 6. *If each task i has a maximum value v_i^*, the allocation that maximizes weighted winrate or consumer welfare is to greedily allocate value for each task according to the objectives in Lemmas 4 or 5 respectively.*

Impacts on consumer welfare: Next, we will turn to analyzing the impact of weighted winrate on consumer welfare: does it lead to better social welfare than winrate? The weighted winrate allocation in Lemma 5 differs from the social welfare allocation in Lemma 4 in two ways: first, the choice of which task to allocate value in may differ, secondly, agents optimizing for social welfare would abstain on some tasks (fail to return a value), while agents optimizing for weighted winrate would return an output of value 0. [8]. Given these differences, it is clear that weighted winrate is weakly worse for consumer welfare than directly optimizing for welfare itself. But, as we show in Theorem 4, weighted winrate is always at least as good as winrate: whenever winrate incentivizes producers to homogenize outputs by equalizing winrate (as in Theorem 3), the incentives produced by weighted winrate would lead to strictly higher consumer welfare.

Theorem 4. *Whenever agents would respond to winrate by equalizing winrate (conditions of Theorem 3), weighted winrate would lead to strictly higher consumer welfare than winrate.*

6 Incentivizing model replacement

In Section 5, we studied incentives in model *creation*, when producers are making an entirely new model. In this section, we will study incentives in model *replacement*, when an existing model has instantaneous changes made to its strengths and weaknesses, mimicking small tweaks like post-training in models. Incremental updates to existing models could occur with post-training of a mostly-finalized model to fine-tune its performance on specific tasks, or minor tweaks to capabilities and tool use, potentially in response to competition from other models or performance on leaderboards.

The high-level story will be that the incentives and outcomes are very similar to with model creation: winrate tends to incentivize homogenization, to the detriment of consumer welfare, while weighted winrate gives incentives that leads to improved consumer welfare. All proofs are deferred to Appendix E. We will assume the following structure: there are $T > 1$ tasks, and exactly two producers $N = 2$. Each of the producers A and B has an existing allocation across tasks $\vec{v}_a, \vec{v}_b$. Each producer can make instantaneous changes to its relative allocation. In particular, they calculate the derivative of some objective (e.g. winrate, weighted winrate, consumer welfare) and change their allocation in order to improve their utility according to this objective. In Appendix B.3 we include experiments generalizing our results to larger numbers of producers $N > 2$.

Model producer actions optimizing different objectives: Again, we will begin by analyzing how model producers would respond to different objectives (social welfare, winrate, and weighted winrate). Specifically, Theorem 5 considers a pair of tasks, and shows that for winrate, producers are always incentivized to improve on the *same* task, but for optimizing for consumer welfare, they would always prefer to optimize *different* tasks. While Theorem 5 shows how producers would make decisions between *pairs* of tasks, this directly implies results about the total ordering over tasks: with winrate, producers would have identical orderings over tasks they choose to improve in, and for consumer welfare, producers would have exactly inverted orderings over tasks they choose to improve in. Note that this matches the pattern we saw in Section 5: optimizing for winrate tends to incentivize homogenization, while optimizing for consumer welfare tends to lead to specialization.

Theorem 5. *For $N = 2$ agents optimizing for winrate, agents would always pick the* same *task to increase value in, whereas agents optimizing for consumer welfare always pick* different *tasks.*

[8] Note that the weighted winrate mechanism leaves producers indifferent between returning an item of value 0 or abstaining, as both give value $\vec{v}_{i,t} \cdot p_i(\vec{v}_t) = 0$: we take the worst-case assumption that producers return a response of value 0. However, these actions are different from the consumer's perspective, because returning a response of value 0 steals probability mass from higher-value responses.

Lemma 7 shows that our proposed mechanism, weighted winrate, incentivizes behavior aligned with social welfare maximization whenever the models start out by being sufficiently specialized: that is, when their winrates begin by being sufficiently different.

Lemma 7. *For $N = 2$ agents responding to weighted winrate, agents are incentivized to improve in* different tasks *whenever agents begin by being sufficiently specialized (e.g. $|p_a(\vec{v}_i) - p_a(\vec{v}_j)|$ is sufficiently large).*

Impacts on consumer welfare: Given that the incentives produced by weighted winrate align with socially optimal behavior, it is natural to think that weighted winrate should increase social welfare relative to winrate. We study this question next. Because agent development is on a trajectory of incremental improvement, not at equilibrium, we will consider *instantaneous consumer welfare*, which is the change in consumer welfare given the instantaneous change in producers' values over tasks. In Section 5, Theorem 4 showed that weighted winrate improves consumer welfare in the model *creation* setting. We might hope to similarly show that weighted winrate always improves consumer welfare in the model *replacement* setting. However, the story turns out to be slightly more nuanced. First, Lemma 8 shows that weighted winrate does *not* always lead to an improvement over winrate.

Lemma 8. *Weighted winrate does* not *always lead to improvement in instantaneous consumer welfare, as compared to the incentives produced by winrate.*

Specifically, this proof is given by an example (formally presented in Table 7 in Appendix E.). However, this example has an unusual property: one model (B) is worse than the other model (A) on *both* tasks, and moreover, the derivative of consumer welfare with respect to improvement (instantaneous consumer welfare) in model B is *negative* for both tasks: it is in the *non-monotone* region as explored in Section 4). Theorem 6 shows that this is not an accident: every example where weighted winrate incentives producers to pick a worse pair of tasks than winrate, it *must* be a setting where one producer is in the non-monotonic regime on *both* tasks. Note that Theorem 6 is not an if-and-only-if condition: it is a conservative guarantee.

Theorem 6. *Consider a model replacement setting with $N = 2$ agents: the only scenario where weighted winrate causes* worse *instantaneous consumer welfare than winrate is when one player is in the non-monotonic regime on* both *picked tasks.*

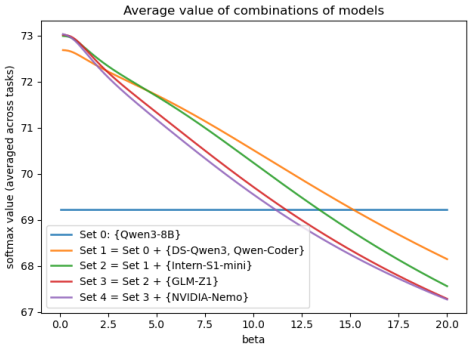
We view this condition as being very useful for understanding how to guide producer incentives. Specifically, given that the only failure of weighted winrate occurs when one player is in the non-monotone regime, the best outcome for consumer welfare would be if that player could be incentivized to *abstain* entirely: that is, to not improve their model at all. However, weighted winrate cannot incentivize producers to abstain, and we view studying how to incentivize model abstention as a fascinating and fruitful avenue for future work.

7 Experiments

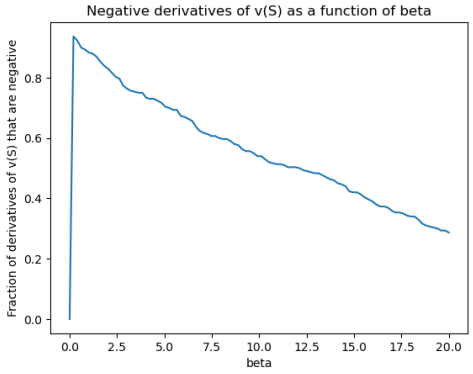
Finally, we will explore what our theoretical results imply about real markets of models. We will use the benchmark LLMRouterbench [29] because it includes real-valued estimates of model performance on discrete tasks with substantial heterogeneity in performance.[9] In Appendix A.3 we include experiments on two other datasets (MT-Bench-101 [4] and WildBench [31]).

Maximizing consumer welfare (extending Section 4): Theorem 1 showed that there are scenarios where adding "weaker" models in isolation could be beneficial, depending on the noise in BTL aggregation (β). Figure 1a shows the optimal subset of models as a function of β: there are settings where a team strictly outperforms the single strongest model alone, and the size of the optimal team decreases as noise increases. Next, we explore the non-monotonicity we studied in Section 4, where we showed that there can be cases where increasing the value of a model could *decrease* consumer welfare. However, our theoretical results left open the question of how "common" this phenomenon was in real datasets. In Figure 1b, we look at every model/task combination and see whether the model is within the non-monotone region: whether improvement by this model would *decrease* consumer welfare. Interestingly, a relatively large proportion of models are in the non-monotone region, though this proportion is decreasing in β (as we would expect from Lemma 3).

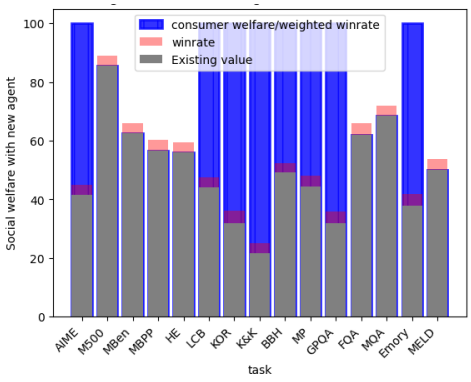
[9] Code and data to reproduce all figures is available at `https://github.com/kpdonahue/aggregation`.



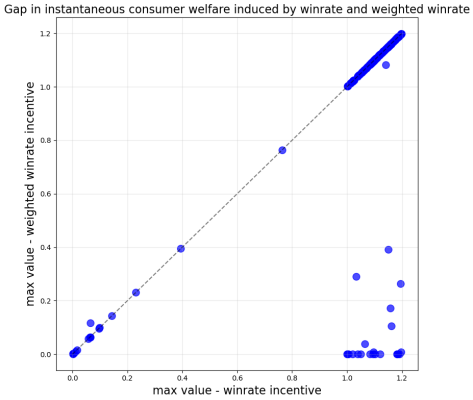
(a) Value of combinations of models.



(b) Prevalence of non-monotonicity.



(c) Change in consumer welfare given different incentives for model creation.



(d) Gap in instantaneous consumer welfare given different incentives in model improvement.

Figure 1: Experiments with LLMRouterBench data [29]

Incentivizing model creation (extending Section 5) Next, we turn to incentivizing producers to create models that improve consumer welfare. Note that in LLMRouterBench [29] the value of each individual task is bounded by value 100, and thus we apply adapted versions of our results (Lemma 6). We consider an existing market comprised of the 5 weakest models and an incoming model producer with average of 53 over all tasks: we chose these parameters to ensure new model is sufficiently powerful, as in Theorem 3. Then, we consider the actions that each objective (winrate, weighted winrate, consumer welfare) would incentivize and their impact on consumer welfare. For this setting, the winrate objective would incentivize the producer to equalize winrate across tasks, while weighted winrate and consumer welfare would both incentivize the producer to maximally specialize on 8 tasks (either abstaining or giving value 0 to the others). Figure 1c visually shows the change in consumer welfare: the prior value of the existing marketplace is in gray, change in value induced by winrate is in red, and change in value induced by consumer welfare or weighted winrate incentives are in blue, with overlapping regions in purple. The high-level takeaway from this figure is that weighted winrate incentives the creation of models that lead to much larger benefit to consumers: the blue region is much larger than red.

Incentivizing model replacement (extending Section 6) In Section 6 we studied model replacement, where the goal was to incentivize model creators to pick tasks to improve in that increase consumer welfare. Theorem 6 showed that weighted winrate improves incentives for consumer welfare, so long as it is *not* the case that one of the two producers is in the non-monotone regime on both tasks. To explore the incentives for model replacement with this dataset, we take every pair of models and calculate tasks they would be incentivized to improve in, given different objectives (winrate, weighted winrate, consumer welfare). In Figure 1d, we display how far from optimal (social welfare) each of weighted winrate and winrate incentives are. The x-axis gives the gap between winrate and optimal, while the y axis gives the gap between weighted winrate and optimal: points are along the diagonal if

they give equal incentives, and below (resp. above) the diagonal if weighted winrate is better (resp. worse). Note that weighted winrate gives better incentives *almost everywhere*: almost every point is along or below the diagonal. This suggests that empirical performance of weighted winrate is stronger than our conservative theoretical guarantees.

8 Future directions

Our contributions: In this work, we have studied a setting where users derive utility from *aggregating* over multiple models, capturing the idea that consumers may noisily select a model based on their expectations of performance on a given task. We showed that under aggregation, the relationship between agent quality and consumer welfare is surprisingly non-trivial: for a broad range of choice functions, there are cases where increasing the quality of any individual agent could *decrease* consumer welfare. Despite this complex interaction, we studied how aggregation can lead to a wedge between producer incentives and consumer welfare: producers may focus too much on competing for attention in the same tasks, whereas consumers benefit more from specialization. We proposed a novel mechanism, weighted winrate, and showed that it better aligns producer objectives with consumer welfare.

Future directions: There are many natural directions for extensions in this space. To start, one remaining gap in our solution was incentivizing producers to *abstain* (fail to give a response). With weighted winrate, producers are indifferent between returning a response of value 0 and failing to respond at all, but when users are noisy, returning a response with low or 0 value can *hurt* their utility. Mechanisms that nudge producers to abstain could help, but these interventions must be done delicately: if multiple agents abstain, it could lead to correlated failures and even worse utility for users.

Another direction could handle complexities of modeling both user and producer behavior in this case. For example, it may be the case that certain tasks are inherently harder than others, or improvement is non-linear in effort (i.e., more complex scaling laws). It may also be the case that value is heterogeneous or stochastic even within tasks, which could make implementation of weighted winrate more challenging. Additional questions around implementation center on acquiring accurate measurements of values: winrate was selected as a metric because users are often inaccurate at providing real-valued measures of outcomes. If user-provided measures of value are sufficiently expensive or noisy, it may be that other mechanisms could improve upon weighted winrate. Finally, our results generally incentivize producers to specialize. This improves consumer welfare when they can consider a large set of candidate agents, but in the case that users are bandwidth-limited, such specialization may no longer be desirable.

Acknowledgments

This work was supported in part by a METEOR Postdoctoral Fellowship from MIT. We are extremely grateful to Nivasini Ananthakrishnan, Meena Jagadeesan, Brendan Lucier, Ariel Procaccia, Han Shao, Kiran Tomlinson, Serena Wang, Ming Yin, members of Bailey Flanigan & Manish Raghavan's lab meeting, and members of Sanmi Koyejo's AI Measurement Science reading group for invaluable discussions, and to Rachel Li especially for assistance with experiments in Appendix A.2.

References

[1] Rui Ai, Yuqi Pan, David Simchi-Levi, Milind Tambe, and Haifeng Xu. 2026. Beyond majority voting: LLM aggregation by leveraging higher-order information. In *International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 306)*.

[2] Nivasini Ananthakrishnan and Meena Jagadeesan. 2026. Power and Limitations of Aggregation in Compound AI Systems. arXiv:2602.21556 [cs.AI] `https://arxiv.org/abs/2602.21556`

[3] Arena Team. 2026. Introducing Max. `https://arena.ai/blog/introducing-max/`. Accessed: 2026-02-09.

[4] Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, et al. 2024. MT-Bench-101: A Fine-Grained Benchmark for

Evaluating Large Language Models in Multi-Turn Dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 7421–7454.

[5] Suhana Bedi, Iddah Mlauzi, Daniel Shin, Sanmi Koyejo, and Nigam H Shah. 2025. The Optimization Paradox in Clinical AI Multi-Agent Systems. *arXiv preprint arXiv:2506.06574* (2025).

[6] Rishi Bommasani, Kathleen A Creel, Ananya Kumar, Dan Jurafsky, and Percy S Liang. 2022. Picking on the same person: Does algorithmic monoculture lead to outcome homogenization? *Advances in Neural Information Processing Systems* 35 (2022), 3663–3678.

[7] Emile Borel. 1921. La théorie du jeu et les équations intégralesa noyau symétrique. *Comptes rendus de l'Académie des Sciences* 173, 1304-1308 (1921), 58.

[8] Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika* 39, 3/4 (1952), 324–345.

[9] Daniele Caliari and Henrik Petri. 2025. The Luce Model, Regularity, and Choice Overload. *arXiv preprint arXiv:2502.21063* (2025).

[10] Simone Cerreia-Vioglio, David Dillenberger, Pietro Ortoleva, and Gil Riella. 2019. Deliberately stochastic. *American Economic Review* 109, 7 (2019), 2425–2445.

[11] Yatong Chen, Guanhua Zhang, and Moritz Hardt. 2026. Leaderboard Incentives: Model Rankings under Strategic Post-Training. In *International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 306)*.

[12] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*.

[13] Pierre Colombo, Nathan Noiry, Ekhine Irurozki, and Stéphan Clémençon. 2022. What are the best systems? new perspectives on nlp benchmarking. *Advances in neural information processing systems* 35 (2022), 26915–26932.

[14] Ohad Einav and Nir Rosenfeld. 2025. A Market for Accuracy: Classification Under Competition. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025 (Proceedings of Machine Learning Research, Vol. 267)*, Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (Eds.). PMLR / OpenReview.net. `https://proceedings.mlr.press/v267/einav25a.html`

[15] Kawin Ethayarajh and Dan Jurafsky. 2020. Utility is in the eye of the user: A critique of NLP leaderboards. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4846–4853.

[16] Lawrence Friedman. 1958. Game-theory models in the allocation of advertising expenditures. *Operations research* 6, 5 (1958), 699–709.

[17] Shashwat Goel, Joschka Strüber, Ilze Amanda Auzina, Karuna K. Chandra, Ponnurangam Kumaraguru, Douwe Kiela, Ameya Prabhu, Matthias Bethge, and Jonas Geiping. 2025. Great Models Think Alike and this Undermines AI Oversight. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025 (Proceedings of Machine Learning Research, Vol. 267)*, Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (Eds.). PMLR / OpenReview.net. `https://proceedings.mlr.press/v267/goel25b.html`

[18] Polina Gordienko, Christoph Jansen, Julian Rodemann, and Georg Schollmeyer. 2026. Beyond Arrow: From Impossibility to Possibilities in Multi-Criteria Benchmarking. *arXiv preprint arXiv:2602.07593* (2026).

[19] Nika Haghtalab, Nicole Immorlica, Brendan Lucier, and Jack Z. Wang. 2020. Maximizing Welfare with Incentive-Aware Evaluation Mechanisms. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, Christian Bessiere (Ed.). ijcai.org, 160–166. doi:10.24963/IJCAI.2020/23

[20] Jason D Hartline, Liren Shan, Yingkai Li, and Yifan Wu. 2023. Optimal scoring rules for multi-dimensional effort. In *The Thirty Sixth Annual Conference on Learning Theory*. PMLR, 2624–2650.

[21] Chris Hays, Rachel Li, Bailey Flanigan, and Manish Raghavan. 2026. Strategic Candidacy in Generative AI Arenas. In *International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 306)*.

[22] Lu Hong and Scott E Page. 2004. Groups of diverse problem solvers can outperform groups of high-ability problem solvers. *Proceedings of the National Academy of Sciences* 101, 46 (2004), 16385–16389.

[23] Meena Jagadeesan, Michael Jordan, Jacob Steinhardt, and Nika Haghtalab. 2023. Improved bayes risk can yield reduced social welfare under competition. *Advances in Neural Information Processing Systems* 36 (2023), 66940–66952.

[24] Albert X Jiang, Leandro S Marcolino, Ariel D Procaccia, Tuomas Sandholm, Nisarg Shah, and Milind Tambe. 2014. Diverse randomized agents vote to win. *Advances in Neural Information Processing Systems* 27 (2014).

[25] Michael Kearns, Aaron Roth, and Emily Ryu. 2026. Networked Information Aggregation via Machine Learning. In *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 4799–4845.

[26] Elliot Myunghoon Kim, Avi Garg, Kenny Peng, and Nikhil Garg. 2025. Correlated Errors in Large Language Models. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025 (Proceedings of Machine Learning Research, Vol. 267)*, Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (Eds.). PMLR / OpenReview.net. https://proceedings.mlr.press/v267/kim25e.html

[27] Jon Kleinberg and Manish Raghavan. 2021. Algorithmic monoculture and social welfare. *Proceedings of the National Academy of Sciences* 118, 22 (2021), e2018340118.

