STRATEGIC COMPETITION AND CONVERGENCE IN LLM LEADERBOARDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Leaderboards are a popular way to compare large language model (LLM) capabilities and can influence how model developers invest their resources. We ask when competition on head-to-head leaderboards leads developers to produce similar models, which we call monoculture, or models specialized in different capabilities. We study this question by formulating leaderboard competition as a game in which model developers invest resources across capabilities to maximize their win rate. We show that similar production functions can induce monoculture, while comparative advantage can sustain specialization. Turning to how competition evolves over time, we give conditions under which diminishing returns and model catch-up lead to crowding at the frontier. Using data from LM Arena and OpenRouter, we find evidence of both competitive catch-up and persistent differentiation: developers catch up where they lag while retaining category-specific advantages, alongside an increasingly crowded frontier and dispersed model usage. Together, our results characterize conditions under which competitive investment can generate convergence while preserving differences.

1 INTRODUCTION

Leaderboards are one of the main ways in which large language models (LLM) are measured and can influence how models are developed (Chan, 2026). While benchmark-style leaderboards remain popular, newer head-to-head leaderboards compare models based on aggregated human preferences (Chiang et al., 2024). This changes strategic incentives for developers, as a model's rank depends on competitors' capabilities (Akhtar et al., 2026).

A natural question is whether this competition pushes developers toward similar models or instead preserves specialization. On one hand, a common evaluation provides developers with the same target, potentially pushing them toward similar capability profiles, or *monoculture* (Kleinberg & Raghavan, 2021). Importantly, model similarity need not arise from competition alone, as developers with similar underlying technologies may produce similar models even absent strategic convergence. By contrast, differences in developers' strengths across capabilities can create incentives to specialize. This distinction matters because leaderboards may shape not only which models perform best, but also the direction of model development (Dominguez-Olmedo et al., 2025; Chen et al., 2026). The resulting diversity of model capabilities can affect whether users have access to models with meaningfully different strengths or an increasingly homogeneous set of alternatives.

These forces connect to two literatures. Contest theory studies how competitors allocate costly effort in response to a common reward (Vojnović, 2015; Roberson, 2006), while algorithmic monoculture studies when shared algorithms lead to similar outcomes (Kleinberg & Raghavan, 2021; Peng & Garg, 2024). LLM leaderboards combine aspects of both settings: developers compete against one another under a common evaluation, but allocate investment across multiple capabilities. Our focus differs from recent work on strategic behavior in LLM evaluation. While Donahue & Raghavan (2026) studies how routers aggregate competing models and Chen et al. (2026) studies competition over leaderboard rank, we study how competition shapes developers' investment across capabilities and, consequently, whether developers produce similar or specialized capability profiles.

We study these questions by modeling AI evaluation as a multidimensional contest where developers allocate investment across capabilities (Figure 1). Our results reveal a tension between convergence

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

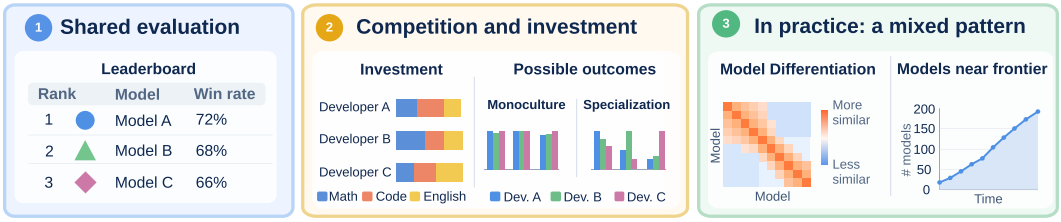

Figure 1: Shared evaluation shapes competitive investment. Head-to-head leaderboards compare models, creating incentives for developers to allocate investment across capabilities. How these incentives interact with developers' production functions can lead to monoculture or specialization. We find competitive catch-up alongside differentiation, together with a crowded model frontier.

and differentiation; competition can push developers toward a common performance frontier without eliminating differences in their capability profiles. Our contributions include:

1. We characterize when multidimensional competition under shared evaluation produces monoculture or specialization. Similar production functions can induce monoculture, while comparative advantage can sustain specialization.
2. We characterize how competition evolves over time and show how diminishing returns and models catching up can lead to crowding at the frontier.
3. Using longitudinal data from LM Arena and OpenRouter, we document competitive catch-up alongside persistent differentiation and a crowded performance frontier.

2 Model and Preliminaries

Our model consists of N model developers (e.g., OpenAI), each of which competes on K dimensions (e.g., math). Developer i has a production function $F_i : \mathbb{R}_+^K \to \mathbb{R}_+^K$, and investment $\mathbf{x}_i$, which produces capabilities $\boldsymbol{\theta}_i := F_i(\mathbf{x}_i)$. Investment at level $\mathbf{x}_i$ results in a cost $c(\|\mathbf{x}_i\|_1)$, where c is a strictly increasing, differentiable, continuous, and unbounded cost function with $c(0) = 0$ (our model works even with per-developer cost functions c_i; see Appendix F.1). We model each developer as maximizing its expected win rate. This objective directly corresponds to maximizing weighted category-level pairwise win probabilities under Bradley–Terry/Elo-style evaluation (see Appendix A). Here, win-rate is weighted according to $\mathbf{w} \geq 0, \sum_k w_k = 1$. We define the utility as

$$u_i(\mathbf{x}) = \frac{1}{N-1} \sum_{i' \neq i} \sum_k w_k \sigma(F_i(\mathbf{x}_i)_k - F_{i'}(\mathbf{x}_{i'})_k) - c(\sum_k x_{i,k}).$$