[28] Dominique Lepelley, Issofa Moyouwou, and Hatem Smaoui. 2018. Monotonicity paradoxes in three-candidate elections using scoring elimination rules. *Social Choice and Welfare* 50, 1 (2018), 1–33.

[29] Hao Li, Yiqun Zhang, Zhaoyan Guo, Chenxu Wang, Shengji Tang, Qiaosheng Zhang, Yang Chen, Biqing Qi, Peng Ye, Lei Bai, Zhen Wang, and Shuyue Hu. 2026. LLMRouterBench: A Massive Benchmark and Unified Framework for LLM Routing. In *Findings of the Association for Computational Linguistics, ACL 2026, San Diego, California, United States, July 2-7, 2026*, Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (Eds.). Association for Computational Linguistics, 37733–37754. doi:10.18653/V1/2026.FINDINGS-ACL.1881

[30] Wenzhe Li, Yong Lin, Mengzhou Xia, and Chi Jin. 2026. Rethinking Mixture-of-Agents: Is Mixing Different Large Language Models Beneficial? *Trans. Mach. Learn. Res.* 2026 (2026). https://openreview.net/forum?id=K6WwK8URlV

[31] Bill Yuchen Lin, Yuntian Deng, Khyathi Raghavi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2025. WildBench: Benchmarking LLMs with Challenging Tasks from Real Users in the Wild. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=MKEHCx25xp

[32] Lydia T Liu, Nikhil Garg, and Christian Borgs. 2022. Strategic ranking. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2489–2518.

[33] Zhuoran Lu, Dakuo Wang, and Ming Yin. 2024. Does more advice help? the effects of second opinions in AI-assisted decision making. *Proceedings of the ACM on Human-Computer Interaction* 8, CSCW1 (2024), 1–31.

[34] R Duncan Luce. 1959. *Individual choice behavior*. Vol. 4. Wiley New York.

[35] Robin L Plackett. 1975. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics* 24, 2 (1975), 193–202.

[36] Ariel D Procaccia, Benjamin Schiffer, Serena Lutong Wang, and Shirley Zhang. [n. d.]. Metritocracy: Representative Metrics for Lite Benchmarks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

[37] Prajit Ramachandran, Barret Zoph, and Quoc V Le. 2017. Searching for activation functions. *arXiv preprint arXiv:1710.05941* (2017).

[38] Baharak Rastegari, Anne Condon, and Kevin Leyton-Brown. 2011. Revenue monotonicity in deterministic, dominant-strategy combinatorial auctions. *Artificial Intelligence* 175, 2 (2011), 441–456.

[39] Mark Rofin, Vladislav Mikhailov, Mikhail Florinsky, Andrey Kravchenko, Tatiana Shavrina, Elena Tutubalina, Daniel Karabekyan, and Ekaterina Artemova. 2023. Vote'n'rank: Revision of benchmarking with social choice theory. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. 670–686.

[40] Kenneth A Small and Harvey S Rosen. 1981. Applied welfare economics with discrete choice models. *Econometrica: Journal of the Econometric Society* (1981), 105–130.

[41] Jiaxin Song, Parnian Shahkar, Kate Donahue, and Bhaskar Ray Chaudhury. 2025. Human-AI Collaboration with Misaligned Preferences. In *Proceedings of the 5th ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization*. 294–294.

[42] Gordon Tullock. 2001. *Efficient Rent Seeking*. Springer US, Boston, MA, 3–16. doi:10.1007/978-1-4757-5055-3_2

[43] Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. 2025. Mixture-of-Agents Enhances Large Language Model Capabilities. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=h0ZfDIrj7T

[44] Xiukun Wei, Min Shi, and Xueru Zhang. [n. d.]. Market Games for Generative Models: Equilibria, Welfare, and Strategic Entry. In *The Fourteenth International Conference on Learning Representations*.

[45] Renzhe Xu, Kang Wang, and Bo Li. 2025. Heterogeneous Data Game: Characterizing the Model Competition Across Multiple Data Sources. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025 (Proceedings of Machine Learning Research, Vol. 267)*, Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (Eds.). PMLR / OpenReview.net. https://proceedings.mlr.press/v267/xu25ad.html

[46] Guanhua Zhang and Moritz Hardt. 2024. Inherent trade-offs between diversity and stability in multi-task benchmarks. In *Proceedings of the 41st International Conference on Machine Learning*. 58984–59002.

[47] Yiqun Zhang, Hao Li, Jianhao Chen, Hangfan Zhang, Peng Ye, Lei Bai, and Shuyue Hu. 2025. Beyond gpt-5: Making llms cheaper and better via performance-efficiency optimized routing. In *Proceedings of the 2025 7th International Conference on Distributed Artificial Intelligence*. 122–129.

A Additional results on model

A.1 Additional related work

Non-monotonicity in aggregation: In Section 4 we demonstrate a "non-monotonicity" result: increasing the value of a model could *decrease* consumer welfare. Related non-monotonicities have been studied in other settings. For example, one close connection is in auctions, where results have shown that adding in more bidders could *decrease* revenue (e.g. [38]). While this phenomenon is very closely related, the types of aggregation mechanisms and objectives differ between our settings and auctions. Related non-monotonicities also occur in social choices, such as settings where it could be the case that *more* votes for a candidate could cause her to go from winning a race to losing (e.g. [28]). This could reflect violations of Pareto efficiency or the unanimity principle, which would declare that more support for an option should not *reduce* its property of winning. One close connection is [40], which shows that consumer welfare is monotone with a different utility model where users derive utility according to the value plus the additive ϵ noise: however, in our setting, we argue that it makes more sense to use the noise terms ϵ_i to model decision errors, rather than true variation in quality that a user perfectly responds to. Our results differ from most voting settings in that we assume there is some true value over candidates (here, model responses) and we are studying the "expected value" of the election, while most voting mechanisms are candidate-neutral and focus on the probability of individual candidates being selected.

We note that our "monotonicity" condition is *not* the same as "regularity" (e.g. [10, 9]), a different property often studied in relation to choice models. Regularity is defined as saying that the probability of an item being picked only stays constant or decreases as more items are added (we refer to this property as "substitutability", in Definition 9). A choice function could exhibit monotonicity or regularity, or neither, or both: regularity is a property of the probability of an item being picked (given other items added to the set), and monotonicity is a property of the value that the choice function gives the user (as items in the set increase in value).

Connection to Colonel Blotto, Tullock contests: Note that the "model-creation" setting in Section 5 has close ties to a rich vein of literature for how agents compete, such as Colonel Blotto and Tullock contests. The Colonel Blotto game models a pair of Colonels competing over a series of "battlefields", with some fixed total amount of value, and an agent "winning" a battlefield according to some cost function based on how strongly they out-compete the other Colonel [7]. Tullock contests similarly study different agents who allocate different value in the hope of winning a probabilistic *contest*, where the aggregation function is typically given by $\frac{v_a}{v_a+v_b}$ [42]. The closest variant of Colonel Blotto is the "Lottery Blotto" game [16], which studies Colonel Blotto where the outcome is determined by a Tullock contest function. Results here show that the unique equilibrium is to have players have allocation that is equal across tasks (if all tasks are equally valuable). While these literatures differ from our setting in multiple ways, one crucial one is the function governing "winrate." Specifically, winrate is often either deterministic (by whichever agent has the larger value), or when probabilistic is often given by convex functions, such as $\frac{v_{ai}}{v_{ai}+v_{bi}}$. Given a function like this, agents are generally incentivized to best-respond by evenly spreading value across tasks. However, in our setting we use BTL to determine winrate, which is *not* convex. This leads to much more complex patterns of behavior. We view this complexity as a benefit, given that it arises from satisfying natural properties like users becoming more easily "distracted" by items that are similar in value, discussed in Section 2.

A.2 Error patterns in routing

In this section we include further explorations of empirical patterns in "errors" in LLM routing. As we noted in Section 3, LLM routing is an active area of research, and many contain substantially different approaches. One key difference is that the source of randomness in LLM routing is different. When humans make choices, the randomness in their decision often comes from inherent uncertainty or variability in human behavior, or "irrelevant" features. However, many LLM routing tools are deterministic at the instance-level decision of where to route any particular query, given a fixed algorithm, but have randomness involved in the training process of the algorithm. In this subsection we will explore how such randomness relates to the model in Section 2 and our paper's general implications.

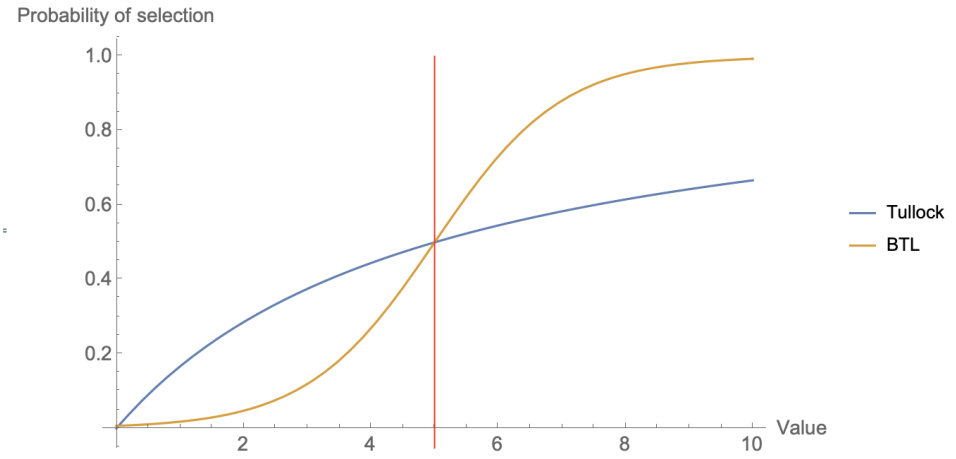


Figure 2: Given a setting with two players with values v_i, v_j, the probability of selection of v_i, given $v_j = 5$ (red vertical line) for Tullock contest function $\frac{v_i}{v_i+v_j}$ and BTL $\frac{\exp(v_i/\beta)}{\exp(v_i/\beta)+\exp(v_j/\beta)}$ with $\beta = 1$. Note that the probability for BTL changes most quickly around $v_i \approx v_j$.

Specifically, we recreate the results in [47], which is a clustering-based routing method. In this approach, the router is trained on multiple benchmark datasets. A randomized 70%/30% split of data is used for training and test performance. In the training process, the data is grouped into 64 clusters. In the inference stage, for each new query the routing algorithm calculates which of the 64 clusters the query is closest to in embedding space, and then deterministically routes the query to whichever model performs best on that cluster.

In Table 1 we include the empirical performance of each model on each benchmark (as calculated in our 70% split), and in Table 2 we return the proportion of test queries (in the 30% split) from that dataset that were routed to each model. Recall that because items are rated based on clusters that imperfectly overlap with benchmarks, the routing tool will *not* always route the query from a particular benchmark. For example, the routing tool maps queries from the AIME dataset to `deepseek-r1-0528` always, even though its training performance on this dataset is only tied for second best (`a22b-thinking-2507` has slightly higher performance, 0.93 as compared to 0.9). However, there is a general connection between strength on a benchmark and frequency with which that model is used on that benchmark.

Figure 3 presents the connections between these. In general, the routing tool is more likely to make "mistakes" in routing between two models that have very similar performance. When a model has substantially lower performance than the best model on a task, its probability of being routed to drops substantially. Note that this pattern of errors does not exactly match those given by BTL - which is at least partially due to fact that this experiment only captures some source of randomness in routing and not others (such as the randomness in the test/train split). However, from this experiment, routers do seem to display many of the key features we referenced in Section 2: they are better than random but worse than perfect, and tend to more frequently make errors between options that are closer in value. Given this, we suspect that the high-level insights from our paper could also apply to routing tools.

A.3 Experiments with benchmark datasets (additional plots from Section 7)

Here, we include additional figures for Section 7.

A.3.1 LLMRouterBench

Figure 4 [29] displays the performance of different models on benchmarks, and is the information used to create Figure 1. Table 3 displays the allocation of value across tasks in the experiment in Figure 1c: firms would respond to objectives (winrate, weighted winrate, consumer welfare) in different ways, which result in the consumer welfare outcomes displayed in Figure 1c.

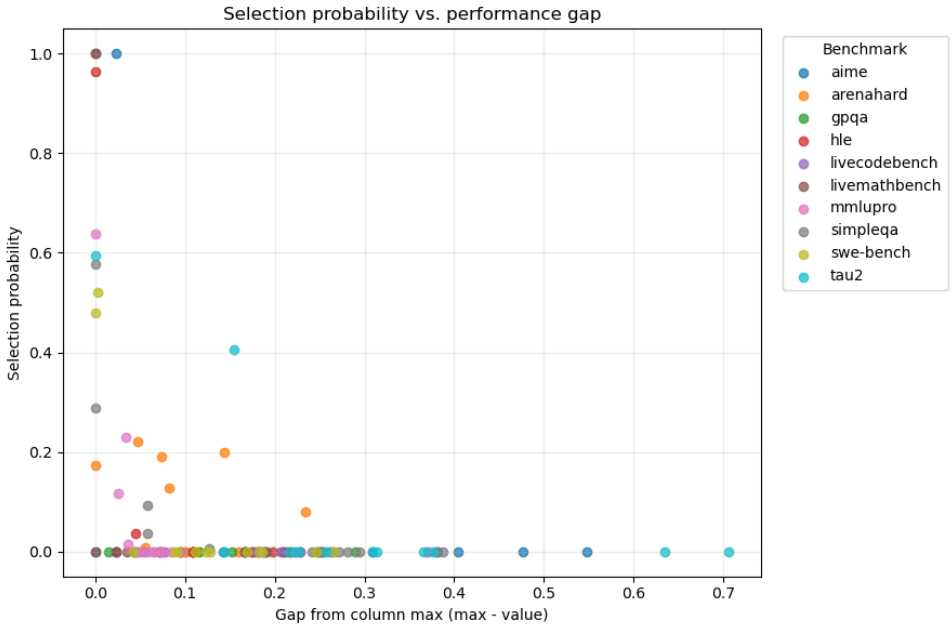


Figure 3: For each benchmark and each model, the x-axis gives the gap between that model's performance on that benchmark (e.g. a gap of 0 means that model is the best), and the y axis measures how frequently that model is selected by the router for that benchmark.

model	aime	arena-hard	gpqa	hle	live-codebench	live-math-bench	mmlupro	sim-pleqa	swe-bench	tau2
claude-sonnet-4	0.38	0.53	0.66	0.05	0.61	0.63	0.83	0.16	0.34	0.48
deepseek-r1-0528	0.9	0.62	0.8	0.16	0.82	0.8	0.84	0.26	0.3	0.39
deepseek-v3-0324	0.45	0.61	0.56	0.04	0.69	0.63	0.78	0.28	0.25	0.07
deepseek-v3.1-terminus	0.55	0.67	0.75	0.09	0.68	0.76	0.84	0.24	0.25	0.49
gemini-2.5-flash	0.67	0.55	0.7	0.08	0.62	0.77	0.81	0.3	0.17	0.34
gemini-2.5-pro	0.88	0.77	0.85	0.22	0.8	0.55	0.85	0.54	0.34	0.45
glm-4.6	0.79	0.65	0.73	0.16	0.65	0.63	0.8	0.26	0.21	0.56
gpt-5	0.9	0.69	0.83	0.26	0.87	0.8	0.87	0.48	0.16	0.71
gpt-5-chat	0.71	0.68	0.74	0.06	0.64	0.61	0.81	0.41	0.1	0
intern-s1	0.52	0.67	0.64	0.08	0.5	0.73	0.82	0.15	0.08	0.33
kimi-k2-0905	0.62	0.7	0.74	0.07	0.7	0.7	0.8	0.28	0.23	0.48
qwen3-235b-a22b-2507	0.76	0.71	0.6	0.09	0.67	0.77	0.82	0.54	0.16	0.4
qwen3-235b-a22b-thinking-2507	0.93	0.72	0.8	0.08	0.8	0.55	0.79	0.48	0.22	0.55

Table 1: Performance of models on different benchmarks (reproduction of [47], 70% split of data for training).

model	aime	arena-hard	gpqa	hle	live-codebench	live-math-bench	mmlupro	sim-pleqa	swe-bench	tau2
claude-sonnet-4	0	0.08	0	0	0	0	0	0	0.48	0
deepseek-r1-0528	1	0.2	0	0	0	1	0.23	0	0	0
deepseek-v3-0324	0	0	0	0	0	0	0	0	0	0
deepseek-v3.1-terminus	0	0	0	0	0	0	0.01	0	0	0
gemini-2.5-flash	0	0	0	0	0	0	0	0	0	0
gemini-2.5-pro	0	0.17	1	0.04	0	0	0.12	0.58	0.52	0
glm-4.6	0	0	0	0	0	0	0	0	0	0
gpt-5	0	0.19	0	0.96	1	0	0.64	0.09	0	0.6
gpt-5-chat	0	0.13	0	0	0	0	0	0.01	0	0
intern-s1	0	0	0	0	0	0	0	0	0	0
kimi-k2-0905	0	0	0	0	0	0	0	0	0	0
qwen3-235b-a22b-2507	0	0.01	0	0	0	0	0	0.29	0	0
qwen3-235b-a22b-thinking-2507	0	0.22	0	0	0	0	0	0.04	0	0.4

Table 2: Empirical routing proportion (30% test split), reproduction of results in [47].

Model	Mathematics			Code			Logical			Knowledge				Affective	
	AIME	M500.	MBen.	MBPP	HE.	LCB.	KOR.	K&K.	BBH	MP.	GPQA	FQA.	MQA.	Emory.	MELD
DH-Llama3-it	0.00	29.80	38.67	54.00	51.22	13.08	31.92	15.57	49.07	36.66	27.78	49.96	57.58	38.16	47.16
DS-Qwen3	68.33	94.20	90.00	35.01	25.00	64.74	56.00	60.86	89.44	70.83	60.61	70.36	81.62	38.74	51.06
DS-Qwen	40.00	88.40	88.00	53.59	45.12	41.14	45.36	50.86	69.17	47.75	35.86	63.30	34.80	27.55	34.90
Fin-R1	11.67	75.40	66.67	68.99	77.44	6.82	33.20	19.00	61.57	48.65	27.27	68.70	63.24	34.72	50.41
GLM-Z1	61.67	94.60	95.33	62.94	60.37	61.52	53.84	47.57	84.26	67.33	56.06	68.70	71.01	35.01	45.86
Intern-S1-mini	76.67	91.40	64.67	44.05	43.29	26.26	58.16	79.43	83.06	56.04	48.99	70.36	68.42	36.87	50.41
Llama-3.1-it	8.33	49.80	46.67	60.68	61.59	16.78	24.16	13.71	58.70	46.65	26.26	52.92	68.26	33.72	48.05
UltraMedical	0.00	42.40	25.33	53.90	56.10	13.36	14.32	15.43	38.06	40.26	23.23	54.58	68.50	31.71	42.21
Llama-Nemo	45.00	90.80	50.00	61.50	62.80	46.07	29.52	22.86	26.20	41.46	33.84	31.12	41.48	29.41	38.80
MiMo-RL	0.00	26.60	11.33	56.78	52.44	6.45	29.84	4.00	29.91	20.78	4.04	17.00	29.38	28.12	39.69
MiniCPM	68.33	93.40	95.33	34.09	32.93	63.70	30.96	62.71	88.06	69.93	52.02	67.22	65.36	35.72	51.30
NVIDIA-Nemo	70.00	93.60	94.67	42.09	48.17	65.50	56.80	70.71	86.48	70.13	52.02	71.23	74.71	39.02	46.75
OpenThinker	41.67	85.60	62.67	11.81	7.32	44.17	11.04	17.29	42.50	42.06	31.82	61.55	52.08	22.81	33.44
Qwen-Coder	1.67	65.20	67.33	75.98	77.44	27.87	34.24	20.43	57.04	45.35	32.32	63.21	48.78	37.02	52.60
Qwen3-8B	76.67	93.60	95.33	54.72	62.20	67.68	53.92	77.00	83.98	69.13	57.58	74.46	79.18	38.74	54.22
Cogito-v1	1.67	51.20	56.00	51.95	69.51	17.25	42.96	24.14	69.44	57.04	35.35	61.64	65.75	37.45	54.95
Gemma-2-it	1.67	48.20	54.67	62.42	64.63	17.25	34.64	9.71	60.74	52.95	27.78	64.60	63.79	39.74	52.60
Glm-4-chat	1.67	49.80	49.33	62.83	74.39	16.87	37.28	12.43	46.94	47.05	23.23	57.28	62.14	39.02	55.28
Granite-3.3-it	6.67	69.00	59.33	36.55	51.22	14.22	32.00	21.71	31.39	44.66	31.82	62.34	61.19	36.59	50.32
Internlm3-it	6.67	69.00	60.00	60.47	65.85	21.23	33.84	25.86	61.94	55.14	33.84	61.99	67.24	36.87	50.97

Table 6: The performance of each model on each dataset under the Routing for Performance setting. The deep red and light red markers denote the best and second-best results, respectively.

Figure 4: Performance of different models on LLMRouterbench [29].

	Winrate	Weighted winrate	Consumer welfare
AIME	45	100	100
M500	89	0	∅
MBen	66	0	∅
MBPP	60	0	∅
HE	60	0	∅
LCB	48	100	100
KOR	36	100	100
K&K	25	100	100
BBH	53	100	100
MP	48	100	100
GPQA	36	100	100
FQA	66	0	∅
MQA	72	0	∅
Emory	42	100	100
MELD	54	0	∅

Table 3: Given a total allocation of value 800 (average 53 across tasks), the allocation that maximizes a) winrate, b) weighted winrate, or c) consumer welfare, with $\beta = 1$. Winrate maintains a constant 0.97 winrate (except for in tasks where that isn't possible due to a max value cap of 100), and weighted winrate and consumer welfare are almost identical, except that consumer welfare abstains where weighted winrate returns an item of value 0.

Model		Perceptivity					Adaptability						Interactivity	
		Memory	Understanding		Interference		Rephrasing		Reflection		Reasoning		Questioning	
	Avg.	CM	SI	AR	TS	CC	CR	FR	SC	SA	MR	GR	IC	PI
Llama2-7B-Chat	6.53	7.64	6.21	7.92	8.23	8.50	8.32	8.56	8.45	4.97	1.88	3.83	5.23	5.11
Qwen-7B-Chat	7.12	7.65	7.75	8.73	8.42	8.76	8.89	9.16	8.49	7.28	2.25	3.57	5.41	6.24
ChatGLM2-6B	5.56	6.14	4.69	7.27	6.13	6.26	7.47	7.98	6.97	4.19	2.11	3.00	5.16	4.90
ChatGLM3-6B	6.47	7.16	5.42	8.21	7.43	8.03	8.38	8.81	7.40	5.63	2.60	3.21	6.19	5.61
InternLM2-Chat-7B-SFT	6.69	7.51	6.26	8.01	8.06	8.70	8.50	8.50	7.68	6.16	3.47	4.48	4.92	4.76
Yi-6B-Chat	6.93	7.57	5.27	8.69	8.37	8.76	8.43	8.44	7.49	7.85	2.18	3.80	7.30	6.00
Mistral-7B-Instruct-v0.2	6.95	7.66	5.64	8.09	8.30	9.35	8.69	8.59	8.16	7.33	2.58	4.52	5.80	5.66
Vicuna-13B-v1.5	6.37	7.06	5.62	7.81	7.45	8.79	7.96	7.72	7.47	6.70	2.31	4.03	5.05	4.80
Baize-13B-v2	6.12	6.78	5.15	7.86	7.40	8.07	7.96	8.15	7.24	6.32	1.67	3.69	4.35	4.95
UltraLM-13B-v2.0	4.61	4.66	4.89	5.99	6.49	8.48	2.87	2.53	6.70	5.27	1.46	2.34	4.13	4.11
Llama2-13B-Chat	7.15	8.03	7.11	9.00	9.39	8.81	9.07	9.11	7.63	7.60	1.75	3.16	6.07	6.23
Qwen-14B-Chat	7.82	8.33	8.36	9.04	9.22	9.50	9.12	9.39	8.41	7.97	3.50	4.55	8.21	6.12
Baichuan2-13B-Chat	7.00	7.71	6.38	8.92	8.36	9.07	9.10	8.95	7.75	6.57	2.50	3.65	6.95	5.15
InternLM2-Chat-20B-SFT	6.95	7.35	6.44	8.08	8.05	9.10	8.59	8.55	7.62	7.36	4.05	5.24	4.99	4.99
Yi-34B-Chat	8.10	8.55	6.79	9.34	9.84	9.34	9.08	9.38	9.01	9.04	4.07	5.90	8.51	6.39
Mixtral-8x7B-Instruct-v0.1	7.38	7.86	5.94	8.49	9.01	9.52	8.91	9.01	8.69	7.78	4.19	5.14	6.03	5.36
GPT-3.5	7.99	8.77	7.67	7.67	9.68	9.87	9.56	9.51	9.18	7.23	4.48	5.31	8.57	6.32
GPT-4	8.86	8.88	8.99	9.58	9.83	9.98	9.54	9.57	9.36	9.52	7.15	7.17	9.00	6.64
Avg.	6.92	7.52	6.37	8.26	7.72	8.24	8.36	8.44	7.98	6.93	3.61	4.84	6.22	5.52

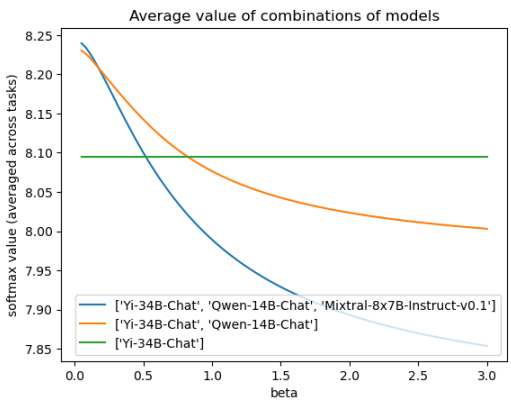
Figure 5: Performance of different models on MT-Bench-101 [4]

A.3.2 MT-Bench-101

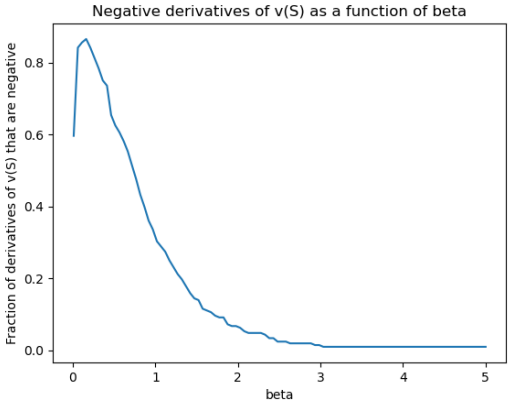
Figure 5 displays MT-Bench-101 [4] performance across tasks. Table 4 gives the allocations that lead to welfare changes in Figure 6c.

A.3.3 Wildbench

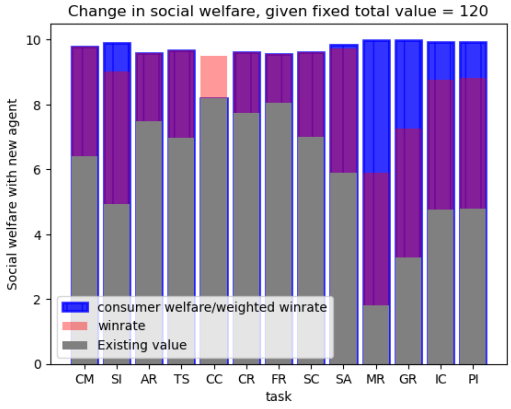
Table 5 displays performance of the top 10 models on Wildbench task-specific benchmarks [31]. Figure 7 displays all figures related to experiments on Wildbench data, and Table 6 gives the allocations that lead to welfare changes in Figure 7c.



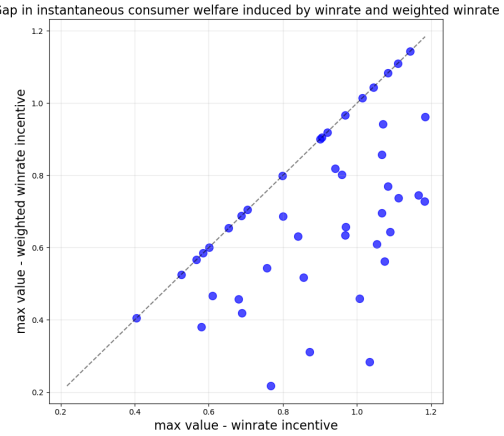
(a) Value of combinations of models.



(b) Fraction of derivatives of consumer welfare that are negative.



(c) Change in consumer welfare given different incentives for model creation.



(d) Gap in instantaneous consumer welfare given different incentives in model improvement.

Figure 6: Experiments with MTBench-101 data [4]

	CM	SI	AR	TS	CC	CR	FR	SC	SA	MR	GR	IC	PI
Winrate	10	9.2	10	10	10	10	10	10	9.88	6.06	7.43	8.93	9
Weighted winrate	10	10	10	10	0	10	10	10	10	10	10	10	10
Consumer welfare	10	10	10	10	$\emptyset$	10	10	10	10	10	10	10	10

Table 4: Given a total allocation of value 120 (average 9.23 across tasks), the allocation that maximizes a) winrate, b) weighted winrate, or c) consumer welfare, with $\beta = 1$. Winrate maintains a constant 0.96 winrate (except for in tasks where that isn't possible due to a max value cap of 10), and weighted winrate and consumer welfare are almost identical, except that consumer welfare abstains for the smallest task, while weighted winrate does not. For space constraints, the task names are abbreviated.

	Creative Tasks	Coding & Debugging	Planning & Reasoning	Information/Advice seeking	Math & Data Analysis
Athene-70B	7.9	7.76	7.96	7.94	7.58
gpt-4o-2024-05-13	7.78	7.91	7.89	7.73	7.59
gpt-4o-mini-2024-07-18	7.87	7.58	7.69	7.61	7.27
Mistral-Large-2	7.75	7.25	7.59	7.6	7.13
yi-large-preview	7.63	7.29	7.53	7.64	7.06
gpt-4-turbo-2024-04-09	7.73	7.37	7.48	7.58	6.96
claude-3-5-sonnet-20240620	7.43	7.52	7.43	7.42	6.88
deepseek-v2-chat-0628	7.51	7.36	7.35	7.14	7.01
gemma-2-9b-it-DPO	7.77	6.92	7.41	7.69	6.58
gemma-2-9b-it-SimPO	7.66	6.95	7.43	7.51	6.72

Table 5: Performance of models on different tasks in Wildbench benchmark [31], retrieved from `https://github.com/allenai/WildBench/blob/main/leaderboard/data_dir/score.json`. For concisesness, only top 10 models out of 61 are presented here, but all were used in analysis for Figure 7 unless otherwise noted.

	Creative Tasks	Coding & Debugging	Planning & Reasoning	Information/Advice seeking	Math & Data Analysis
Winrate	7.46	4.69	6.14	7.07	4.35
Weighted winrate	0	10	10	0	10
Consumer welfare	$\emptyset$	10	10	$\emptyset$	10

Table 6: Allocation of values for Figure 7c.

B Experimental generalizations

B.1 Beyond Bradley-Terry-Luce [8, 35, 34]

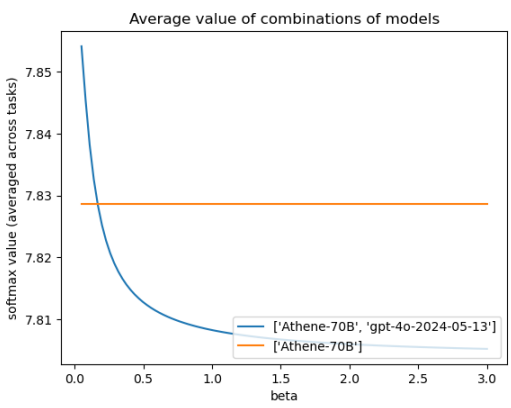
Our theoretical results assume BTL aggregation. Here we check whether the model creation experiment in Figure 1c depends on this assumption. We repeat it under three other choice functions.

Choice models. For each task t, the user picks model m with probability $p_m(\vec{v}_t)$ given by one of:

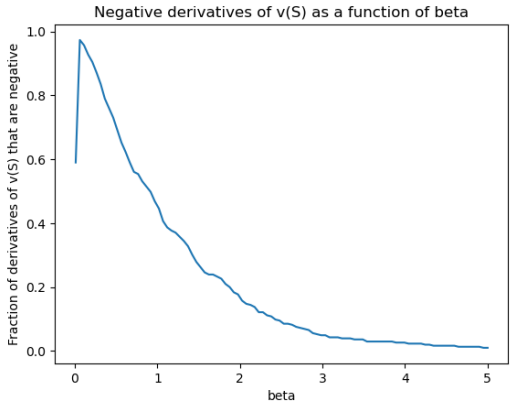
- *BTL* (Definition 1) with $\beta = 5$.
- *Thurstone (probit)*: the user observes each value with independent Gaussian noise and picks the highest. That is, $p_m(\vec{v}_t) = \Pr\left[m = \text{argmax}_{m'} v_{m',t} + \sigma Z_{m'}\right]$ with $Z_{m'} \sim \mathcal{N}(0,1)$ i.i.d. We compute these probabilities by numerical integration.
- *δ-cutoff*: the user picks uniformly at random among all models within δ of the best value on the task, $\{m : v_{m,t} \geq \max_{m'} v_{m',t} - \delta\}$.
- *Top-k shortlist*: the user picks uniformly at random among the k highest-valued models on the task. Ties at the boundary of the shortlist are broken at random. Top-1 is optimal aggregation.

Models that abstain ($v_{m,t} = \emptyset$) are never picked under any of these choice functions.

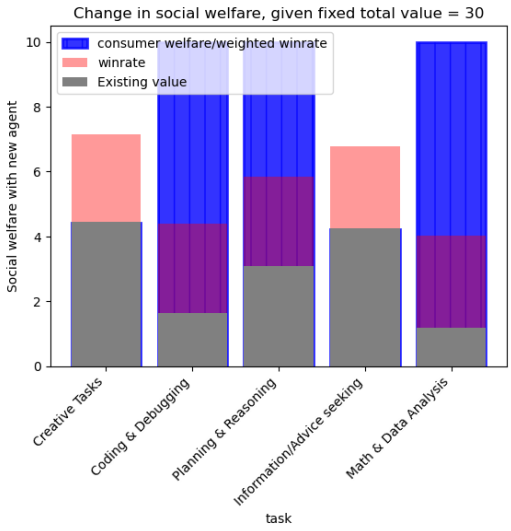
Parameter choice. We choose σ and δ so that each choice function is about as noisy as BTL with $\beta = 5$. For every pair of models in LLMRouterBench and every task, we treat the pair as a two-model market and compute the probability that the lower-valued model is picked. Averaged over all 190 pairs and 15 tasks, this probability is 0.22 under BTL with $\beta = 5$. Matching this average gives $\sigma = 11.89$ and $\delta = 10.69$. Top-k cannot be matched in this way: in a two-model market, top-1 never picks the lower-valued model and top-2 picks at random. We therefore report top-2 as an example of a much noisier shortlist. In the market of the 5 weakest models, top-2 fails to pick the best model 47% of the time on average, compared to 36% under BTL with $\beta = 5$.



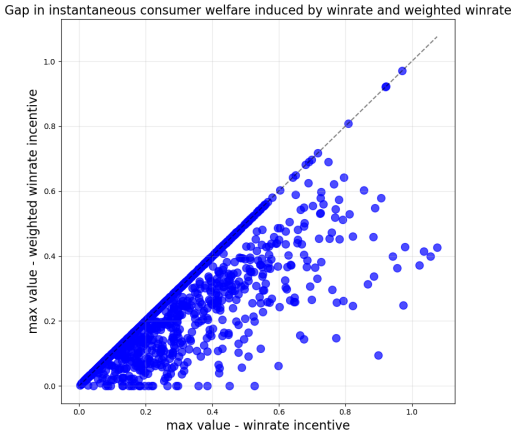
(a) Value of combinations of models.



(b) Fraction of derivatives of consumer welfare that are negative.



(c) Change in consumer welfare given different incentives for model creation.



(d) Gap in instantaneous consumer welfare given different incentives in model improvement.

Figure 7: Experiments with Wildbench data [31]

Best responses. We use the same setting as Figure 1c: the existing market is the 5 weakest models, the incoming model has total value $V = 800$, and each task's value is capped at 100. The BTL-specific characterizations in Section 5 do not apply to these choice functions, so we compute each objective's best response directly. Each objective (winrate, weighted winrate, consumer welfare) is a sum of per-task terms. The best response is therefore a knapsack problem over tasks, which we solve exactly by dynamic programming over values discretized in steps of 0.1, subject to $\sum_t v_t \leq V$. As in the main text, a producer maximizing consumer welfare may abstain on a task, while producers maximizing winrate or weighted winrate return a value of 0 instead.

Under the δ-cutoff and top-k rules, p_m is piecewise constant in $v_{m,t}$. As a result, an objective often has many allocations that are all optimal, and these allocations can differ in consumer welfare. In those cases we report both the best and the worst consumer welfare among the optimal allocations. No such ties arise under BTL or Thurstone.

Results. Figure 8 shows that Thurstone aggregation gives essentially the same result as BTL. Winrate incentivizes homogenization, while weighted winrate and consumer welfare both specialize on the same 8 tasks. The total change in consumer welfare is 344.0 for winrate and 519.7 for both weighted winrate and consumer welfare, compared to 335.4 and 523.4 under BTL.

The hard-threshold rules behave differently (Figures 9 and 10). Under the δ-cutoff, once the incoming model beats every existing model by more than δ on a task, it is picked with probability 1, so its weighted winrate on that task is exactly v_t. Weighted winrate is therefore indifferent between two very different allocations, since both give weighted winrate V:

- winning every task by a small margin, which is essentially the winrate allocation, with a consumer welfare gain of 170.4;
- putting value 100 on the 8 tasks where existing models are weakest, which is the welfare-optimal allocation, with a gain of 517.3.

The same happens under top-2: once the incoming model is in the shortlist, it is picked with probability exactly $1/2$, and weighted winrate gives consumer welfare gains between 86.1 and 288.5. Weighted winrate's preference for specialization in our theoretical results relies on p_m strictly increasing in value, which rewards a larger margin. Hard thresholds remove this reward. Even so, under the same tie-breaking rule weighted winrate is at least as good as winrate in every case we consider. Winrate's gains range from 0.4 to 86.1 under top-2. In the worst case, weighted winrate matches winrate's best case.

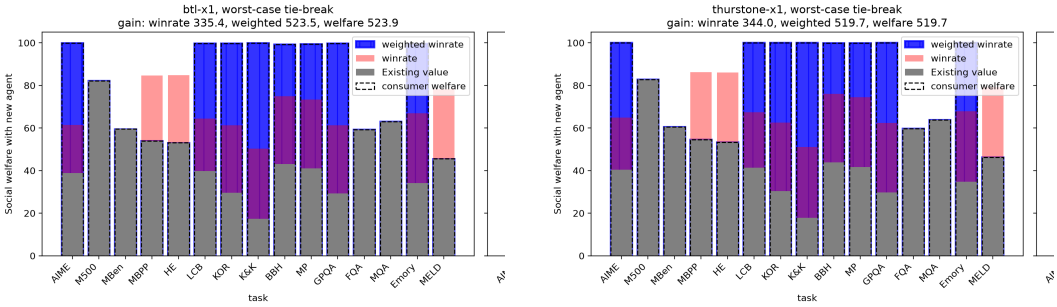


Figure 8: Consumer welfare given different incentives for model creation, under BTL (left, as in Figure 1c) and Thurstone aggregation (right). Gray shows the existing value, red the welfare induced by winrate incentives, and blue by weighted winrate incentives. The dashed outline shows the welfare-optimal allocation.