Here, σ denotes the win-probability according to some pairwise preference model. We assume σ is strictly increasing, twice differentiable, bounded in $[0, 1]$, $\sigma'(x) > 0$, and symmetric where $\sigma(x)+\sigma(-x) = 1$ (e.g., Bradley-Terry). Our model abstracts from the details of developers' internal R&D processes and instead captures three features of model development: developers care about their capabilities relative to competitors, allocate costly investment across multiple capabilities, and differ in how effectively investment translates into capability improvements. We do not require that developers directly optimize a particular leaderboard; tournament-style evaluation provides a tractable representation of competition based on relative model performance. The model can also be viewed as a multidimensional contest, related to Colonel Blotto games (Roberson, 2006), in which investment is transformed into capabilities through heterogeneous production functions F_i.

Let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ be a Nash equilibrium, and let $H_{i,i'}(\mathbf{x}) = \cos(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{i'})$ be the pairwise monoculture for developers i and i', with $H_{i,i'}(\mathbf{x}) = 1$ indicating that $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_{i'}$ are parallel. We can view $\boldsymbol{\theta}_i$ as consisting of a direction $\mathbf{d}_i^* \in \Delta_{K-1}$ and a magnitude α_i^*, with only the direction impacting $H_{i,i'}$. We focus on cosine similarity here to capture capability profiles rather than capability levels; while two developers might differ in their overall level of output, they can still be similar if their relative category outputs are similar (Bommasani et al., 2022). We return to similarity in the overall level of output in Section 5. We define overall monoculture as $H(\mathbf{x}) = \frac{1}{\binom{N}{2}} \sum_{i<i'} H_{i,i'}(\mathbf{x})$, with $H = 1$ indicating that all developers have parallel $\boldsymbol{\theta}_i$. When exactly one capability vector is zero we define $H_{i,i'}(\mathbf{x}^*) = 0$, and when both are zero we define $H_{i,i'}(\mathbf{x}^*) = 1$.

3 Existence of Pure Strategy Equilibrium

We discuss when the competition model exhibits pure-strategy equilibrium, where we focus on pure strategies because they correspond to stable, deterministic investment profiles, which do not require developers to randomize. We first consider the intuition in the $N = 2, K = 1$ setting. A sufficient condition for existence is that diminishing returns in production are sufficiently strong relative to the curvature of the evaluation function. That is, when c is convex and $f_i'(x_i) > 0$, we can write this as

$$\sup_z \frac{\sigma''(z)}{\sigma'(z)} < -\frac{f_i''(x_i)}{f_i'(x_i)^2} \forall x_i.$$

The following theorem generalizes this condition to arbitrary N and K.

Theorem 1 (Equilibrium Existence)**.** *Suppose that $g_{i,k}(\mathbf{x}_i) = [F_i(\mathbf{x}_i)]_k$ is twice continuously differentiable and c convex. Let $\rho_\sigma := \sup_{z\in\mathbb{R}} \frac{\sigma''(z)}{\sigma'(z)}$. There is a pure-strategy Nash equilibrium if*

$$-\nabla^2 g_{i,k}(\mathbf{x}_i) \succeq \rho_\sigma \nabla g_{i,k}(\mathbf{x}_i) \nabla g_{i,k}(\mathbf{x}_i)^\top \qquad \forall\, \mathbf{x}_i,\ \forall\, i, k.$$

Equilibrium exists when developers' production functions have sufficiently strong diminishing returns relative to the curvature of the evaluation function. Since the win-rate payoff is at most 1, any investment with $c(\|\mathbf{x}_i\|_1) > 1$ is dominated by zero investment. By unboundedness of c, we can restrict each developer's strategy space to a compact convex set $\mathcal{X}_i = \{\mathbf{x}_i \geq 0 : \|\mathbf{x}_i\|_1 \leq \bar{x}\}$. Under the theorem's curvature condition, the Hessian of u_i is negative semidefinite, so u_i is concave in $\mathbf{x}_i$. Debreu–Fan–Glicksberg then gives existence (details in Appendix F.2). When production functions violate the conditions, pure-strategy equilibrium need not exist (see Appendix B).

4 When Does Monoculture Arise?

Given that a pure-strategy equilibrium exists, we next ask what determines whether developers produce similar capability profiles or specialize in different capabilities. We first give interpretable sufficient conditions for monoculture and specialization, before a general characterization.

4.1 Similar Production Functions Can Induce Monoculture

A simple sufficient condition for monoculture is developers having identical production functions.