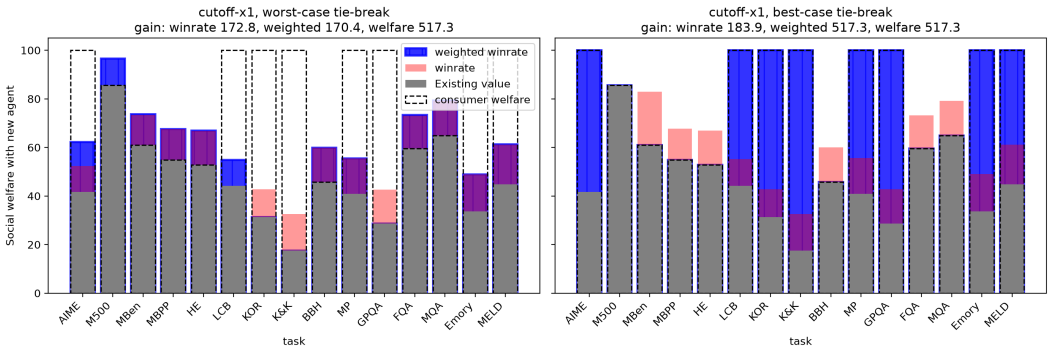


Figure 9: As in Figure 8, under the δ-cutoff choice function. Among allocations that are optimal for each objective, the left panel shows the one with the lowest consumer welfare and the right panel the one with the highest.

B.2 Repeated best-response

Section 5 studies a single incoming producer. Here we ask what happens when producers enter sequentially: each new producer best responds to the current market (including all earlier entrants) and then joins it. Every producer in a trajectory optimizes the same objective (winrate, weighted winrate, or consumer welfare), and aggregation is BTL with $\beta = 1$. For computational reasons, rather

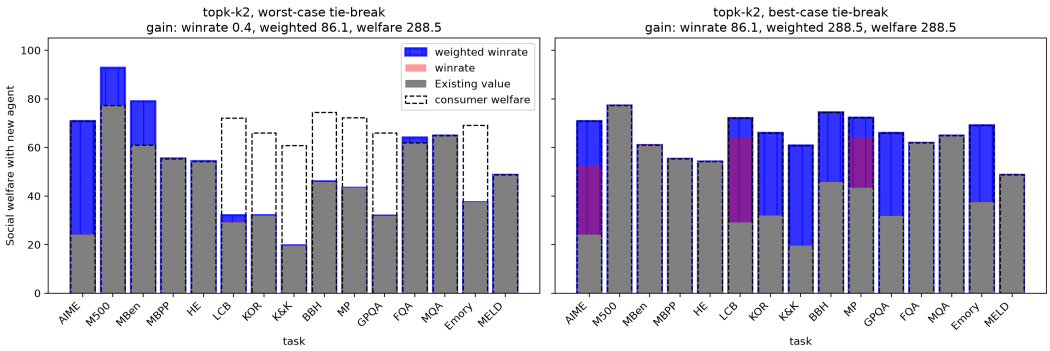


Figure 10: As in Figure 9, under the top-2 shortlist choice function.

than exactly solve best response, we compute each entrant's best response on a value grid of step 0.5, by dynamic programming over the budget $\sum_t v_t \le V$ with $0 \le v_t \le 100$; as in Section 5, entrants optimizing weighted winrate return value 0 rather than abstaining, and only the consumer-welfare entrant may abstain ($\emptyset$). Ties are broken against consumer welfare.

First, we use synthetic markets rather than the benchmarks of Section 7, which has $T = 40$ tasks, each capped at value 100, and an existing market of 3 models with values drawn i.i.d. from $U[0, 20]$. Each entrant has total value $V = 1800$, so the condition of Theorem 3 holds for the first entrant ($2T\beta \max_j \log \sum_i \exp(\vec{v}_{ji}/\beta) \le 1688 < V$): recall that this condition indicates when an entering player optimizing for winrate would be incentivized to *homogenize*. When this condition fails, players still may be incentivized to homogenize, or may pick some other allocation. We run 6 entrants over 10 draws of the initial market.

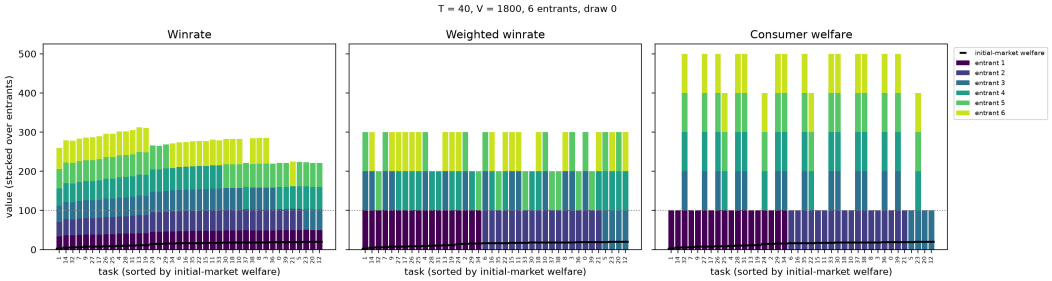
Figure 11 shows the results: the top figure shows the *allocation* that each player entering picks, while the bottom figure shows consumer utility. Visually looking at the *allocation* figure on the top, note that with winrate, players tend to spread out their allocation substantially more (bands of the same color tend to be wide and short), while with weighted winrate or consumer welfare objectives, players tend to specialize more (bands of the same color that are tall and narrow). Looking at the *consumer welfare* figure on the bottom, we can see that weighted winrate and consumer welfare objectives on the right have maximized consumer welfare - the colored bars go all the way to the top. However, for a market with incentives given by winrate, consumer welfare still has very large gaps.

Next, we generalize this to a real dataset: LLMRouterBench. Figure 12 begins with the same starting point as the one-shot response in Section 7: a market is composed of the 5 weakest models, and a new firm enters and creates a new model. Figure 12 shows a very similar pattern to Figure 11: weighted winrate and consumer value incentivize greater specialization than winrate does, which results in higher consumer welfare.

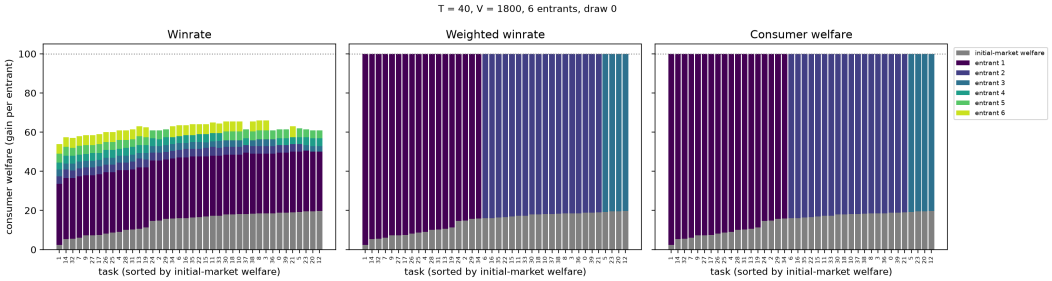
B.3 Model improvement with $N > 2$ agents

Our results in Section 6 are stated for $N = 2$ producers. Here we explore through simulation whether they extend to larger markets. We draw synthetic markets with $N \in \{2, 3, 5, 10\}$ producers and $T \in \{N, 20\}$ tasks, with values in $[0, 100]$ drawn from one of three distributions: i.i.d. uniform, "specialized" (each model is weak everywhere except on $\max(1, \lfloor T/N \rfloor)$ tasks), and "correlated" (value is a model-level quality plus a task-level effect plus small noise). For each market and each $\beta \in \{3, 5, 10, 20, 50\}$, every producer computes the derivative of its objective (winrate, weighted winrate, or consumer welfare) with respect to its own value on each task, as in Lemma 17, and improves the task with the largest derivative. For N producers, the derivative of consumer welfare is $p_i(\vec{v}) \left(1 + \sum_j (\vec{v}_i - \vec{v}_j) p_j(\vec{v})/\beta\right)$, which reduces to the expression in Lemma 17 when $N = 2$. We use 1000 markets per configuration with fixed random seeds.

Homogenization. Figure 13 shows the fraction of producer pairs that pick the same task to improve. At $N = 2$ this recovers Theorem 5 exactly: under winrate both producers pick the same task in every market, and under consumer welfare they never do. For $N > 2$, incentives for homogenization

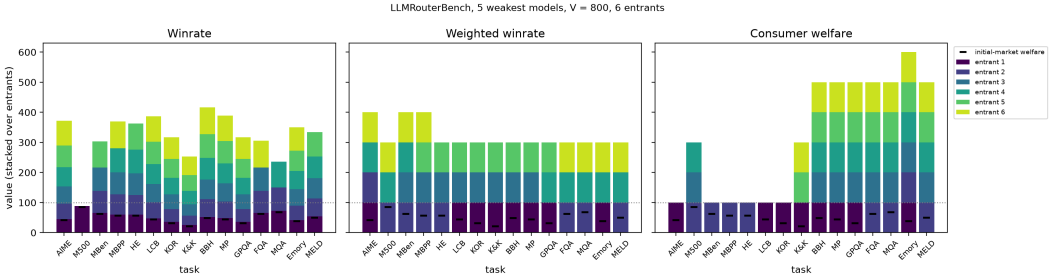


(a) Allocations of value (e.g. model strength) across different tasks.

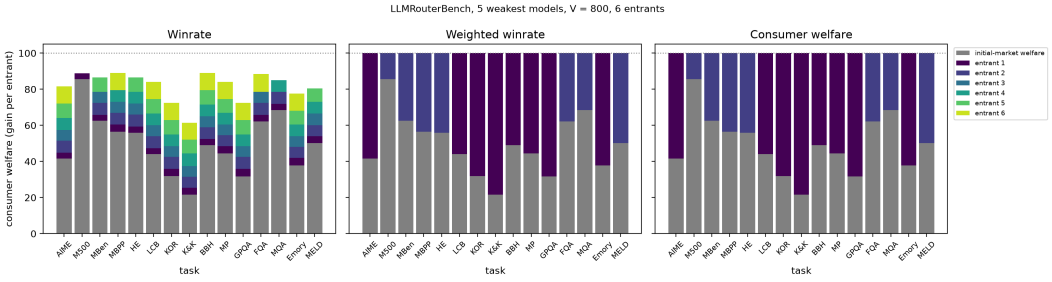


(b) Consumer welfare across different tasks.

Figure 11: Repeated best-response with synthetic dataset



(a) Allocations of value (e.g. model strength) across different tasks.



(b) Consumer welfare across different tasks.

Figure 12: Repeated best-response with LLM RouterBench dataset [29]

under winrate and weighted winrate tend to become much closer, and both converge towards the $1/T$ baseline.

Instantaneous consumer welfare. The previous section covered *homogenization*: here, we study consumer welfare. Instantaneous consumer welfare (Definition 11) is separable across producers, so the welfare-optimal profile is the one chosen under the consumer-welfare objective. Figure 14 reports how often weighted winrate yields strictly *lower* instantaneous consumer welfare than winrate. Consistent with Lemma 8, this does happen: in approximately 20% of markets with i.i.d. uniform starting points, between 0 and 40% of markets that are specialized, and for a much larger fraction of markets when firms begin with correlated strengths. The final figure shows a scatter plot of the gap in magnitude of incentives produced by winrate and weighted winrate, showing that even for markets that produce worse incentives under weighted winrate, the gap is usually very small in magnitude.

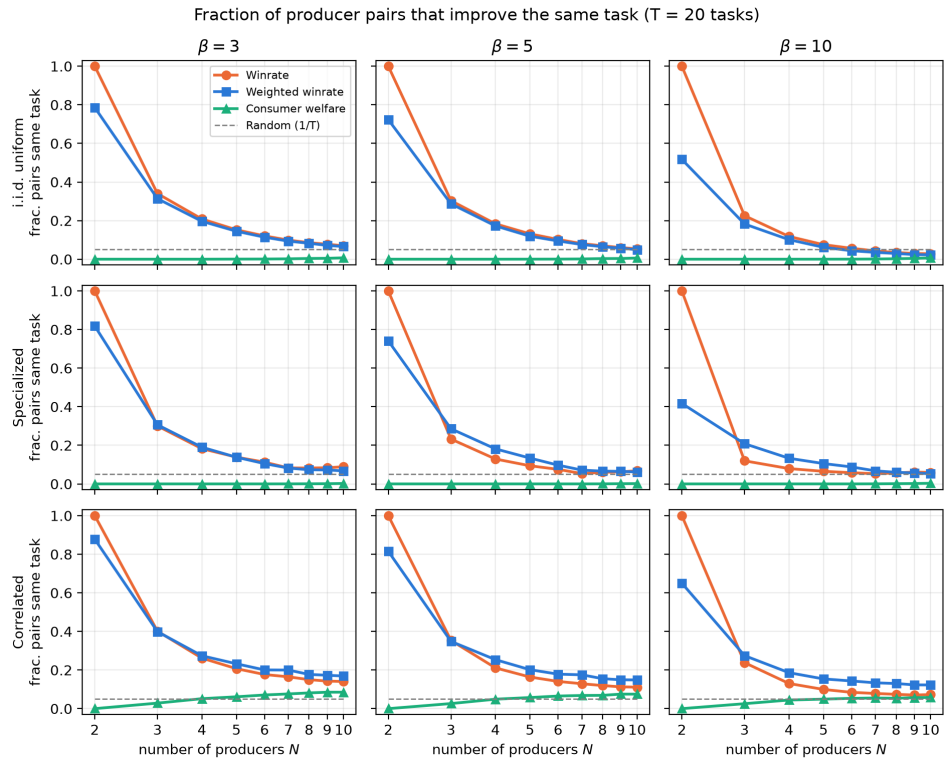


Figure 13: Fraction of producer pairs that pick the same task to improve, as a function of the number of producers N, with $T = 20$ tasks. Rows give the value distribution and columns give β. The dashed gray line is the rate $1/T$ if producers picked tasks uniformly at random. At $N = 2$, winrate always leads to the same task and consumer welfare never does (Theorem 5).

C Proofs for Section 4

Lemma 9. *Adding a model B to a model A increases consumer welfare with BTL aggregation if and only if*

$$0 \leq \sum_{t \in [T]} \frac{\exp((v_{t,b} - v_{t,a})/\beta)}{\exp((v_{t,b} - v_{t,a})/\beta) + 1} \cdot (v_{t,b} - v_{t,a}) \tag{1}$$

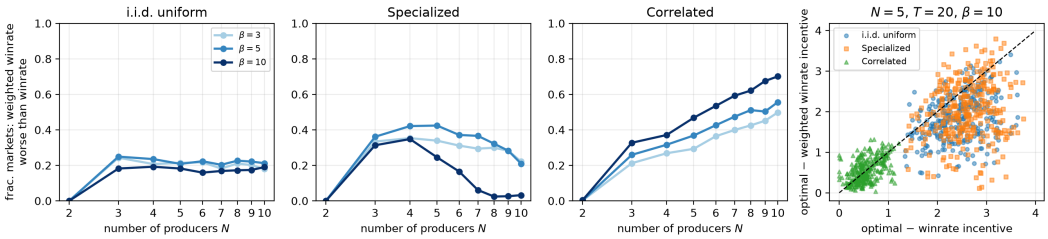


Figure 14: Instantaneous consumer welfare with N producers ($T = 20$). Left three panels: fraction of markets where weighted winrate gives strictly lower instantaneous consumer welfare than winrate, by value distribution and β. Right: gap to the welfare-optimal profile under winrate (x-axis) and weighted winrate (y-axis) for $N = 5$, $\beta = 10$, analogous to Figure 1d. Points above the diagonal are markets where weighted winrate does worse.

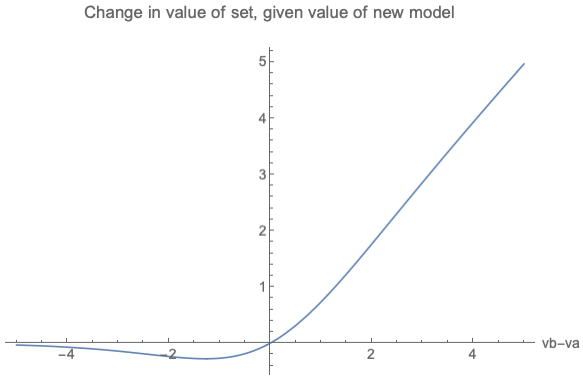


Figure 15: Plot of the function in Lemma 9 for a single task, with $\beta = 1$: equivalent to $\frac{\exp(\Delta/\beta)}{\exp(\Delta/\beta)+1} \cdot \Delta$, for $\Delta = v_b - v_a$. Note that in the example in Lemma 1, as $\Delta = v_b - v_a$ goes from -4 to -1, the curve becomes *more* negative, meaning that it reduces consumer welfare.

Proof. Fixing a single task t, consumer welfare with models $\{A, B\}$ is given by:

$$\begin{aligned} & v_{t,a} \cdot \frac{\exp(v_{t,a}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)} + v_{t,b} \cdot \frac{\exp(v_{t,b}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)} \\ = v_{t,a} \cdot & \left(1 - \frac{\exp(v_{t,b}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)}\right) + v_{t,b} \cdot \frac{\exp(v_{t,b}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)} \\ & = (v_{t,b} - v_{t,a}) \cdot \frac{\exp(v_{t,b}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)} + v_{t,a} \end{aligned}$$

Note that the value of agent A alone is exactly $v_{t,a}$. Thus, $\{A, B\}$ together has greater value exactly when

$$\sum_{t \in [T]} (v_{t,b} - v_{t,a}) \cdot \frac{\exp(v_{t,b}/\beta)}{\exp(v_{t,a}/\beta) + \exp(v_{t,b}/\beta)} \geq 0$$

□

Note that this function is commonly known as the swish function when used for non-linearity in modern neural networks [37] in Lemma 9 satisfies a few properties for consumer welfare, proven in Theorem 1.

Theorem 1. *Given BTL aggregation, it is possible to have a set of existing models $\mathcal{M}$ and a new model B such that either 1) B has higher average value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$ decreases consumer welfare, or 2) B has* lower *average value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$* increases *consumer welfare. Moreover, this effect is dependent on temperature (β).*

Proof. 1. First, we will show an example where B has *higher* value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$ *reduces* consumer welfare. This proof is by example: consider a setting with exactly two types of tasks $T = 2$. Suppose that initially there are two different models: A and B, with values across both tasks of $[8, 5]$, and $[5, 8]$, and the incoming model C has values $[6, 8.1]$. Suppose BTL aggregation is done with accuracy value $\beta = 2$. Then, the consumer welfare of different sets are:

- Each model in isolation: $\{13, 13, 14.1\}$.
- $\{A, B\}$: 14.91.
- The new set $A, B, C = 14.85$

Note that $u(A, B, C) < u(A, B)$, even though C has higher value in isolation than either A or B.

2. Second, we will show an example where B has *lower* value than every model in $\mathcal{M}$, and yet adding B to $\mathcal{M}$ *increases* consumer welfare.

Consider a setting with exactly two tasks $T = 2$, and $\mathcal{M}$ is made up of a single model A, which has values $[6, 8]$, while model B has values $[7, 5]$. Note that while model B has lower total value, its relative strengths are anti-correlated with model A. If choice-based aggregation is "strong" enough, adding model B could help. Specifically:

- BTL aggregation with noise level $\beta = 1$ leads to higher value: $u(\{m_a, m_b\}) = 14.6 > u(m_a) = 14$.
- BTL aggregation with noise level $\beta = 5$ leads to *lower* value: $u(\{m_a, m_b\}) = 13.5 < u(m_a) = 14$.

□

Lemma 1. *BTL aggregation is non-monotone: e.g. a model could weakly* increase *the value of its responses on every task, and consumer welfare might* decrease*.*

Proof. Proof by example: assume there is a single task $T = 1$ and two agents, A and B, with aggregation by BTL with $\beta = 1$. Suppose that A has value 5. If B has value 1, value of $u([1, 5]) = 4.92$, and B is picked with probability $p_b([1, 5]) = 0.017$. If B has value 4, value of $u([1, 5]) = 4.73$, and B is picked with probability $p_b([1, 5]) = 0.27$. □

Definition 5 (Monotone region)**.** *A choice function $p(\cdot)$ is in the* monotone region *with respect to agent i, for a set of values $\vec{v}$ and parameters of choice function ϕ, if the derivative of consumer utility with respect to $\vec{v}_i$ is non-negative, and is in the non-monotone region if the derivative is negative.*

Theorem 2. *A separable choice function cannot be monotone.*

Proof. For a separable function to be monotone, we require that for all $\vec{v}$:

$$0 \leq \frac{\partial}{\partial \vec{v}_i} u(\vec{v}) = \frac{\partial}{\partial \vec{v}_i} \sum_{k \in [N]} \vec{v}_k \cdot \frac{f(\vec{v}_k)}{\sum_{j \in [N]} f(\vec{v}_j)} = \frac{\partial}{\partial \vec{v}_i} \frac{\sum_{k \in [N]} \vec{v}_k \cdot f(\vec{v}_k)}{\sum_{j \in [N]} f(\vec{v}_j)}$$

In this proof, we will show that for any separable function $f(\cdot)$, we can construct a $\vec{v}$ such that it is not monotone. We will begin by assuming that for at least one element k, $\vec{v}_k > 0$, which means that $\sum_{k \in [N]} f(\vec{v}_k) > 0$. For the fraction above, the derivative of the numerator is given by $\vec{v}_i \cdot \frac{\partial}{\vec{v}_i} f(\vec{v}_i) + f(\vec{v}_i)$ and the derivative of the denominator is given by $\frac{\partial}{\vec{v}_i} f(\vec{v}_i)$. Thus, the sign of the

derivative of the overall fraction is given by:

$$\left(\vec{v}_i \cdot \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) + f(\vec{v}_i)\right) \cdot \left(\sum_{j \in [N]} f(\vec{v}_j)\right) - \left(\sum_{j \in [N]} \vec{v}_j \cdot f(\vec{v}_j)\right) \cdot \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \geq 0$$

$$\left(\vec{v}_i \cdot \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) + f(\vec{v}_i)\right) \cdot E(\vec{v}) - \left(\sum_{j \in [N]} \vec{v}_j \cdot f(\vec{v}_j)\right) \cdot \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \geq 0$$

(Defining $E(\vec{v}) = \sum_{j \in [N]} f(\vec{v}_j)$)

$$E(\vec{v}) \cdot f(\vec{v}_i) + \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot \left(\vec{v}_i \cdot E(\vec{v}) - \sum_{j \in [N]} \vec{v}_j \cdot f(\vec{v}_j)\right) \geq 0$$

$$E(\vec{v}) \cdot f(\vec{v}_i) + \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot (\vec{v}_i \cdot E(\vec{v}) - E(\vec{v}) \cdot u(\vec{v})) \geq 0$$

(Using $u(\vec{v}) = \sum_{j \in [N]} \frac{\vec{v}_j \cdot f(\vec{v}_j)}{E(\vec{v})}$)

$$E(\vec{v}) \cdot f(\vec{v}_i) + \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot E(\vec{v}) \cdot (\vec{v}_i - u(\vec{v})) \geq 0$$

$$f(\vec{v}_i) + \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot (\vec{v}_i - u(\vec{v})) \geq 0$$

(Assuming $E(\vec{v}) > 0$)

$$f(\vec{v}_i) + \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot \vec{v}_i \geq \frac{\partial}{\partial \vec{v}_i} f(\vec{v}_i) \cdot u(\vec{v}) \tag{2}$$

We're assuming that there as at least one $\vec{v}_i$ with $f(v_i) > 0$. We will now construct a vector $\vec{v}$ such that Equation 2 cannot always hold (and thus, $f(\cdot)$ cannot always be monotone. Specifically, consider a setting with $N = 2$. Then,

$$u(\vec{v}) = \frac{f(\vec{v}_1) \cdot \vec{v}_1 + f(\vec{v}_2) \cdot \vec{v}_2}{f(\vec{v}_1) + f(\vec{v}_2)}$$

Assume that $\vec{v}_2 > \vec{v}_1 > 0$, and $\frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) > 0$. Then, Equation 2 written with respect to $\vec{v}_1$ becomes:

$$f(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) \cdot \vec{v}_1 \geq \frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) \cdot u(\vec{v})$$

$$\frac{f(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) \cdot \vec{v}_1}{\frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1)} \geq u(\vec{v}) \qquad (\tfrac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) > 0)$$

$$\frac{f(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1) \cdot \vec{v}_1}{\frac{\partial}{\partial \vec{v}_1} f(\vec{v}_1)} \geq \frac{f(\vec{v}_1) \cdot \vec{v}_1 + f(\vec{v}_2) \cdot \vec{v}_2}{f(\vec{v}_1) + f(\vec{v}_2)} \tag{3}$$

Note that the lefthand side of Equation 3 depends solely on $\vec{v}_1$, while the righthand side additionally depends on $\vec{v}_2$. Because $f(\vec{v}_2)$ is weakly increasing in $\vec{v}_2$, as $\vec{v}_2 \to \infty$, the righthand side of Equation 3 also approaches ∞. However, because the lefthand side is a constant in $\vec{v}_2$, for sufficiently large $\vec{v}_2$ Equation 3 must not hold, and thus the function given by $f(\cdot)$ cannot always be monotonic. □

Definition 6. *[Monotone example] Consider an aggregation function given by* $p_i(\vec{v}) = \frac{1}{\binom{|\vec{v}|}{2}} \cdot w_i(\vec{v})$ *with:*

$$w_i(\vec{v}) = \sum_{j=1, i \neq j}^{|\vec{v}|} w_{ij}(\vec{v}) = \sum_{j=1, i \neq j}^{|\vec{v}|} \frac{\max(\vec{v}_i - \vec{v}_j, 0) + c}{|\vec{v}_i - \vec{v}_j| + 2c}$$

for any $c > 0$.

Lemma 2. *There exists an aggregation function that is monotone in value.*

Proof. We will use the function in Example 6, restated here for convenience, and show that it is always monotone:

Definition 6. *[Monotone example] Consider an aggregation function given by* $p_i(\vec{v}) = \frac{1}{\binom{|\vec{v}|}{2}} \cdot w_i(\vec{v})$ *with:*

$$w_i(\vec{v}) = \sum_{j=1, i\neq j}^{|\vec{v}|} w_{ij}(\vec{v}) = \sum_{j=1, i\neq j}^{|\vec{v}|} \frac{\max(\vec{v}_i - \vec{v}_j, 0) + c}{|\vec{v}_i - \vec{v}_j| + 2c}$$

for any $c > 0$.

As we must know from Theorem 2, the function in Example 6 is *not* separable: it relies on the gap in value between items.

First, we will prove that this function is a valid choice-based aggregation method: specifically that it always sums up to a probability of 1. First, note that $w_{ij} + w_{ji} = 1$ for all $i \neq j$. Summing over all $\binom{|\vec{v}|}{2}$ pairs gives $\sum_{i\in[|\vec{v}|]} \sum_{j\in[|\vec{v}|], j\neq i} w_{i,j}(\vec{v}) = \binom{|\vec{v}|}{2}$, so $\sum_{i\in[|\vec{v}|]} w_i(\vec{v}) = 1$, as desired. Because $w_{i,j} \geq 0$, we know that $w_i(\vec{v}) \geq 0$ and thus each probability is non-negative, as required.

Next, we will show that this function is monotone in value, or

$$\begin{aligned}
0 \leq & \frac{\partial}{\partial \vec{v}_i} u(\vec{v}) \quad \forall i \in [|\vec{v}|] \\
= & \frac{\partial}{\partial \vec{v}_i} \sum_{i\in[|\vec{v}|]} p_i(\vec{v}) \cdot \vec{v}_i \\
= & p_i(\vec{v}) + \vec{v}_i \cdot \frac{\partial}{\partial v_i} p_i(\vec{v}) + \sum_{j\neq i, j\in[|\vec{v}|]} \frac{\partial}{\partial \vec{v}_i} \vec{v}_j \cdot p_j(\vec{v}) \\
\propto & \sum_{k\in[|\vec{v}|], k\neq i} w_{ik}(\vec{v}) + \vec{v}_i \cdot \frac{\partial}{\partial v_i} \sum_{k\in[|\vec{v}|], k\neq i} w_{ik}(\vec{v}) + \sum_{j\neq i, j\in[|\vec{v}|]} \vec{v}_j \sum_{k\in[|\vec{v}|], k\neq i, j} \frac{\partial}{\partial \vec{v}_i} \cdot w_{jk}(\vec{v}) \\
& \qquad \text{(Definition 6)} \\
= & \sum_{k\in[|\vec{v}|], k\neq i} w_{ik}(\vec{v}) + \vec{v}_i \cdot \frac{\partial}{\partial v_i} \sum_{k\in[|\vec{v}|], k\neq i} w_{ik}(\vec{v}) + \sum_{j\neq i, j\in[|\vec{v}|]} \vec{v}_j \frac{\partial}{\partial \vec{v}_i} \cdot w_{ji}(\vec{v}) \\
& \qquad (\tfrac{\partial}{\partial \vec{v}_i} w_{jk} = 0 \text{ if } j \neq i \text{ and } k \neq i) \\
= & \sum_{j\in[|\vec{v}|], j\neq i} w_{ij}(\vec{v}) + \vec{v}_i \cdot \frac{\partial}{\partial v_i} \sum_{j\in[|\vec{v}|], j\neq i} w_{ij}(\vec{v}) - \sum_{j\neq i, j\in[|\vec{v}|]} \vec{v}_j \frac{\partial}{\partial \vec{v}_i} \cdot w_{ij}(\vec{v}) \quad (w_{ji} = 1 - w_{ij}) \\
= & \sum_{j\in[|\vec{v}|], j\neq i} \left(w_{ij}(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot \frac{\partial}{\partial \vec{v}_i} \cdot w_{ij}(\vec{v}) \right)
\end{aligned}$$

Next, we will show that every term within the sum above is positive: that is, that $w_{ij}(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot \frac{\partial}{\partial \vec{v}_i} \cdot w_{ij}(\vec{v}) \geq 0$ always.

Note that by the definition in Definition 6, we know that:

$$\begin{cases} \frac{\partial}{\partial \vec{v}_i} w_{ij} = \frac{\partial}{\partial \vec{v}_i} \frac{\vec{v}_i - \vec{v}_j + c}{\vec{v}_i - \vec{v}_j + 2c} = \frac{c}{(|(|v_i - v_j) + 2c)^2} & \vec{v}_i \geq \vec{v}_j \\ \frac{\partial}{\partial \vec{v}_i} w_{ij} = \frac{\partial}{\partial \vec{v}_i} \frac{c}{\vec{v}_j - \vec{v}_i + 2c} = \frac{c}{(|(\vec{v}_i - \vec{v}_j) + 2c)^2} & \vec{v}_i < \vec{v}_j \end{cases}$$

Thus, the desired condition holds for every pair with $\vec{v}_i > \vec{v}_j$ if:

$$\begin{aligned}
0 < & w_{ij}(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot \frac{\partial}{\partial \vec{v}_i} \cdot w_{ij}(\vec{v}) \\
= & \frac{\vec{v}_i - \vec{v}_j + c}{\vec{v}_i - \vec{v}_j + 2c} + (\vec{v}_i - \vec{v}_j) \cdot \frac{c}{(\vec{v}_i - \vec{v}_j + 2c)^2}
\end{aligned}$$

which is automatically satisfied because both terms are positive. The desired condition holds for every j pair with $\vec{v}_i < \vec{v}_j$ if:

$$\begin{aligned}
0 <& w_{ij}(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot \frac{\partial}{\partial \vec{v}_i} \cdot w_{ij}(\vec{v}) \\
0 <& \frac{c}{\vec{v}_j - \vec{v}_i + 2c} + (\vec{v}_i - \vec{v}_j) \cdot \frac{c}{(\vec{v}_j - \vec{v}_i + 2c)^2} \\
(\vec{v}_j - \vec{v}_i) \cdot \frac{c}{(\vec{v}_j - \vec{v}_i + 2c)^2} <& \frac{c}{\vec{v}_j - \vec{v}_i + 2c} \\
(\vec{v}_j - \vec{v}_i) \cdot \frac{1}{\vec{v}_j - \vec{v}_i + 2c} <& 1 \qquad (\vec{v}_j - \vec{v}_i + c > 0) \\
\vec{v}_j - \vec{v}_i <& \vec{v}_j - \vec{v}_i + 2c
\end{aligned}$$

which is satisfied because $c > 0$, by assumption. □

Lemma 10. *Consider a vector of values $\vec{v}$ and a new vector $\vec{v}'$ where with index j changed and all else constant. Then, $u(\vec{v}') \geq u(\vec{v})$ if and only if the following condition holds:*

$$\sum_{i \in |\vec{v}|, i \neq j} (p_i(\vec{v}) - p_i(\vec{v}')) \cdot (\vec{v}'_j - \vec{v}_i) > (\vec{v}_j - \vec{v}'_j) \cdot p_j(\vec{v})$$

Proof. This can be seen simply by rewriting the value function:

$$u(\vec{v}) = \vec{v}_j \cdot p_j(\vec{v}) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) \cdot \vec{v}_i$$

and

$$u(\vec{v}') = \vec{v}'_j \cdot p_j(\vec{v}') + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}') \cdot \vec{v}_i$$

Note that we can write:

$$\begin{aligned}
p_j(\vec{v}') =& p_j(\vec{v}) + p_j(\vec{v}') - p_j(\vec{v}) \\
=& p_j(\vec{v}) + 1 - \sum_{i \neq j} p_i(\vec{v}') - 1 + \sum_{i \neq j} p_i(\vec{v}) \\
=& p_j(\vec{v}) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) - p_i(\vec{v}')
\end{aligned}$$

The condition where the value of $\vec{v}'$ is larger is given by:

$$\begin{aligned}
u(\vec{v}') \geq& u(\vec{v}) \\
\vec{v}'_j \cdot p_j(\vec{v}') + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}') \cdot \vec{v}_i \geq& \vec{v}_j \cdot p_j(\vec{v}) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) \cdot \vec{v}_i \\
\vec{v}'_j \cdot \left(p_j(\vec{v}) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) - p_i(\vec{v}') \right) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}') \cdot \vec{v}_i \geq& \vec{v}_j \cdot p_j(\vec{v}) + \sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) \cdot \vec{v}_i \\
\vec{v}'_j \cdot \left(\sum_{i \in \vec{v}, i \neq j} p_i(\vec{v}) - p_i(\vec{v}') \right) + \sum_{i \in \vec{v}, i \neq j} (p_i(\vec{v}') - p_i(\vec{v})) \cdot \vec{v}_i \geq& \vec{v}_j \cdot p_j(\vec{v}) - \vec{v}'_j \cdot p_j(\vec{v}) \\
\sum_{i \in \vec{v}, i \neq j} (p_i(\vec{v}) - p_i(\vec{v}')) \cdot (\vec{v}'_j - \vec{v}_i) >& (\vec{v}_j - \vec{v}'_j) \cdot p_j(\vec{v})
\end{aligned}$$

giving the desired condition. □

Lemma 11. *Consider any continuous increasing function $f(\vec{v})$. Then, if f is monotone it must have $p_j(\vec{v}) > 0$ always.*

Proof. This proof will work by considering a hypothetical vector of values $\vec{v}$ and showing that monotonicity is impossible to guarantee if we allow $p_j(\vec{v}) = 0$. Specifically, we will construct a separate vector $\vec{v}'$ that is identical to $\vec{v}$ except in index j. We will rely on the result from Lemma 10, which says that such a vector has $u(\vec{v}') \geq u(\vec{v})$ if and only if:

$$\sum_{i \in |\vec{v}|, i \neq j} (p_i(\vec{v}) - p_i(\vec{v}')) \cdot (\vec{v}'_j - \vec{v}_i) > (\vec{v}_j - \vec{v}'_j) \cdot p_j(\vec{v})$$

Consider a hypothetical $\vec{v}$ with a minimal element $\vec{v}_1$ and $p_1(\vec{v}) = 0$. Assume that 1 is the only element with $p_1(\vec{v}) = 0$. In particular, consider the second smallest element 2, and assume that $p_2(\vec{v}) > 0$.

Consider the hypothetical vector $\vec{v}'$ with $\vec{v}'_1 = \vec{v}'_2 = \vec{v}_2$. Because we assumed anonymity, we require $p_1(\vec{v}') = p_2(\vec{v}')$. Note by substitutability, we must have that increasing $\vec{v}_1$ can only keep constant or decrease the probability of other elements $\vec{v}_i$ being picked. This means that we must have $p_1(\vec{v}') = p_2(\vec{v}') > 0$: if this is not the case, then by conservation of total probability, we must have $p_j(\vec{v}') > p_j(\vec{v})$, which violates substitutability.

Given, $p_1(\vec{v}') = p_2(\vec{v}') > 0$, by continuity, there must exist a point $\vec{v}_1 < \vec{v}''_1 < \vec{v}_2$ such that $p_1(\vec{v}'') > 0$. Note that this means that $p_1(\vec{v}'') > p_1(\vec{v}) = 0$. We will again show that monotonicity is violated here.

We note that:

1. By assumption, $\vec{v}_j < \vec{v}'_j$ and so $\vec{v}_j - \vec{v}'_j < 0$.
2. By assumption, for all other $i \neq j$, we have $\vec{v}'_j < \vec{v}_i$, and so $\vec{v}'_j - \vec{v}_i < 0$.
3. By substitutability (Def. 9), $p_i(\vec{v}) - p_i(\vec{v}') \geq 0$ for all $i \neq j$.

Considering the inequality from Lemma 10, we know that $u(\vec{v}') \geq u(\vec{v})$ exactly whenever:

$$\sum_{i \in \vec{v}, i \neq j} (p_i(\vec{v}) - p_i(\vec{v}')) \cdot (v'_j - v_i) > (v_j - v'_j) \cdot p_j(\vec{v}) \tag{4}$$

For the lefthand side, property 2 tells us that $v'_j - v_i$ is negative always and property 3 tells us that $p_i(\vec{v}) - p_i(\vec{v}') \geq 0$. Note that the LHS can only be 0 if $p_i(\vec{v}) = p_i(\vec{v}')\ \forall i \neq j$: however, this is impossible by the requirement that $p_j(\vec{v}') > p_j(\vec{v})$ (given by the continuity of the selection function). Thus, we know that the lefthand side of Equation 4 is always strictly negative. Thus, in order for the function to be monotone, we must have that the RHS of Equation 4 is always strictly negative. By Property 1, we know that $v_j - v'_j < 0$, and so we must have $p_j(\vec{v}) > 0$ strictly. □

C.1 Tension between monotonicity and model creation

Definition 7 (Substitutability)**.** *A selection function $p(\cdot)$ satisfies* substitutability *if increasing the value of an item $\vec{v}_{m,t}$ either holds constant or decreases the probability of every other item being picked. That is, for $\vec{v}_t, \vec{v}_{t'}$ identical except with $\vec{v}_{m,t} < \vec{v}'_{m,t}$, we must have $p_m(\vec{v}_t) \leq p_m(\vec{v}'_t)$.*

Definition 8 (Anonymity)**.** *A selection function $p(\cdot)$ satisfies* anonymity *if the probability of picking an item depends only on its value.*

Here, we want to highlight a tension between consumer welfare increase from model *replacement* (monotonicity) and consumer welfare increase from model *creation*. That is, for any choice functions satisfying natural properties (substitutability and anonymity, Definitions 9,10), these two properties cannot be simultaneously satisfied.