Lemma 1 (Symmetric Developers)**.** *Suppose the conditions of Theorem 1 hold. If $F_i = F_{i'}$, then there exists a pure-strategy Nash equilibrium $\mathbf{x}^*$ such that $\mathbf{x}_i^* = \mathbf{x}_{i'}^*$ and $H_{i,i'}(\mathbf{x}^*) = 1$. Moreover, if $F_i = F$ for every developer i, then there exists a symmetric pure-strategy Nash equilibrium satisfying $\mathbf{x}_1^* = \cdots = \mathbf{x}_N^*$, and at this equilibrium, $\theta_1^* = \cdots = \theta_N^*$, with $H(\mathbf{x}^*) = 1$.*

Restricting attention to profiles with $\mathbf{x}_i = \mathbf{x}_{i'}$, the two developers have identical best-response correspondences; Kakutani's fixed-point theorem then gives the equilibrium (Appendix F.3).

Next, we note that strict equality is not necessary for near-monoculture. To establish this, we first introduce the *pseudogradient* of the game, $G(\mathbf{x})$, which we define as

$$G(\mathbf{x}) = \left(-\nabla_{\mathbf{x}_1} u_1(\mathbf{x}), \ldots, -\nabla_{\mathbf{x}_N} u_N(\mathbf{x})\right).$$

We next show that similarity in F_i and ∇F_i induces similarity in outputs.

Lemma 2 (Similarity in Production Functions)**.** *Suppose the conditions of Theorem 1 hold, while the production functions are L-Lipschitz, and the pseudogradient G is μ-strongly monotone, which ensures equilibrium stability to perturbations:*

$$\left(G(\mathbf{x}) - G(\mathbf{x}')\right)^\top (\mathbf{x} - \mathbf{x}') \geq \mu \|\mathbf{x} - \mathbf{x}'\|_2^2 \qquad \forall \mathbf{x}, \mathbf{x}'.$$

If developers i and i' have similar production functions in both value and gradient:

$$\sup_{\mathbf{x}_i \in \mathcal{X}_i} \|F_i(\mathbf{x}_i) - F_{i'}(\mathbf{x}_i)\|_2 \leq \epsilon, \qquad \sup_{\mathbf{x}_i \in \mathcal{X}_i} \|\nabla g_{i,k}(\mathbf{x}_i) - \nabla g_{i',k}(\mathbf{x}_i)\|_2 \leq \epsilon \quad \forall k.$$

then there exists an equilibrium $\mathbf{x}^*$ that satisfies

$$\|\mathbf{x}_i^* - \mathbf{x}_{i'}^*\|_2 \leq O\left(\frac{\epsilon}{\mu}\right), \qquad \|F_i(\mathbf{x}_i^*) - F_{i'}(\mathbf{x}_{i'}^*)\|_2 \leq O\left(\epsilon\left(1 + \frac{L}{\mu}\right)\right).$$

Moreover, if $\|\boldsymbol{\theta}_i^*\|_2, \|\boldsymbol{\theta}_{i'}^*\|_2 \geq m > 0$*, then their pairwise monoculture satisfies*

$$H_{i,i'}(\mathbf{x}^*) \geq 1 - O\left(\frac{\epsilon^2}{m^2}\left(1 + \frac{L}{\mu}\right)^2\right).$$

When developers have similar production functions, their equilibrium investments are similar. Evaluation coordinates developers and makes it possible for them to optimally respond in the same way. The proof constructs a counterfactual game and uses strong monotonicity to translate perturbation in the pseudogradient into bounds on equilibrium outputs (Appendix F.4).

4.2 Comparative Advantage Can Induce Specialization

While similar production functions induce monoculture, comparative advantage, where developers' production functions favor different capabilities, can induce specialization.

Example 1. *Suppose* $N = K = 2$ *with production functions of the form* $F_i([x_1, x_2]) = [a_i \log(x_1 + 1), b_i \log(x_2 + 1)]$*, with* $w_1 = w_2 = \frac{1}{2}$*. Suppose that* $\mathbf{x}_1^* = [s_1, 0]$ *and* $\mathbf{x}_2^* = [0, s_2]$ *for some* $s_1, s_2 > 0$*. Then using the KKT conditions, we show that* $\mathbf{x}_1^*$ *and* $\mathbf{x}_2^*$ *are only if*

$$\frac{a_1 b_2}{a_2 b_1} \geq (1 + s_1)(1 + s_2).$$

More generally, suppose developers' production functions concentrate output on different sets of dimensions, with γ_i measuring spillover outside developer i's specialized dimensions.

Lemma 3 (Specialized Production Functions)**.** *Consider two developers* i *and* i' *with disjoint sets* $S_i, S_{i'} \subseteq [K]$*. Suppose that, for each* $j \in \{i, i'\}$*, there exists* $\gamma_j \in [0, 1]$ *such that*

$$\left\| F_j(\mathbf{x}_j)_{S_j^c} \right\|_2 \leq \gamma_j \left\| F_j(\mathbf{x}_j)_{S_j} \right\|_2 \qquad \forall \mathbf{x}_j \in \mathcal{X}_j.$$

Then any equilibrium $\mathbf{x}^*$ *with* $\|\boldsymbol{\theta}_i^*\|_2, \|\boldsymbol{\theta}_{i'}^*\|_2 > 0$ *satisfies*

$$H_{i,i'}(\mathbf{x}^*) \leq \frac{\gamma_i + \gamma_{i'}}{\sqrt{1 + \gamma_i^2}\sqrt{1 + \gamma_{i'}^2}}.$$

We demonstrate two ways in which comparative advantage can induce specialization. Example 1 shows how differences in production functions lead to different competitive responses, while Lemma 3 shows that production functions concentrated on disjoint capability sets yield differentiated outputs when cross-capability spillovers are small (see Appendix F.5 for proof).

4.3 Characterizing Monoculture and Specialization Generally

While Lemma 2 provides a sufficient condition, even drastically different production functions can result in monoculture (see Appendix C for an example). Instead, the key requirement for monoculture is that developers must be able to locally support the same output direction under competitive incentives. Using the decomposition $\boldsymbol{\theta}_i^* = \alpha_i^* d_i^*$, for a profile of candidate directions $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N) \in (S_+^{K-1})^N$ and $\boldsymbol{\alpha} \in \mathbb{R}_+^N$, define

$$p_{i,k}(\mathbf{D}, \boldsymbol{\alpha}) = \frac{w_k}{N - 1} \sum_{j \neq i} \sigma'(\alpha_i d_{i,k} - \alpha_j d_{j,k}).$$

For a profile of output directions, $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N) \in (S_+^{K-1})^N$, we say that $\mathbf{D}$ is locally supportable if there exist $\boldsymbol{\alpha} \in \mathbb{R}_+^N$ and $\mathbf{x}_i \in \mathbb{R}_+^K$ such that

$$F_i(\mathbf{x}_i) = \alpha_i \mathbf{d_i} \neq 0 \forall i, \quad \text{and} \quad \nabla F_i(\mathbf{x}_i)^\top \mathbf{p}_i(\mathbf{D}, \boldsymbol{\alpha}) \leq c'(\|\mathbf{x}_i\|_1)\mathbf{1} \forall i,$$

with equality in coordinates ℓ when $x_{i,\ell} > 0$. This captures the set of feasible output directions based on equilibrium conditions. We use this to characterize monoculture and specialization.

Theorem 2 (Directional Characterization of Monoculture and Specialization)**.** *Suppose that the conditions of Theorem 1 hold. A profile of directions* $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N) \in (S_+^{K-1})^N$ *is induced by a pure-strategy Nash equilibrium with nonzero outputs if and only if* $\mathbf{D}$ *is locally supportable.*

A pure-strategy equilibrium with monoculture exists if and only if $(\mathbf{d}, \ldots, \mathbf{d})$ *is locally supportable for some* $\mathbf{d} \in S_+^{K-1}$*, and a pure-strategy equilibrium with complete specialization exists if and only if there is a locally supportable profile* $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N)$ *with* $\mathbf{d}_i^\top \mathbf{d}_{i'} = 0$ *for all* $i \neq i'$*.*

We show that developers' directional response sets characterize which output directions can arise under competition. The result follows from the first-order conditions, by noting that the equilibrium direction is locally supportable by every developer (details in Appendix F.6). Monoculture does not require similar production functions, but only that developers' production functions can support the same output direction under competitive incentives. We next tackle whether performance levels converge over time even when developers retain different relative strengths across capabilities.

5 Competitive Dynamics

Section 4 characterized when developers' equilibrium capability profiles exhibit monoculture or specialization. We now turn from the direction of capabilities at equilibrium to the evolution of performance levels over time. In particular, we ask when competing developers become closer in performance as investment accumulates, even if differences in their capability profiles persist.

Let $\mathbf{X}_{i,t} = \sum_{s=1}^{t} \mathbf{x}_{i,s}$ denote developer i's cumulative investment through time t. Additionally, let $\boldsymbol{\theta}_i^t = F_i(\mathbf{X}_i^t)$ denote developer i's capability vector at time t. At each time step t, each developer chooses $\mathbf{x}_{i,t}$ to maximize its current-period utility:

$$\frac{1}{N-1} \sum_{i' \neq i} \sum_{k} w_k \sigma(F_i(\mathbf{X}_{i,t-1} + \mathbf{x}_{i,t})_k - F_{i'}(\mathbf{X}_{i',t-1} + \mathbf{x}_{i',t})_k) - c(\|\mathbf{x}_{i,t}\|_1).$$

This abstraction isolates how repeated investment changes relative performance without introducing intertemporal strategic considerations. We ask whether developers' relative performance converges over time. We do not expect competition to guarantee convergence: developers may remain separated because their cumulative investments diverge or because differences in their production functions persist at scale. Rather than characterize the full dynamic equilibrium, we isolate conditions on investment trajectories and production functions under which relative performance must converge.

Proposition 1 (Catch-Up and Long-Run Crowding)**.** *Let* $\boldsymbol{\theta}_i^t = F_i(\mathbf{X}_i^t)$ *denote developer* i*'s capability vector at time* t*, and define its category-specific gap to the frontier as* $g_{i,k}^t = \max_j \theta_{j,k}^t - \theta_{i,k}^t$*. Let* $\mathbf{g}^t \in \mathbb{R}_+^{NK}$ *stack these gaps across developers and categories at timestep* t*, and let*

$$\Delta\boldsymbol{\theta}^t = \left(\boldsymbol{\theta}_1^{t+1} - \boldsymbol{\theta}_1^t, \ldots, \boldsymbol{\theta}_N^{t+1} - \boldsymbol{\theta}_N^t\right) \in \mathbb{R}^{NK}.$$