Definition 9 (Substitutability)**.** *A selection function $p(\cdot)$ satisfies* substitutability *if increasing the value of an item $\vec{v}_{m,t}$ either holds constant or decreases the probability of every other item being picked. That is, for $\vec{v}_t, \vec{v}_{t'}$ identical except with $\vec{v}_{m,t} < \vec{v}'_{m,t}$, we must have $p_m(\vec{v}_t) \leq p_m(\vec{v}'_t)$.*

Definition 10 (Anonymity)**.** *A selection function $p(\cdot)$ satisfies* anonymity *if the probability of picking an item depends only on its value.*

Theorem 7. *These two properties are mutually exclusive for choice-based, continuous aggregation functions satisfying substitutability and anonymity*[10]*: 1) the function is monotone, and 2) there are always weak benefits to adding in another agent.*

Proof. This comes almost directly from Lemma 11. First, suppose that the function is always monotone in value. Then, by Lemma 11, we know that there must be a positive probability of picking any item within $\vec{v}$. Then, if we create a $\vec{v}'$ with a new element such that the vector becomes $\vec{v} \oplus v$, where it is smaller than the current minimal element $v < \min_{i \in |\vec{v}|} \vec{v}_i$, then we must have $p_{N+1}(\vec{v}') > 0$. Because total probability is conserved, we must see that the probability of picking every other item $i \neq j$ weakly decreases. Because every other item in $\vec{v}'$ has strictly higher value than v, this means that we must have $u(\vec{v}') < u(\vec{v})$, and so there are strict harms from adding in another agent.
Next, we assume that there are always weak benefits to adding in another agent: that is, $u(\vec{v} \oplus v) > u(\vec{v})$ always. In this case, we must have that for an element $v < \min_{i \in |\vec{v}|} \vec{v}_i$, if $u(\vec{v} \oplus v) > u(\vec{v})$, then we must have $p_{N+1}(\vec{v}) = 0$. By the conditions of Lemma 11, this must mean that the associated selection function cannot be monotone. □

Lemma 3. *BTL aggregation function is in the monotone region when 1) $\beta = 0$, corresponding to perfect aggregation, or 2) when β is sufficiently high: that is, for a given set of values $\vec{v}$ and a model i, BTL aggregation is guaranteed to be in the monotone region whenever $\beta > \max_{i,j} \vec{v}_j - \vec{v}_i$.*

Proof. First, we note that perfect aggregation ($\beta = 0$) is always monotone because increasing elements of a list $\vec{v}$ only increases the maximum element in that list. Similarly, random aggregation ($\beta = \infty$) is always monotone because increasing elements of a list $\vec{v}$ only increases the average of that list.

Next, we turn to analyzing the conditions for monotonicity with $\beta \in (0, \infty)$. A function is monotone whenever the below quantity is positive for all $i \in |\vec{v}|$:

$$\begin{aligned}
\frac{\partial}{\partial \vec{v}_i} u(\vec{v}) &= \sum_{j \in |\vec{v}|} \frac{\partial}{\partial \vec{v}_i} \vec{v}_j \cdot p_j(\vec{v}) \\
&= p_i(\vec{v}) + \vec{v}_i \cdot \frac{\partial}{\partial \vec{v}_i} p_i(\vec{v}) + \sum_{j \neq i} \vec{v}_j \cdot \frac{\partial}{\partial \vec{v}_i} p_j(\vec{v}) \\
&= p_i(\vec{v}) + \vec{v}_i \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta - \sum_{j \neq i} \vec{v}_j \cdot p_i(\vec{v}) \cdot p_j(\vec{v})/\beta \qquad \text{(Properties of BTL)} \\
&= p_i(\vec{v}) \cdot \left(1 + \sum_{j \neq i} (\vec{v}_i - \vec{v}_j) p_j(\vec{v})/\beta \right)
\end{aligned}$$

The above quantity is positive so long as

$$\begin{aligned}
1 + \sum_{j \neq i} (v_i - v_j) p_j(\vec{v})/\beta > 0 \\
\beta + \sum_{j \neq i} (v_i - v_j) p_j(\vec{v}) > 0 \\
\beta > \sum_{j \neq i} (v_j - v_i) \cdot p_j(\vec{v}) \qquad (5)
\end{aligned}$$

If $\vec{v}_i \geq \vec{v}_j$ for all $i \neq j$ (that is, if element i is the largest) then Equation 5 is automatically satisfied for player i because the righthand side is negative. Next, we will consider cases where there are some agents such that $\vec{v}_i < \vec{v}_j$. Note that as β increases, $p_j(\vec{v})$ may increase or decrease, but is upper bounded by 1, and we have $\sum_{j \neq i} p_j(\vec{v}) \leq 1$. Thus, we have:

$$\sum_{j \neq i} (v_j - v_i) \cdot p_j(\vec{v}) \leq \max_{j \neq i} \vec{v}_j - \vec{v}_i$$

[10] Informally, a function satisfies anonymity if the probability of picking an item depends only on its value, and substitutability if the probability of picking an item only stays constant or increases when its value increases.

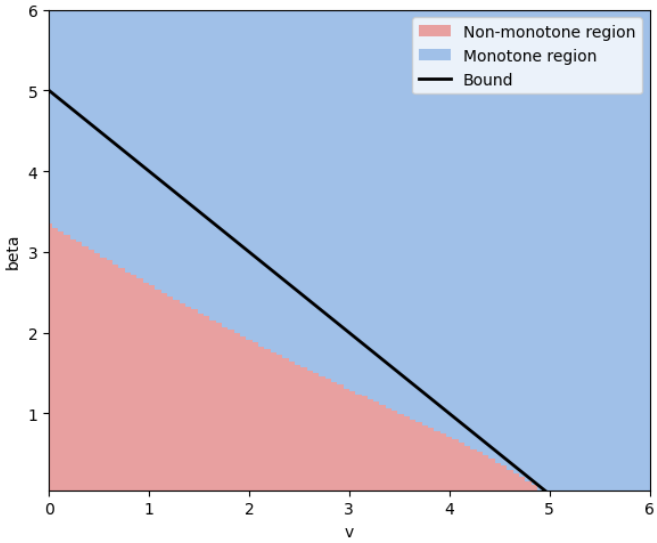


Figure 16: Figure demonstrating the bound in Lemma 3, where there is a set of values $[1, 3, 5, v]$, with v varying along the x axis and β varying along the y axis. Blue regions denote settings where BTL aggregation is in the monotone region with respect to v, and red regions denote settings where it is not. The black line gives the bound in Lemma 3: above this line BTL aggregation is guaranteed to be in the monotone region with respect to v.

For any agent i, for any $\beta > \beta_i^* = \max_{j \neq i} \vec{v}_j - \vec{v}_i$, Equation 5 is guaranteed to be positive, and the condition is monotone. □

D Proofs for Section 5

Lemma 12. *The change in consumer welfare when a new agent with value v' joins a task with existing list of values $\vec{v}$ is given by:*

$$u(\vec{v} \oplus v') - u(\vec{v}) = p_{N+1}(\vec{v} \oplus v') \cdot (\vec{v}' - u(\vec{v}))$$

where $p_{N+1}(\cdot)$ gives the probability that the new item is picked.

Proof. Suppose that we have $N = |\vec{v}| - 1$ agents initially. Then, we can calculate the change in value as:

$$\begin{aligned}
u(\vec{v} \oplus v') - u(\vec{v}) =& \sum_{i \in [N]} \vec{v}_i \cdot p_i(\vec{v} \oplus v') + v' \cdot p_N(\vec{v} \oplus v') - \sum_{i \in [N]} \vec{v}_i \cdot p_i(\vec{v}) \\
=& v' \cdot p_N(\vec{v} \oplus v') - \left(\sum_{i \in [N]} \vec{v}_i \cdot (p_i(\vec{v}) - p_i(\vec{v} \oplus v')) \right) \\
=& v' \cdot \frac{\exp(v'/\beta)}{E + \exp(v'/\beta)} - \sum_{i \in [N]} \vec{v}_i \cdot \exp(\vec{v}_i/\beta) \left(\frac{1}{E} - \frac{1}{E + \exp(v'/\beta)} \right) \\
=& v' \cdot \frac{\exp(v'/\beta)}{E + \exp(v'/\beta)} - \sum_{i \in [N]} \vec{v}_i \cdot \exp(\vec{v}_i/\beta) \frac{\exp(v'/\beta)}{E \cdot (E + \exp(v'/\beta)}) \\
=& \frac{\exp(v'/\beta)}{E + \exp(v'/\beta)} \left(v' - \sum_{i \in [N]} \vec{v}_i \cdot \frac{\exp(\vec{v}_i/\beta)}{E} \right) \\
=& p_N(\vec{v} \oplus v') \cdot (\vec{v}' - u(\vec{v})) \qquad (6)
\end{aligned}$$

as desired. □

Lemma 13. *Consider a pair of tasks where the existing allocation is $\vec{v}_1, \vec{v}_2$. A new agent has total value V to allocate across both tasks, and is optimizing for consumer welfare. Then, the optimal arrangement is always to put all value on a single task and abstain on the other.*

Proof. Recall from Lemma 12 that when new value v' is added to a task with existing values $\vec{v}$, the change in consumer welfare is given by:

$$u(\vec{v} \oplus v') - u(\vec{v}) = p_N(\vec{v} \oplus v') \cdot (\vec{v}' - u(\vec{v}))$$

where $p_N(\cdot)$ gives the probability that the new item is picked. Thus, given an allocation $v, V - v$ across both tasks, the *change* in consumer welfare across both tasks is given by:

$$p_{N+1}(\vec{v}_1 \oplus v) \cdot (v - u(\vec{v}_1)) + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) \tag{7}$$

Our goal is to show that the objective in Equation 7 is maximized at one of the endpoints. Note that this objective is *not* convex, and in particular can have multiple local maxima and minima. We will prove that Equation 7 is maximized at its endpoints by considering different values of $v \in (0, V)$ and showing that the value obtained there must be lower.

First, consider the setting where we have v such that $v < u(v_1)$ or $V - v < u(v_2)$: that is, at least one of the allocation is *decreasing* the average value on the task. First, note that if the allocation is decreasing on both tasks, then the total change is negative:

$$\begin{aligned} p_{N+1}(\vec{v}_1 \oplus v) \cdot (v - u(\vec{v}_1)) + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) &\leq 0 \\ &\leq \max_{i \in [1,2]} p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i)) \end{aligned}$$

And so the condition is satisfied. Next, suppose that the allocation leads to negative change on exactly one task: WLOG, assume it occurs on task 1. Then, we know that:

$$\begin{aligned} p_{N+1}(\vec{v}_1 \oplus v) \cdot (v - u(\vec{v}_1)) + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) &< p_{N+1}(\vec{v}_1 \oplus v) \cdot 0 \\ &\quad + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) \\ &= p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) \\ &< p_{N+1}(\vec{v}_2 \oplus V) \cdot (V - u(\vec{v}_2)) \\ &\leq \max_{i \in [1,2]} p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i)) \end{aligned}$$

again, as desired. Finally, we consider the case where $v \geq u(v_1)$ and $V - v \geq u(v_2)$: the allocation leads to positive benefits on both tasks. WLOG, assume that the task has a higher chance of being picked on task 1 than 2: that is:

$$p_{N+1}(\vec{v}_1 \oplus v) \geq p_{N+1}(\vec{v}_2 \oplus V - v) \tag{8}$$

Then, we know that:

$$\begin{aligned} p_{N+1}(\vec{v}_1 \oplus v) \cdot (v - u(\vec{v}_1)) + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (V - v - u(\vec{v}_2)) &< p_{N+1}(\vec{v}_1 \oplus v) \cdot (v - u(\vec{v}_1)) \\ &\quad + p_{N+1}(\vec{v}_1 \oplus V - v) \cdot (V - v) \\ &\quad + p_{N+1}(\vec{v}_2 \oplus V - v) \cdot (-u(\vec{v}_2)) \\ &\qquad \text{(Eq. 8)} \\ &= p_{N+1}(\vec{v}_1 \oplus v) \cdot (V - u(\vec{v}_1)) \\ &\quad - p_{N+1}(\vec{v}_2 \oplus V - v) \cdot u(\vec{v}_2) \\ &< p_{N+1}(\vec{v}_1 \oplus V) \cdot (V - u(\vec{v}_1)) \\ &\leq \max_{i \in [1,2]} p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i)) \end{aligned}$$

again, as desired.

□

Lemma 4. *If a new model with total value V aims to maximize consumer welfare, its best action is: if V has higher value than the minimum existing values on tasks ($V \geq \min_{i \in [T]} u(\vec{v}_i)$), specialize by allocating all value on a single task i which maximizes $p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i))$. If V is not sufficiently high, then its best action is to abstain (fail to create a model).*

Proof. First, Lemma 12 tells us that when a new value v' is added to a task with existing list of values $\vec{v}$, the change in consumer welfare is given by:

$$u(\vec{v} \oplus v') - u(\vec{v}) = p_N(\vec{v} \oplus v') \cdot (\vec{v}' - u(\vec{v}))$$

where $p_N(\cdot)$ gives the probability that the new item is picked.

This immediately handles the setting where $V < u(\vec{v})$: if total value is less than current value, any possible allocation would harm consumer welfare, and the best action is to entirely abstain from entering the market.

Next, Lemma 13 tells us that the optimal allocation for maximizing value must place all value on a single task.

Finally, we can directly see which allocation maximizes consumer welfare: it is whichever task i maximizes:

$$p_{N+1}(\vec{v}_i \oplus V) \cdot (V - u(\vec{v}_i))$$

□

Lemma 14. *Consider a pair of tasks where the existing allocation is $\vec{v}_1, \vec{v}_2$. A new agent has total value V to allocate across both tasks, and is optimizing for total winrate. Then, the optimal arrangement for total winrate with BTL by the new agent is given by:*

- *If $V < \beta \cdot \log(E_1 \cdot E_2)$, where $E_i = \sum_{j \in |\vec{v}|} \exp(\vec{v}_{j,i}/\beta))$, have the allocation maximally anti-correlated, with all value on the task with smaller E_i value: $v_1 = V, v_2 = 0$ (or vice-versa).*
- *If $V \geq \beta \cdot \log(E_1 \cdot E_2)$, where $E_i = \sum_{j \in |\vec{v}|} \exp(\vec{v}_{j,i}/\beta))$, the optimal arrangement is to have the allocation such that winrate is equal across both tasks.*

Proof. Denote the new agent's allocation on tasks by $v_1, V - v_1$. Denote the new agent by index c and denote $E_i = \sum_{j \in |\vec{v}_i|} \exp(v_j/\beta)$. Then, new agent c winning on a given task with its allocated value v_1 is given by the BTL function:

$$p_c(\vec{v}_i \oplus v_i) = \frac{\exp(v_i/\beta)}{E_i + \exp(v_i/\beta)}$$

The total winrate for agent c across the two tasks is:

$$W(v_1) = p_c(\vec{v}_1 \oplus v_1) + p_c(\vec{v}_2 \oplus V - v_1) = \frac{\exp(v_1/\beta)}{E_1 + \exp(v_1/\beta)} + \frac{\exp((V - v_1)/\beta)}{E_2 + \exp((V - v_1)/\beta)}$$

Next, we will analyze this function and show when it is maximized at the endpoints, rather than towards the center. First, note that the derivative of W with respect to v_1 is:

$$\frac{\partial W}{\partial v_1} = p_c'(\vec{v}_1 \oplus v_1) - p_c'(\vec{v}_2 \oplus V - v_1) = (p_c(\vec{v}_1 \oplus v_1) \cdot (1 - p_c(\vec{v}_1 \oplus v_1)) - p_c(\vec{v}_2 \oplus V - v_1) \cdot (1 - p_c(\vec{v}_2 \oplus V - v_1))) \cdot 1/\beta$$

And thus $\frac{\partial}{\partial v_1} W(v_1)$ is 0 either a) when $p_c(v_1) = p_c(V - v_1)$ or b) when $p_c(v_1) = 1 - p_c(V - v_1)$. We will first show that $b)$ is only possible if $V = \log(E_1 \cdot E_2)$. This occurs when:

$$\begin{aligned} p_c(\vec{v}_1 \oplus v_1) &= 1 - p_c(\vec{v}_2 \oplus V - v_1) \\ \frac{\exp(v_1/\beta)}{E_1 + \exp(v_1/\beta)} &= \frac{E_2}{E_2 + \exp((V - v_1)/\beta)} \\ E_2 \cdot \exp(v_1/\beta) + \exp(V/\beta) &= E_1 \cdot E_2 + E_2 \cdot \exp(v_1/\beta) \\ \exp(V/\beta) &= E_1 \cdot E_2 \\ V/\beta &= \log(E_1 \cdot E_2) \end{aligned}$$

Note that this condition holds irrespective of v_1: that is, if we have $V = \beta \cdot \log(E_1 \cdot E_2)$, then we must have $\frac{\partial}{\partial v_1} W = 0$ always, and so any allocation will give equal total winrate across both tasks to player c.