Suppose each F_i *is continuously differentiable with*

$$\max_{\ell \in [K]} \left\| \frac{\partial F_i(\mathbf{x})}{\partial x_\ell} \right\|_2 \to 0 \qquad \text{as } \|\mathbf{x}\|_1 \to \infty, \quad \forall i.$$

Suppose there exists $\beta > 0$ *such that, for sufficiently large* t*,* $\langle \mathbf{g}^t, \Delta\boldsymbol{\theta}^t \rangle \geq \beta \|\mathbf{g}^t\|_2^2$*. Then we have that* $\|\mathbf{g}^t\|_2 \to 0$*, and* $\max_{i'} \theta_{i',k}^t - \theta_{i,k}^t \to 0$ *for all developers* i *and categories* k*.*

The competitive dynamics generate a sequence of relative performance gaps and per-period investments that decrease over time, while the catch-up condition requires improvements to be directed toward those gaps. Persistent catch-up allows lagging developers to catch-up while vanishing marginal returns prevent developers that are ahead from taking too far a lead (proof in Appendix F.7). In Appendix F.8, we demonstrate an illustrative sufficient condition for pairwise catch-up.

Sections 4 and 5 characterize two dimensions of convergence. Developers may become similar in their capability profiles (monoculture), or in their overall performance levels (crowding). These need not coincide because developers can approach a common performance frontier while retaining capability-specific strengths, allowing competitive catch-up and differentiation to coexist.

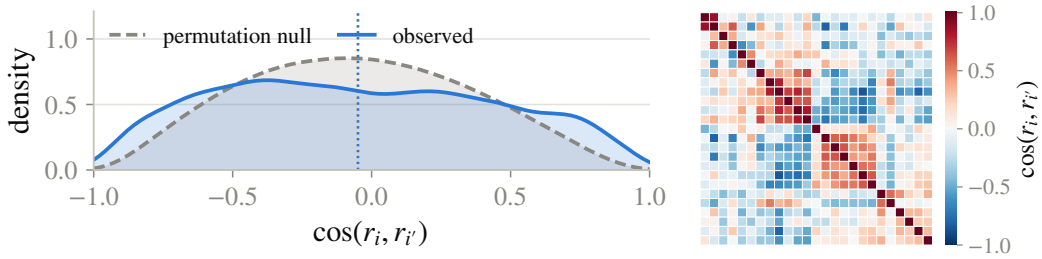

Figure 2: Developers form clusters with similar relative capability profiles, but do not all look alike. Left: distribution of pairwise capability-profile similarities across developers, compared with a permutation null that randomly reassigns capabilities across developers. Right: pairwise capability-profile similarities reveal clusters of developers with similar capability profiles.

6 Leaderboard Competition in Practice

Our empirics examine observable counterparts of these theoretical objects. Our first question studies capability-profile differentiation, corresponding to the output directions characterized in Section 4. Our second question measures whether improvements are directed toward performance gaps, corresponding to the catch-up condition in Proposition 1. Our third question measures whether performance gaps narrow over time, corresponding to the crowding conclusion of Proposition 1. Because we do not observe developers' production functions, we cannot directly test the diminishing-returns condition in Proposition 1. These analyses do not directly test the assumptions of our theoretical model, but ask whether the patterns generated by its mechanisms appear in observed model competition. We rely on two sources of data: leaderboards from LM Arena and usage data from OpenRouter. From LM Arena, we obtain 221 leaderboard snapshots and focus on 8 categories, while from OpenRouter, we use daily traffic data where we observe the top-50 models used each day (details for both in Appendix D.1).

6.1 RQ1: Do model developers exhibit similar capability profiles?

Section 4 motivates examining the direction of developers' capability profiles. While we cannot observe developers' production functions, we can investigate whether capability profiles exhibit monoculture or differentiation. However, directly applying the metrics from Section 4 to Elo scores is inappropriate because Elo scores are identified only up to an additive shift. As a result, we instead construct a shift-invariant empirical analogue by fitting, at each time step, $\theta_{i,k} = \delta + \eta_i + \gamma_k + r_{i,k}$ for each time step separately. We do this to account for both category difficulty (e.g., math being easier than English) and dominant developers (e.g., Anthropic being broadly better), so we can understand relative strengths across capabilities. In Figure 2 (left), we plot the distribution of $\cos(r_{i,\cdot}, r_{i',\cdot})$ between pairs of developers. We find little evidence of a common capability profile across all developers; the observed distribution is close to a null that reassigns capabilities across developers (see Appendix D.4 for details and statistical tests).

Despite this lack of global structure, Figure 2 (right) plots pairwise similarity between developers and reveals clusters of models that share similar residuals. For example, models from Google and Meta are stronger in creative writing and multi-turn interactions and weaker in coding and math. In Appendix E.1, we explore how developer similarities evolve over time, and find that some relative strengths persist while many fluctuate. Together, these results suggest that developers retain some persistent differences in their relative strengths, even as some groups exhibit similar profiles.

6.2 RQ2: Do developers catch up in capabilities where they lag competitors?

Proposition 1 shows that persistent catch-up, combined with vanishing marginal returns, is sufficient for long-run crowding. We investigate the empirical analog for persistent catch-up by asking whether developers catch up in capabilities where they lag competitors. We relate a developer's improvement during new model releases, $\Delta\theta_{i,k,t}$, to its pre-release gap relative to the best rival $g_{i,k,t} = \theta_{\text{leader},k,t} - \theta_{i,k,t}$. We fit $\Delta\theta_{i,k,t} = \eta_{i,k} + \gamma_{k,t} + \beta g_{i,k,t} + \varepsilon_{i,k,t}$, where t is the calendar

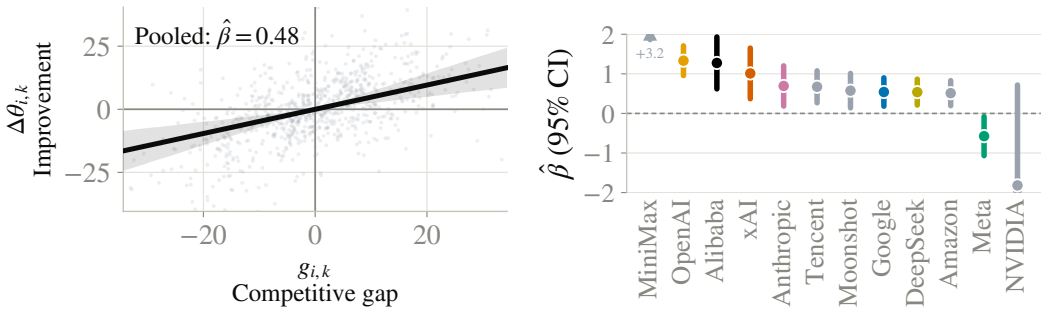

Figure 3: Developers catch up more in categories where they lag further behind the leader. Left: relationship between competitive gaps and subsequent improvements, where $\hat{\beta}$ denotes the slope. Right: developer-specific estimates of the catch-up coefficient, with 95% confidence intervals.

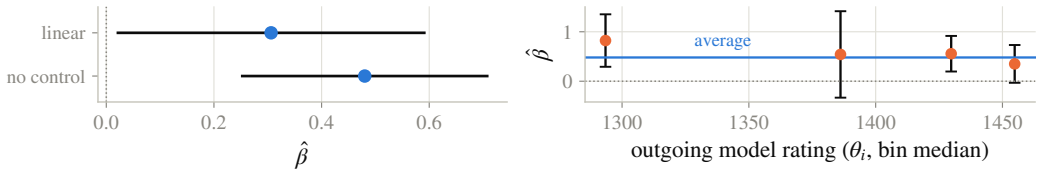

Figure 4: The catch-up relationship remains positive after controlling for overall model strength. Left: estimates of $\hat{\beta}$ after adding a linear control for overall model strength, $\theta_{i,\cdot}$. Right: estimates of $\hat{\beta}$ across quartiles of overall model strength.

quarter of the release, and η and γ represent fixed effects (details in Appendix D.5). We estimate $\hat{\beta}$, which captures whether larger competitive gaps predict subsequent improvement. This regression is the empirical analog of the catch-up condition from Proposition 1. In Appendix D.3, we detail how we compute $\hat{\beta}$, including noise corrections.

Figure 3 shows a consistent competitive catch-up pattern: developers improve more in categories where they previously lagged the frontier. For 10 of the 12 developers, $\hat{\beta} > 0$ and the 95% confidence interval excludes zero. One potential explanation is that weaker models may have greater room for improvement, so we control for the overall rating of the outgoing model $\theta_{i,\cdot}$, and re-estimate $\hat{\beta}$. Figure 4 accounts for baseline performance in two ways: by directly controlling for $\theta_{i,\cdot}$ in the regression (left), and by estimating $\hat{\beta}$ separately across quartiles of $\theta_{i,\cdot}$ (right). In both cases, $\hat{\beta}$ remains positive, though the catch-up relationship is largest among models with lower baseline performance. In Appendix E.2, we account for other confounds, and demonstrate that the relationship remains positive after accounting for regression to the mean, release year, and release-event heterogeneity. Therefore, developers robustly improve more where they previously lag, although our observational analysis does not establish that competitive pressure causes this relationship.

6.3 RQ3: Is the evaluated model frontier becoming more crowded?

We finally examine the crowding pattern characterized by Proposition 1 by looking at whether the evaluation frontier crowds over time. We do so by measuring the performance gap between the leader and lower-ranked models, which is the empirical analog of $\mathbf{g}^t$, averaged across categories. Figure 5 shows increasing crowding; between July 2024 and September 2026, the absolute gaps between the leader and the 10th-, 20th-, and 50th-ranked models decrease by 53%, 61%, and 66%, respectively. This pattern is not driven solely by developers releasing multiple models: the number of distinct developers within 100 and 200 points of the leader increases by 2.4× and 1.4×, respectively. We further investigate this in Figure 6 (left), where we show that models in 2026 lie closer to the frontier than models from 2025.

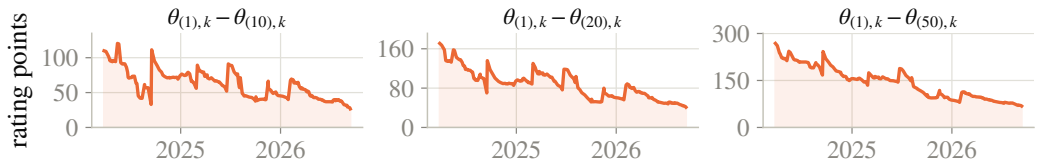

Figure 5: The evaluation frontier becomes increasingly crowded over time. Performance gap between the leader and the 10th-, 20th-, and 50th-ranked models, averaged across categories.