We next turn to the setting where $V \neq \beta \cdot \log(E_1 \cdot E_2)$: thus, this means that the only point of zero derivative $\frac{\partial}{\partial v_1} W$ occurs exactly where the probabilities of winning are identical: $p_c(v_1) = p_c(V - v_1)$. We will show whether this is a maximum or minimum by examining the second derivative of the function W at the this point. First, we note that by the properties of the BTL function:

$$\frac{\partial W}{\partial v_1} = (p_c(\vec{v}_1 \oplus v_1) \cdot (1 - p_c(\vec{v}_1 \oplus v_1)) - p_c(\vec{v}_2 \oplus V - v_1) \cdot (1 - p_c(\vec{v}_2 \oplus V - v_1))) \cdot 1/\beta$$

and thus the second derivative is given by:

$$\begin{aligned}
\frac{\partial^2}{\partial v_1^2} W =& p_c(\vec{v}_1 \oplus v_1) \cdot (1 - p_c(\vec{v}_1 \oplus v_1)/\beta^2 - 2 \cdot p_c(\vec{v}_1 \oplus v_1) \cdot (1 - p_c(\vec{v}_1 \oplus v_1))/\beta^2 \\
& - (-p_c(\vec{v}_2 \oplus V - v_1) \cdot (1 - p_c(\vec{v}_2 \oplus V - v_1)) + 2 p_c(\vec{v}_2 \oplus V - v_1)^2 \cdot (1 - p_c(\vec{v}_2 \oplus V - v_1)))/\beta^2 \\
=& 1/\beta^2 \cdot (p_c(\vec{v}_1 \oplus v_1) \cdot (1 - p_c(\vec{v}_1 \oplus v_1)) \cdot (1 - 2 \cdot p_c(\vec{v}_1 \oplus v_1))) \\
& + 1/\beta^2 \cdot (p_c(\vec{v}_2 \oplus V - v_1) \cdot (1 - p_c(\vec{v}_2 \oplus V - v_1)) \cdot (1 - 2 \cdot p_c(\vec{v}_2 \oplus V - v_1)))
\end{aligned}$$

Next, we can evaluate this second derivative at the point where $p_c(\vec{v}_1 \oplus v_1) = p_c(\vec{v}_2 \oplus V - v_1) = p^*$, and thus we can simplify the above term to:

$$2 \cdot p^* \cdot (1 - p^*) \cdot (1 - 2 \cdot p^*) \cdot 1/\beta^2$$

Note that $p^* \cdot (1 - p^*) > 0$ always, and so the sign of the second derivative depends entirely on whether the term $1 - 2 \cdot p^*$ is positive or negative. This occurs exactly when equalizing the winrates allows the new agent to have winrate > 0.5 on both tasks. When does this occur? The value giving equal winrate is:

$$\begin{aligned}
p_c(\vec{v}_1 \oplus v_1) =& p_c(\vec{v}_2 \oplus V - v_1) \\
\frac{\exp(v_1/\beta)}{E_1 + \exp(v_1/\beta)} =& \frac{\exp((V - v_1)/\beta)}{E_2 + \exp((V - v_1)/\beta)} \\
E_2 \cdot \exp(v_1/\beta) + \exp(V/\beta) =& E_1 \cdot \exp((V - v_1)/\beta) + \exp(V/\beta) \\
E_2 \cdot \exp(v_1/\beta) =& E_1 \cdot \exp((V - v_1)/\beta) \\
\exp(2v_1/\beta) =& \frac{E_1}{E_2} \cdot \exp(V/\beta) \\
v_1 =& 0.5 \cdot \beta \cdot \log\left(\frac{E_1}{E_2}\right) + 0.5 \cdot V
\end{aligned}$$

This gives winrate > 0.5 whenever:

$$\begin{aligned}
\exp(v_1/\beta) >& E_1 \\
\sqrt{\frac{E_1}{E_2}} \cdot \exp(V/(2\beta)) >& E_1 \\
\frac{E_1}{E_2} \cdot \exp(V/\beta) >& E_1^2 \\
\exp(V/\beta) >& E_1 \cdot E_2 \\
V >& \beta \cdot \log(E_1 \cdot E_2)
\end{aligned}$$

Thus, if V is below this bound, then the midpoint has a positive second derivative, and thus is a minimum: in this case, the function W is maximized at the endpoints, with $v_1 = V, v_2 = 0$. Note that the allocation will always have nonzero allocation on the task with smaller E_i, because if $E_1 < E_2$, then:

$$\begin{aligned}
p_{N+1}(\vec{v}_1 \oplus V) + p_{N+1}(\vec{v}_2 \oplus 0) >& p_{N+1}(\vec{v}_1 \oplus 0) + p_{N+1}(\vec{v}_2 \oplus V) \\
\frac{\exp(V/\beta)}{\exp(V/\beta) + E_1} + \frac{1}{1 + E_2} >& \frac{\exp(V/\beta)}{\exp(V/\beta) + E_2} + \frac{1}{1 + E_1} \\
E_2 >& E_1
\end{aligned}$$

On the other hand, if $V > \beta \cdot \log(E_1 \cdot E_2)$, then the internal zero has a *negative* second derivative, and thus is a *maximum*: in this case, the function W is maximized at this internal point. □

Theorem 3. *Consider a set of tasks with existing values $\{\vec{v}_i\}$ across tasks. Then, an incoming model with total value V optimizing for* total winrate *must equalize winrate across tasks whenever:* $V > 2 \cdot T \cdot \beta \cdot \max_{j\in[T]} \log\left(\sum_{i\in[N]} \exp(\vec{v}_{ji}/\beta)\right)$

Proof. This proof comes almost directly from the proof of Lemma 14. Note that Lemma 14 tells us that if, for any *pair* of tasks $1, 2$, if the total value the new agent c has between those two tasks is at least

$$V \geq \beta \cdot \left(\log(\sum_{i\in|\vec{v}_1|} \exp(\vec{v}_{i1}/\beta)) + \log(\sum_{i\in|\vec{v}_2|} \exp(\vec{v}_{i2}/\beta))\right) = \beta \cdot (\log(E_1) + \log(E_2))$$

where we have used $E_i = \sum_{j\in|\vec{v}|} \exp(\vec{v}_{j,i}/\beta))$ for conciseness. If this is satisfied, then the optimal allocation *across those two tasks* is for agent C to equalize winrate (with winrate > 0.5) across both of them.

Next, we will show that if agent C's total value across all tasks is at least $V \geq 2 \cdot \max_{i\in[T]} \beta \cdot T \log(E_i)$, any allocation besides one with equal winrate across tasks must have *lower* total winrate.

First, we note that if for all tasks i, we have $v_i > \beta \cdot \log(E_i) = \beta \cdot \log(\sum_{j\in[N]} \exp(\vec{v}_{j,i}/\beta))$, then the precondition of Lemma 14 is satisfied, and the optimal allocation is always to equalize winrate.

Next, we will consider the case where $v_k < \beta \cdot \log(E_k) = \beta \cdot \log(\sum_{i\in[N]} \exp(\vec{v}_{k,i}/\beta)$ for some task k, and show by the pigeonhole principle that there must exist some other task j such that $v_k + v_j > \beta \cdot (\log(E_k) + \log(E_j))$, and again by Lemma 14 it would be optimal to equalize winrate across tasks. Because we know that the total value across all tasks is $V \geq 2 \cdot \max_{i\in[T]} \beta \cdot T \log(E_i)$, if on a particular task k we have $v_k < \beta \cdot \log(E_k)$, we must have that the *average* value across all other tasks is lower bounded by:

$$\begin{aligned}
\frac{V - v_k}{T-1} \geq& \frac{2 \cdot \beta \cdot T \cdot \max_{i\in[T]} \log(E_i) - \beta \cdot \log(E_k)}{T-1} \\
>& \frac{2 \cdot \beta \cdot (T-1) \cdot \max_{i\in[T]} \log(E_i)}{T-1} \\
=& 2 \cdot \beta \cdot \max_{i[T]} \log(E_i)
\end{aligned}$$

By this reasoning, because the *average* value across all other tasks is at least $2 \cdot \beta \cdot \max_{i[T]} \log(E_i)$, there must exist at least one task j with value

$$v_j \geq 2 \cdot \beta \cdot \max_{i[T]} \log(E_i)$$

Which implies that:

$$v_j + v_k \geq 2 \cdot \beta \cdot \max_{i[T]} \log(E_i) \geq \beta \cdot (\log(E_j) + \log(E_k))$$

If this task exists, then by the preconditions of Lemma 14, total winrate could be increased by equalizing winrate across both task j and k. □

Lemma 5. *Suppose there is a set of existing models and tasks with values $\{\vec{v}_t\}$. Then, an incoming model with total value V optimizing for weighted winrate would best respond by putting all value on a single task (the task that maximizes $p_{N+1}(\vec{v}_i \oplus V)$), and value 0 elsewhere.*

Proof. In order to prove this, we will again consider any arrangement $\{\vec{v}_t\}$ with positive value across multiple tasks and show that the alternative arrangement $\{\vec{v}'_t\}$ with value on only a single task strictly improves the $\sum_{t\in[T]} v_{ct} \cdot p_c(\vec{v}_t)$ objective. Specifically, we will assume that the allocation has positive value only on the task with the smallest current total logexp sum:

$$\text{argmin}_{t\in[T]} \sum_{i\in[N]} \exp(\vec{v}_{i,t}/\beta) = \text{argmin}_{t\in[T]} E_t$$

WLOG, call this task 1. The condition we wish to show is, for new agent c:

$$\begin{aligned}
\sum_{t\in[T]} \vec{v}'_{c,t} \cdot p_c(\vec{v}'_t) &> \sum_{t\in[T]} \vec{v}_{c,t} \cdot p_c(\vec{v}_t) \\
V_c \cdot p_c(\vec{v}'_1) + 0 \cdot \sum_{t\neq 1} 0 \cdot p_c(\vec{v}'_t) &> \sum_{t\in[T]} \vec{v}_{c,t} \cdot p_c(\vec{v}_t) \\
\sum_{t\in[T]} \vec{v}_{c,t} \cdot p_c(\vec{v}'_1) &> \sum_{t\in[T]} \vec{v}_{c,t} \cdot p_c(\vec{v}_t) && \text{(Rewriting } V_c\text{)} \\
\sum_{t\in[T]} \vec{v}_{c,t} \cdot (p_c(\vec{v}'_1) - p_c(\vec{v}_t)) &> 0
\end{aligned}$$

This last inequality holds because if $p_c(\vec{v}'_1) - p_c(\vec{v}_t)$ for any $t \in [T]$ by design:

$$\begin{aligned}
p_c(\vec{v}'_1) &\geq p_c(\vec{v}_t) \\
\frac{\exp(V/\beta)}{\exp(V/\beta) + E_1} &\geq \frac{\exp(v_{ct}/\beta)}{\exp(v_{ct}/\beta) + E_t} && (E_t = \textstyle\sum_{i\neq c, i\in[N]} \exp(\vec{v}_{it}/\beta)) \\
\exp((V + v_{ct})/\beta) + \exp(V/\beta) \cdot E_t &\geq \exp((V + v_{ct})/\beta) + E_1 \cdot \exp(v_{ct}/\beta) \\
\exp(V/\beta) \cdot E_t &\geq E_1 \cdot \exp(v_{ct}/\beta)
\end{aligned}$$

This holds by a pair of inequalities: we know that $\exp(V_c/\beta) \geq \exp(\vec{v}_{c,t}/\beta)$ because $V_c \geq \vec{v}_{c,t}$. Additionally, we know that $E_t \geq E_1$ by the definition of task 1. □

Lemma 15. *Consider a set of tasks $T \geq 1$ with constant value $\{\vec{v}\}$ across tasks. Then, whenever:*

$$V \leq T \cdot \beta \cdot \log\left(\sum_{i\in|\vec{v}|} \exp(\vec{v}_i/\beta)\right)$$

equalizing winrate across tasks does not *maximize total winrate.*

Proof. Consider the allocation that equalizes winrate: because each task is identical, this must also involve equalizing value across tasks. Because

$$V < T \cdot \beta \cdot \log\left(\sum_{i\in|\vec{v}|} \exp(\vec{v}_i/\beta)\right) = T \cdot \beta \cdot (\log(E))$$

we know that this allocation must have $v_i = V/T < \beta \cdot \log(E)$. By Lemma 14, we know that this means that for any two tasks, total winrate can be improved by placing all value on a single task, and none on another. This means that equalizing winrate across tasks cannot be the allocation that maximizes total winrate. □

Theorem 4. *Whenever agents would respond to winrate by equalizing winrate (conditions of Theorem 3), weighted winrate would lead to strictly higher consumer welfare than winrate.*

Proof. By Lemma 12, the change in consumer welfare when a new agent with value v' joins is given by:

$$u(\vec{v} \oplus v') - u(\vec{v}) = p_N(\vec{v} \oplus v') \cdot (\vec{v}' - u(\vec{v}))$$

where $\vec{v}$ is the vector of values for the task before a new agent joins, and $p_N(\cdot)$ gives the probability of the new (Nth) agent's response being picked.

Using the results of Theorem 3, agents best-responding to winrate would spread their value V across tasks so as to equalize winrate, while by Lemma 5, for weighted winrate they would put all value on a single task, and value 0 elsewhere.

Denote $\vec{v}_i \oplus 0, \vec{v}_i \oplus V, \vec{v}_i \oplus v_i$ as the allocations where the new agent allocates $0, V, v_i$ respectively.

The change in utility for winrate is:

$$\sum_{i\in[T]} p_N(\vec{v}_i \oplus v_i)\cdot(v_i - u(\vec{v}_i)) = \sum_{i\in[T]} p_N^*\cdot(v_i - u(\vec{v}_i))$$

where in the second step we have used that $p(\cdot)$ is constant for all tasks with winrate best response. The change in utility for weighted winrate is:

$$p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot(0 - u(\vec{v}_i))$$

where without loss of generality we have assumed that the task that weighted winrate places all weight on is task 1. Recall from Lemma 5 that this task must be the task that maximizes $p_N(\vec{v}_1 \oplus V)$. Weighted winrate is better whenever:

$$\begin{aligned}
p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot(0 - u(\vec{v}_i)) &\geq \sum_{i\in[T]} p_N^*\cdot(v_i - u(\vec{v}_i)) \\
p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) - \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot u(\vec{v}_i) &\geq \sum_{i\in[T]} p_N^*\cdot(v_i - u(\vec{v}_i)) \\
p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) &\geq \sum_{i\in[T]} p_N^*\cdot(v_i - u(\vec{v}_i)) \\
&\quad + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot u(\vec{v}_i) \\
p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) - \sum_{i\in[T]} p_N^*\cdot v_i &\geq -\sum_{i\in[T]} p_N^*\cdot u(\vec{v}_i) + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot u(\vec{v}_i) \\
p_N(\vec{v}_1 \oplus V)\cdot(V - u(\vec{v}_1)) - p_N^*\cdot V &\geq -\sum_{i\in[T]} p_N^*\cdot u(\vec{v}_i) + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot u(\vec{v}_i) \\
V\cdot(p_N(\vec{v}_1 \oplus V) - p_N^*) &\geq p_N(\vec{v}_1 \oplus V)\cdot u(\vec{v}_1) - \sum_{i\in[T]} p_N^*\cdot u(\vec{v}_i) \\
&\quad + \sum_{i\in[T],i\neq 1} p_N(\vec{v}_i \oplus 0)\cdot u(\vec{v}_i) \\
V\cdot(p_N(\vec{v}_1 \oplus V) - p_N^*) &\geq u(\vec{v}_1)\cdot(p_N(\vec{v}_1 \oplus V) - p_N^*) \\
&\quad + \sum_{i\in[T],i\neq 1} (p_N(\vec{v}_i \oplus 0) - p_N^*)\cdot u(\vec{v}_i)
\end{aligned}$$

Note that $p_N(\vec{v}_i \oplus 0) \leq p_N^*$, and thus we can drop the last term from the summation on the righthand side because it is always negative, in order to upper bound the term on the righthand side.

$$V\cdot(p_N(\vec{v}_1 \oplus V) - p_N^*) \geq \cdot u(\vec{v}_1)\cdot(p_N(\vec{v}_1 \oplus V) - p_N^*)$$

This holds whenever $V \geq u(\vec{v}_1)$, which is true by the assumption of Theorem 3 (when agents would spread out for winrate), which holds that:

$$V \geq 2\cdot T\cdot\beta\cdot\max_{j\in[T]}\log\left(\sum_{i\in|\vec{v}_j|}\exp(\vec{v}_{i,j}/\beta)\right)$$

Note that by properties of the logexpsum, we have that:

$$\log\left(\sum_{i\in|\vec{v}_j|}\exp(\vec{v}_{i,j}/\beta)\right) \geq \max_{k\in|\vec{v}_j|}\vec{v}_{k,j}/\beta$$

Additionally, by the properties of BTL, we know that $u(\vec{v}_i) \leq \max_{k \in |\vec{v}_i|} \vec{v}_{k,i}$. Thus, we know that:

$$\begin{aligned} V \geq & 2 \cdot T \cdot \beta \cdot \max_{j \in [T]} \log \left(\sum_{i \in |\vec{v}_j|} \exp(\vec{v}_{i,j}/\beta) \right) \\ > & 2 \cdot T \cdot \beta \cdot \max_{j \in [T]} \cdot \max_{k \in |\vec{v}_j|} \vec{v}_{k,j}/\beta \\ \geq & 2 \cdot T \cdot \max_{j \in [T]} u(\vec{v}_j) \end{aligned}$$

which implies $V \geq u(\vec{v}_1)$, as desired. □

We note that from Lemmas 4 and 5, optimizing for consumer welfare and for weighted winrate would both incentivize producers to place all value on a single task. However, it is *not* the case that maximizing for weighted winrate and consumer welfare would both incentivize placing weight on the *same* task: it could be that they would pick different tasks $i \neq j$. For intuition, weighted winrate incentivizes producers to focus on a task where they maximize their probability of being selected, while consumer welfare would be maximized by picking a task that maximizes the probability of being picked, multiplied by the gap in value $V - u(v_i)$. Lemma 16 provides a bound on the gap in social welfare induced by this slight misalignment in objectives.

Lemma 16. *Given a set of existing tasks with values $\{\vec{v}_t\}$, weighted winrate and consumer welfare may* not *incentivize picking the same task to place all value on. However, the suboptimality of weighted winrate* on this task *is bounded by $u(\vec{v}_{i^*}) - u(\vec{v}_{j^*})$, where i^* is the index of the task that maximizes the probability $p_{N+1}(\vec{v}_i \oplus V)$, and j^* is the index of the task with smallest current value $u(\vec{v}_{j^*})$.*

Proof. WLOG suppose that task 1 is the task that maximizes consumer welfare, and task 2 is the one that maximizes weighted winrate. Note from Lemma 5 that we must have that task 2 maximizes $p_{N+1}(\vec{v}_i \oplus V)$. The gap in consumer welfare is given by:

$$\begin{aligned} & p_{N+1}(\vec{v}_1 \oplus V) \cdot (V - u(\vec{v}_1)) - p_{N+1}(\vec{v}_2 \oplus V) \cdot (V - u(\vec{v}_2)) \\ < & p_{N+1}(\vec{v}_2 \oplus V) \cdot (V - u(\vec{v}_1)) - p_{N+1}(\vec{v}_2 \oplus V) \cdot (V - u(\vec{v}_2)) \\ = & p_{N+1}(\vec{v}_2 \oplus V) \cdot (V - u(\vec{v}_1) - V + u(\vec{v}_2)) \\ = & p_{N+1}(\vec{v}_2 \oplus V) \cdot (u(\vec{v}_2) - u(\vec{v}_1)) \\ \leq & u(\vec{v}_2) - u(\vec{v}_1) \end{aligned}$$

□

Lemma 6. *If each task i has a maximum value v_i^*, the allocation that maximizes weighted winrate or consumer welfare is to greedily allocate value for each task according to the objectives in Lemmas 4 or 5 respectively.*

Proof. This can be seen almost immediately from the proofs of Lemmas 4 or 5: the final allocation is one that allocates up to the maximum value on the task v_i^* sequentially according to which task maximizes that objective. This is optimal by the same reasoning with proofs of Lemmas 4 and 5: any other allocation could be strictly improved by re-allocating value to tasks that are higher on that objective. Finally, note that for consumer welfare objective, if the "remainder" of value is less than the current aggregate value on a task $u(\vec{v}_i)$, then the optimal choice is to abstain. □

E Proofs for Section 6

Definition 11. *For model replacement, instantaneous consumer welfare is the* sum *of change in consumer welfare, given the instantaneous change in all producers' values over tasks. That is, if t_j^* is the task that producer j chooses to improve in, then instantaneous consumer welfare is given by:*

$$\sum_{i \in [N]} \frac{\partial}{\partial \vec{v}_{i,t_j^*},} u(\vec{v}_{t_j^*}) = \sum_{i \in [N]} \frac{\partial}{\partial \vec{v}_{i,t_j^*},} \vec{v}_{i,t_j^*} \cdot p_i(\vec{v}_{t_j^*})$$

Lemma 17. *Given a fixed task i, the instantaneous change in the producer's objective given a change in $\vec{v}_i$ is given by:*

- *For winrate:* $p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$.
- *For weighted winrate:* $p_i(\vec{v}) + \vec{v}_i \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$.
- *For consumer welfare, with two agents:* $p_i(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$.

Proof. This can be seen simply by taking the derivative of each objective.