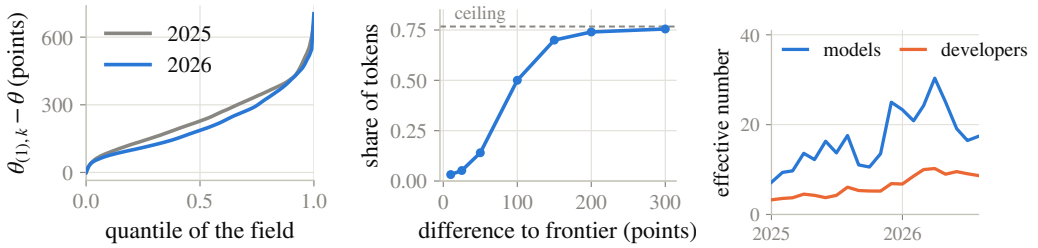

Figure 6: Models become closer in performance over time, alongside increasingly dispersed usage. Left: models in 2026 lie closer to the performance frontier than in earlier years. Middle: most models used on OpenRouter are within 200 points of the top model. Right: the effective number of models and developers used on OpenRouter has increased over time.

We relate frontier crowding to model usage by matching models between Arena and OpenRouter, covering 76% of OpenRouter token volume (see Appendix D.2). We note two caveats: 1) OpenRouter is not necessarily representative of all developers, and 2) OpenRouter can sometimes automatically select the model which is used based on user preferences. Nonetheless, we use OpenRouter as a way to understand how usage varies in a segment of the population. Figure 6 shows that OpenRouter usage is concentrated among near-frontier models; models within 150 points of the Arena leader account for 70% of observed tokens. At the same time, OpenRouter usage is becoming dispersed across the models that reach this frontier. Between January 2025 and August 2026, the effective number of models and developers–defined as the inverse Herfindahl index of token shares–used on OpenRouter increases by 2.4× and 2.7×, respectively. Taken together, these results mirror the distinction highlighted by our theory; convergence in performance levels need not imply convergence in capability profiles. Developers retain some persistent relative strengths across capabilities even as they catch up where they lag and increasingly crowd near the performance frontier.

7 Discussion

What role does evaluation play? Public evaluations provide a common signal of which capabilities are rewarded and how developers compare, giving competitors a shared target toward which to optimize. Absent such a common signal, developers' incentives may be more dispersed, potentially encouraging differentiation or specialization (Raghavan, 2024).

What role do evaluators play? A natural question is the role evaluators play in shaping competitive dynamics. In our model, evaluators modify $\mathbf{w}$, which impacts developer incentives. Evaluators might be able to influence where competitive pressure is concentrated. For example, increasing the weight on a capability may increase developers' incentives to improve along that dimension. Evaluation design is therefore not only a measurement decision, but can shape the direction of competitive investment.

Limitations Our theoretical model abstracts features of real-world competition. We assume developers optimize expected win rate rather than rank, which simplifies the strategic environment and affects equilibrium existence. In Appendix A we show that this objective directly captures weighted

category-level pairwise win probabilities under Bradley–Terry/Elo-style evaluation. Our theoretical results characterize mechanisms through which evaluation can shape competitive investment, while our empirical results test whether patterns consistent with these mechanisms appear in practice; they do not establish that evaluation causes the observed patterns of convergence. Causal attribution is particularly difficult in this setting because competition and model development evolve jointly.

8 Related Work

Leaderboards Leaderboards crowdsource preferences and provide granular information about preference (Chiang et al., 2024; Frick et al., 2025). Prior work studies how to aggregate heterogeneous preferences into representative rankings (Rodriguez et al., 2021; Haghtalab et al., 2026; Lanctot et al., 2023) and make these rankings robust to strategic manipulation (Min et al., 2025; Singh et al., 2025; Hays et al., 2026; Huang et al., 2026). Closest to our theoretical setting is Chen et al. (2026), which studies strategic competition over leaderboard rank. We instead study how competition shapes developers' investment across multiple capabilities, and whether models specialize.

Contest Theory Contest theory studies how strategic agents expend costly effort to improve their probability of success (Borel, 1953; Friedman, 1958; Skaperdas, 1996). Our model is closely related to multi-battle contest models (Tullock, 2001; Konrad & Kovenock, 2009): developers allocate costly investment across categories, with pairwise success in each category determined by the function σ. We differ in studying when multidimensional competition leads developers to produce similar or specialized capability profiles.

Strategic Machine Learning In many real-world settings, agents respond strategically to machine learning algorithms, as deploying prediction algorithms might either impact agent responses for future predictions themselves (Hardt et al., 2016; Kleinberg & Raghavan, 2020; Perdomo et al., 2020; Liu et al., 2022). These strategic responses can also impose social costs, for example by inducing agents to expend additional effort simply to maintain favorable classifications (Milli et al., 2019). Our setting reverses the usual direction of strategic response: rather than individuals adapting to a deployed model, model developers adapt their investments to an evaluation mechanism.