For winrate,

$$\frac{\partial}{\partial \vec{v}_i} p_i(\vec{v}) = p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$$

For weighted winrate,

$$\frac{\partial}{\partial \vec{v}_i} \vec{v}_i \cdot p_i(\vec{v}) = p_i(\vec{v}) + \vec{v}_i \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$$

For consumer welfare,

$$\begin{aligned}
\frac{\partial}{\partial \vec{v}_i} u(\vec{v}) =& \frac{\partial}{\partial \vec{v}_i} \sum_{j \in |\vec{v}|} \vec{v}_j \cdot p_j(\vec{v}) \\
=& \frac{\partial}{\partial \vec{v}_i} \vec{v}_i \cdot p_i(\vec{v}) + \frac{\partial}{\partial \vec{v}_i} \sum_{j \neq i} \vec{v}_j \cdot p_j(\vec{v}) \\
=& p_i(\vec{v}) + \vec{v}_i \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta - \vec{v}_j \cdot p_j(\vec{v}) \cdot p_i(\vec{v})/\beta \qquad \text{(Properties of BTL)}
\end{aligned}$$

For the case with $N = 2$, the above term reduces to:

$$p_i(\vec{v}) + \vec{v}_i \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta - \vec{v}_j \cdot (1 - p_i(\vec{v})) \cdot p_i(\vec{v})/\beta = p_i(\vec{v}) + (\vec{v}_i - \vec{v}_j) \cdot p_i(\vec{v}) \cdot (1 - p_i(\vec{v}))/\beta$$

as desired. □

Theorem 5. *For $N = 2$ agents optimizing for winrate, agents would always pick the* same *task to increase value in, whereas agents optimizing for consumer welfare always pick* different *tasks.*

Proof. This is proved through Lemmas 18 and 19. □

Lemma 18. *For $N = 2$ agents optimizing for winrate, agents would always pick the same task to increase value in.*

Proof. If there are only two agents, then they split the winrate, so $p_1(\vec{v}_t) = 1 - p_2(\vec{v}_t)$. This immediately implies that $\frac{\partial}{\partial v_1} p_1(\vec{v}_t) = \frac{\partial}{\partial v_2} p_2(\vec{v}_t)$, which means that both producers have the exact same preferences over all tasks to improve in. □

Lemma 19. *For $N = 2$ agents, if they are optimizing for consumer welfare, then they would* never *prefer to pick the same task.*

Proof. WLOG, suppose that agent A would increase value in task 1. This means that:

$$\begin{aligned}
\frac{\partial}{\partial \vec{v}_{a,1}} u(\vec{v}_1) \geq& \frac{\partial}{\partial \vec{v}_{a,2}} u(\vec{v}_2) \\
\frac{\partial}{\partial \vec{v}_{a,1}} \vec{v}_{a,1} \cdot p_a(\vec{v}_1) + \vec{v}_{b,1} \cdot (1 - p_a(\vec{v}_1)) \geq& \frac{\partial}{\partial \vec{v}_{a,2}} \vec{v}_{a,2} \cdot p_a(\vec{v}_2) + \vec{v}_{b,2} \cdot (1 - p_a(\vec{v}_2)) \\
p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \geq& p_a(\vec{v}_2) + (\vec{v}_{a,2} - \vec{v}_{b,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta
\end{aligned}$$

Note by symmetry, the condition where agent B would prefer to increase value in task 2 is given by:

$$\begin{aligned}
p_b(\vec{v}_1) + (\vec{v}_{b,1} - \vec{v}_{a,1}) \cdot p_b(\vec{v}_1) \cdot (1 - p_b(\vec{v}_1))/\beta \leq & p_b(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta \\
1 - p_a(\vec{v}_1) + (\vec{v}_{b,1} - \vec{v}_{a,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \leq & 1 - p_a(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta \\
& (p_a(\cdot) = 1 - p_b(\cdot)) \\
-p_a(\vec{v}_1) - (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \leq & - p_a(\vec{v}_2) - (\vec{v}_{a,2} - \vec{v}_{b,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta \\
p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \geq & p_a(\vec{v}_2) + (\vec{v}_{a,2} - \vec{v}_{b,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta
\end{aligned}$$

which is exactly equal to the condition when player A would prefer task 1 over 2.
This tells us that for any pair of tasks i, j, agents A and B would have exactly inverted preferences over which task they would prefer to improve in. Applied to all T tasks, this says that agents A and B would have exactly inverted orders over which all tasks they would choose to improve in. □

Lemma 7. *For $N = 2$ agents responding to weighted winrate, agents are incentivized to improve in* different tasks *whenever agents begin by being sufficiently specialized (e.g. $|p_a(\vec{v}_i) - p_a(\vec{v}_j)|$ is sufficiently large).*

Proof. Given any two tasks $1, 2$, the condition where player A prefers to increase value in task 1 occurs exactly when:

$$\begin{aligned}
\frac{\partial}{\partial \vec{v}_{a,1}} \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \geq & \frac{\partial}{\partial \vec{v}_{a,2}} \vec{v}_{a,2} \cdot p_a(\vec{v}_2) \\
p_a(\vec{v}_1) + \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1)/\beta \geq & p_a(\vec{v}_2) + \vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)/\beta \\
p_a(\vec{v}_1) - p_a(\vec{v}_2) \geq & \left(\vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) - \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))\right)/\beta := A_{12}
\end{aligned}$$

Note by identical reasoning the condition where player B prefers to increase value in task 1 is given by:

$$\begin{aligned}
p_b(\vec{v}_1) - p_b(\vec{v}_2) \geq & \left(\vec{v}_{b,2} \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2)) - \vec{v}_{b,1} \cdot p_b(\vec{v}_1) \cdot (1 - p_b(\vec{v}_1))\right)/\beta \\
-p_a(\vec{v}_1) + p_a(\vec{v}_2) \geq & \left(\vec{v}_{b,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) - \vec{v}_{b,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))\right)/\beta \\
& (p_a(\cdot) = 1 - p_b(\cdot)) \\
p_a(\vec{v}_1) - p_a(\vec{v}_2) \leq & -\left(\vec{v}_{b,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) - \vec{v}_{b,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))\right)/\beta := B_{12}
\end{aligned}$$

Thus, the condition where player A and player B both prefer task 1 to 2 is when:

$$A_{12} < |p_a(\vec{v}_1) - p_a(\vec{v}_2)| = |p_b(\vec{v}_1) - p_b(\vec{v}_2)| < B_{12}$$

The condition where player A and player B both prefer task 2 to 1 is when:

$$B_{12} < |p_a(\vec{v}_1) - p_a(\vec{v}_2)| = |p_b(\vec{v}_1) - p_b(\vec{v}_2)| < A_{12}$$

And the condition where they must have different preferences is when:

$$\max(A_{12}, B_{12}) < |p_a(\vec{v}_1) - p_a(\vec{v}_2)| = |p_b(\vec{v}_1) - p_b(\vec{v}_2)|$$

Finally, note that the scenario where players both pick the same task (e.g. task 1), occurs when:

$$p_a(\vec{v}_1) + \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1)/\beta \geq p_a(\vec{v}_2) + \vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)/\beta$$

and:

$$p_b(\vec{v}_2) + \vec{v}_{b,2} \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2)/\beta \leq p_b(\vec{v}_1) + \vec{v}_{b,1} \cdot p_b(\vec{v}_1) \cdot (1 - p_b(\vec{v}_1)/\beta$$

and implies:

$$(\vec{v}_{a,1} + \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1)/\beta \geq (\vec{v}_{a,2} + \vec{v}_{b,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)/\beta$$

□

This table shows an example where weighted winrate does lead to increased specialization: winrate would incentivize players to improve on the same task and weighted winrate would incentivize producers to improve on different tasks. However, this is the *wrong* type of specialization: consumer welfare would be higher if each producer swapped which task they improved on, and in fact, would be higher if producers picked the same task to improve on, as given by winrate.

	1	2
A	74.3	16.45
B	70.4	14.34

(a) Value: player A has higher value on both tasks.

	1	2
A	0.029	**0.0979**
B	0.029	**0.0979**

(b) Winrate (derivative).

	1	2
A	**2.48**	2.47
B	1.44	**1.48**

(c) Weighted winrate (derivative).

	1	2
A	1.057	**1.095**
B	**-0.057**	-0.095

(d) Consumer welfare (derivative)

Table 7: Case with two producers A, B picking between two tasks $1, 2$ (maximum value in each row **bolded** for convenience). Here, note that producers A, B each pick different tasks $1, 2$ according to weighted winrate, but end up resulting in worse instantaneous consumer welfare than if they had picked according to winrate. Note that player B is dominated in both tasks, and in particular, on both tasks it has *negative* derivative for consumer welfare. That is, no matter which task player B picked, its increase in value strictly harms consumer welfare. Note all quantities are rounded.

Theorem 6. *Consider a model replacement setting with $N = 2$ agents: the only scenario where weighted winrate causes* worse *instantaneous consumer welfare than winrate is when one player is in the non-monotonic regime on* both *picked tasks.*

Proof. WLOG, we assume agent A picks task 1 and agent B picks task 2. We will begin by eliminating conditions where weighted winrate improves consumer welfare, and then prove that in all remaining situations, one player must be in the non-monotonic regime on both picked tasks.

By results from other lemmas, we know that the following situations are ones where weighted winrate strictly improves consumer welfare over winrate:

1. (By Lemma 22): Both players get higher winrate in picking their task: $p_b(\vec{v}_2) > p_b(\vec{v}_1)$ and $p_a(\vec{v}_1) > p_a(\vec{v}_2)$.
2. (By Lemma 21): At least one agent has winrate < 0.5 on the task that it picks.

Combining both facts, this must mean that it has winrate < 0.5 on both tasks, and the other agent has winrate > 0.5 on both tasks. WLOG, assume that it is agent A that does worse. Note that this implies $\vec{v}_{bi} > \vec{v}_{ai}$ for $i = 1, 2$. When is the case where we get *worse* total sum of derivatives by using weighted winrate?

$$\begin{aligned}
\frac{\partial}{\partial \vec{v}_{a,1}} u(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_{b,2}} u(\vec{v}_2) <& 1 \\
p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta & \\
+ p_b(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta < & 1 \\
p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta & \\
+ 1 - p_a(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta < & 1 \\
p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1})/\beta \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1)) <& p_a(\vec{v}_2) + (\vec{v}_{a,2} - \vec{v}_{b,2})/\beta \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2)) \\
p(\Delta_1) + \Delta_1/\beta \cdot p(\Delta_1) \cdot (1 - p(\Delta_1)) <& p(\Delta_2) + \Delta_2/\beta \cdot p(\Delta_2) \cdot (1 - p(\Delta_2)) \\
f(\Delta_1) <& f(\Delta_2)
\end{aligned}$$

We will apply results from the lemmas above to tell us more about when the condition above holds. Consider the function $f(x)$ as defined above. Because the winrate of agent a is smaller in task 2, and player A has winrate < 0.5 in each of them, we know that $\Delta_2 < \Delta_1 < 0$.

By Lemma 23, we know that the function $f(x)$ has a single point of 0 derivative for $x < 0$, and the regime where $f(x)$ is decreasing in x (and where $p(x) < 0.5$) is also one where $f(x) < 0$.

Applying this here, we know that $\Delta_2 < \Delta_1$, and yet $f(\Delta_2) > f(\Delta_1)$. Because there is only a single point of 0 derivative, we know that this means that $f'(\Delta_2) < 0$, and thus $f(\Delta_2) < 0$, which means that $f(\Delta_1) < 0$ as well, as desired. □

Lemma 20. *Given the setting of Theorem 6, whenever both agents pick the same task to improve on, the change in consumer welfare is constant (equal to 1).*

Proof. Denote the players as A and B, and WLOG label the task they both pick as 1. Then, the instantaneous consumer welfare is given by:

$$\begin{aligned}
&\frac{\partial}{\partial \vec{v}_{a,1}} u(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_{b,1}} u(\vec{v}_1) \\
=&p_a(\vec{v}_1) + (\vec{v}_{1a} - \vec{v}_{1b}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \\
&+ p_b(\vec{v}_1) + (\vec{v}_{1b} - \vec{v}_{1a}) \cdot p_b(\vec{v}_1) \cdot (1 - p_b(\vec{v}_1))/\beta && \text{(Value of derivative)} \\
=&1 - p_b(\vec{v}_1) + (\vec{v}_{1a} - \vec{v}_{1b}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \\
&+ p_b(\vec{v}_1) + (\vec{v}_{1b} - \vec{v}_{1a}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta && (p_a(\vec{v}_1) = 1 - p_b(\vec{v}_1)) \\
=&1
\end{aligned}$$

□

Lemma 21. *Given the setting of Theorem 6, if both players pick tasks where they have winrate at least 0.5, then they're doing strictly better than if they both picked the same task.*

Proof. Denote the players as A and B, and WLOG assume that A prefers task 1, and B prefers task 2. The instantaneous consumer welfare is given by:

$$\begin{aligned}
&\frac{\partial}{\partial \vec{v}_{a,1}} u(\vec{v}_1) + \frac{\partial}{\partial \vec{v}_{b,2}} u(\vec{v}_2) \\
=&p_a(\vec{v}_1) + (\vec{v}_{1a} - \vec{v}_{1b}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \\
&+ p_b(\vec{v}_2) + (\vec{v}_{2b} - \vec{v}_{2a}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta && \text{(Value of derivative)} \\
=&p_a(\vec{v}_1) + p_b(\vec{v}_2) \\
&+(\vec{v}_{1a} - \vec{v}_{1b}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta + p_b(\vec{v}_2) + (\vec{v}_{2b} - \vec{v}_{2a}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta \\
>&1 && (p_a(\vec{v}_1), p_b(\vec{v}_2) > 0.5) \\
&+(\vec{v}_{1a} - \vec{v}_{1b}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta + p_b(\vec{v}_2) + (\vec{v}_{2b} - \vec{v}_{2a}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta \\
>&1 && ((\vec{v}_{1a} - \vec{v}_{1b}) > 0, (\vec{v}_{2b} - \vec{v}_{2a}) > 0)
\end{aligned}$$

as desired. □

Lemma 22. *Consider the setting of Theorem 6, and assume that at least one player has* lower *winrate on the task that it picks (e.g., $p_b(\vec{v}_1) > p_b(\vec{v}_2)$, if player B picks task 2). Then, instantaneous consumer welfare for this setting is at least as large as if both players picked the same task.*

Proof. First, let's write out multiple conditions. We know that players A and B prefer tasks 1 and 2. We also assume that at least one player has *lower* winrate on the task that it prefers: WLOG, we will assume that this is player B or:

$$p_b(\vec{v}_1) > p_b(\vec{v}_2) \tag{9}$$

If we have that player A prefers task 1, then we know that:

$$\begin{aligned}
p_a(\vec{v}_1) + \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta >& p_a(\vec{v}_2) + \vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) \\
p_a(\vec{v}_1) - p_a(\vec{v}_2) >& \vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) - \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta
\end{aligned} \tag{10}$$

If we know that player B prefers task 2, we know that:

$$\begin{aligned}
p_b(\vec{v}_1) + \vec{v}_{b,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta <& p_b(\vec{v}_2) + \vec{v}_{b,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta \\
p_b(\vec{v}_1) - p_b(\vec{v}_2) <& \vec{v}_{b,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta - \vec{v}_{b,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta
\end{aligned} \tag{11}$$

Note that by Equation 9, we know that the LHS of Equation 11 is positive, and therefore the RHS is also positive. Next, the condition we want to prove is that instantaneous consumer welfare for having player A pick task 1 and player B pick task 2 is at least as large as if they both picked the same task (which gives consumer welfare 1). That is, we want to show:

$$\begin{aligned}
&p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \\
&+p_b(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_b(\vec{v}_2) \cdot (1 - p_b(\vec{v}_2))/\beta > 1 \\
\Leftrightarrow\quad &p_a(\vec{v}_1) + (\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta \\
&+1 - p_a(\vec{v}_2) + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta > 1 && (p_b(\vec{v}_2) = 1 - p_a(\vec{v}_2)) \\
\Leftrightarrow\quad &p_a(\vec{v}_1) - p_a(\vec{v}_2) \\
&+(\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta > 0 \\
\Leftarrow\quad &\vec{v}_{a,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2)) - \vec{v}_{a,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta && \text{(By Eq. 10)} \\
&+(\vec{v}_{a,1} - \vec{v}_{b,1}) \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta + (\vec{v}_{b,2} - \vec{v}_{a,2}) \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta > 0 \\
\Leftrightarrow\quad &- \vec{v}_{b,1} \cdot p_a(\vec{v}_1) \cdot (1 - p_a(\vec{v}_1))/\beta + \vec{v}_{b,2} \cdot p_a(\vec{v}_2) \cdot (1 - p_a(\vec{v}_2))/\beta > 0
\end{aligned}$$

Note that the last condition is satisfied by Equation 11: knowing that the RHS of this equation is positive is equivalent to satisfying the condition above. □

Lemma 23. *Consider the function $f(x, \beta) = p(x/\beta) + x/\beta \cdot p(x/\beta) \cdot (1 - p(x/\beta))$, where $p(x/\beta)$ gives the BTL probability of picking an item, given a gap in values of x and noise level β. Then, a) $f(x, \beta)$ has a single point of 0 derivative for $x < 0$, and b) whenever $f'(x, \beta) < 0$ and $p(x/\beta) < 0.5$, we know that $f(x, \beta) < 0$.*

Proof. First, we calculate the derivative of $f(x)$ (for simplicity, replacing $p = p(x/\beta)$):

$$\begin{aligned}
f'(x) =&p \cdot (1 - p)/\beta + p \cdot (1 - p)/\beta + p/\beta \cdot (p \cdot (1 - p)^2/\beta + p \cdot (0 - p \cdot (1 - p))/\beta) \\
=&2 \cdot p \cdot (1 - p)/\beta + x/\beta^2 \cdot p \cdot ((1 - p)^2 - p \cdot (1 - p)) \\
=&2 \cdot p \cdot (1 - p)/\beta + x/\beta^2 \cdot p \cdot (1 - p) \cdot (1 - p) - p) \\
=&p \cdot (1 - p)/\beta \cdot (2 + x/\beta \cdot (1 - 2p))
\end{aligned}$$

This is negative if $2 + x/\beta \cdot (1 - 2p) < 0$. First, we will note that there is only a single point of zero derivative when $x < 0$. This is because the derivative is zero whenever:

$$\begin{aligned}
2 + x/\beta \cdot (1 - 2p) &= 0 \\
1 - 2 \cdot p &= -2/(x/\beta) \\
-2 \cdot p &= -1 - 2/(x/\beta) \\
p(x/\beta) &= 0.5 + 1/(x/\beta)
\end{aligned}$$

where in the last line, we have added in the fact that $p(\cdot)$ is a function of x/β. First, we note that $p(x/\beta)$ is always increasing in x. Second we note that $1/x$ is negative for $x < 0$. As x increases (becomes less negative), $1/|x|$ increases, and thus $1/x$ *decreases*. Thus, because the LHS is *increasing* in x and the RHS is *decreasing* in x, they can only have a single point where they cross for $x < 0$.

Next, we want to show that if $f'(x) < 0$ and $x < 0$, then $f(x) < 0$. Note that $f(x) < 0$ if:

$$\begin{aligned}
p + x/\beta \cdot p(1 - p) &< 0 \\
1 + x/\beta \cdot (1 - p) &< 0
\end{aligned}$$

Note that we have:

$$0 > 2 + x/\beta \cdot (1 - 2p) > 1 + x/\beta \cdot (1 - 2p) > 1 + x/\beta \cdot (1 - p)$$

where the last line is because $0 < 1 - 2 \cdot p < 1 - p$, because $p < 0.5$ by assumption. □

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our main claim is that winrate can incentivize homogenization, and weighted winrate improves incentives: these are proven in the relevant sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: These are discussed in Section 2, which describes our model and key assumptions. We also discuss generalizations in Section 8.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: These are discussed in Section 2, which describes our model and key assumptions. All proofs are in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The appendix describes each experiment in detail. We either provide data directly or link to the data source. In the final paper we will provide all code to reproduce experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code will be released upon a final version.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: These are only relevant for Appendix A.2, where we're reproducing results in an existing paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: No results involve or require error bars.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: All experiments can be done easily on a laptop.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: There are no human experiments or novel data, and our core goal of our research is to improve consumer welfare.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Discussed in the introduction, conclusion, and model section.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: Not relevant for given experiments.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All data and results are cited when appropriate.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: No new assets are created.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: No human experiments.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No human experiments.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs only used for minor editing and literature review.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.