Competition and Monoculture Finally, our work connects to a growing literature on algorithmic monoculture and the effects of competition on diversity. Algorithmic monoculture describes settings in which multiple decision makers rely on similar algorithms, rankings, or predictions, causing their decisions to become correlated (Kim et al., 2025; Jo et al., 2026; Bommasani et al., 2022). Prior work has shown that competition can act as an anti-coordination force, encouraging developers to differentiate (Raghavan, 2024). We identify a countervailing force; when competitors are evaluated against a common benchmark, evaluation can coordinate investment toward the same capabilities. Related work similarly shows that competition among increasingly capable models need not increase diversity and can, in some settings, reduce welfare (Jagadeesan et al., 2023). Closest to our empirical setting is Donahue & Raghavan (2026), which studies whether model aggregation through services such as OpenRouter produces homogenization. While both papers study diversity in model ecosystems, their mechanism is aggregation across existing models; ours is endogenous investment, asking how common evaluation shapes competing developers' investments and capability profiles.

9 Conclusion

Leaderboards not only measure model capabilities, but can also shape how developers allocate resources. Our model shows that this competition can lead to monoculture or specialization depending on the underlying production functions. Similar production functions are sufficient for monoculture, while comparative advantage can induce specialization. In practice, these forces coexist: developers improve more where they lag while retaining persistent category-specific differences, alongside an increasingly crowded model frontier. Our results highlight a mechanism through which shared evaluation can shape not only which models succeed, but also which capabilities developers choose to pursue. For evaluation designers, this means that choosing which capabilities to evaluate and how heavily to weight them is not only a measurement decision, but can also shape investment.

LLM DISCLOSURE

We used LLMs to assist us with the generation and proving of theorems, along with the writing of code to test experiments. All proofs were checked manually by the authors, and writing in the paper was originally done by the authors. LLM-generated code was manually checked by authors. We additionally used LLMs to assist with proofreading and editing this paper. LLMs assisted with finding additional related work after the authors of the papers did their own related work search first. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The attached supplementary material contains code necessary to reproduce the experiments. In addition, Appendix D contains experimental details, and Appendix F contains details on proofs.

REFERENCES

Mubashara Akhtar, Anka Reuel, Prajna Soni, Sanchit Ahuja, Pawan Sasanka Ammanamanchi, Ruchit Rawal, Vilém Zouhar, Srishti Yadav, Chenxi Whitehouse, Dayeon Ki, et al. When ai benchmarks plateau: A systematic study of benchmark saturation. *arXiv preprint arXiv:2602.16763*, 2026.

Rishi Bommasani, Kathleen A Creel, Ananya Kumar, Dan Jurafsky, and Percy S Liang. Picking on the same person: Does algorithmic monoculture lead to outcome homogenization? *Advances in neural information processing systems*, 35:3663–3678, 2022.

Emile Borel. The theory of play and integral equations with skew symmetric kernels. *Econometrica: journal of the Econometric Society*, pp. 97–100, 1953.

Alex Chan. Benchmark mineability and the financing of ai innovation. Technical report, National Bureau of Economic Research, 2026.

Yatong Chen, Guanhua Zhang, and Moritz Hardt. Leaderboard incentives: Model rankings under strategic post-training. *arXiv preprint arXiv:2603.08371*, 2026.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

Ricardo Dominguez-Olmedo, Florian Eddie Dorner, and Moritz Hardt. Training on the test task confounds evaluation and emergence. In *International Conference on Learning Representations*, volume 2025, pp. 68564–68594, 2025.

Kate Donahue and Manish Raghavan. Impacts of aggregation on model diversity and consumer utility. *arXiv preprint arXiv:2602.23293*, 2026.

Evan Frick, Connor Chen, Joseph Tennyson, Tianle Li, Wei-Lin Chiang, Anastasios N Angelopoulos, and Ion Stoica. Prompt-to-leaderboard. *arXiv preprint arXiv:2502.14855*, 2025.

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

Nika Haghtalab, Ariel D Procaccia, Han Shao, Serena Lutong Wang, and Kunhe Yang. Pluralistic leaderboards. *arXiv preprint arXiv:2606.02547*, 2026.

Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, pp. 111–122, 2016.

Chris Hays, Rachel Li, Bailey Flanigan, and Manish Raghavan. Strategic candidacy in generative ai arenas. *arXiv preprint arXiv:2603.26891*, 2026.

Jenny Huang, Yunyi Shen, Dennis Wei, and Tamara Broderick. Dropping just a handful of preferences can change top large language model rankings. In *International Conference on Learning Representations*, volume 2026, pp. 90685–90721, 2026.

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

Nathanael Jo, Nikhil Garg, and Manish Raghavan. The subjectivity of monoculture. *arXiv preprint arXiv:2602.24086*, 2026.

Elliot Kim, Avi Garg, Kenny Peng, and Nikhil Garg. Correlated errors in large language models. *arXiv preprint arXiv:2506.07962*, 2025.

Jon Kleinberg and Manish Raghavan. How do classifiers induce agents to invest effort strategically? *ACM Transactions on Economics and Computation (TEAC)*, 8(4):1–23, 2020.

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

Kai A Konrad and Dan Kovenock. Multi-battle contests. *Games and Economic Behavior*, 66(1): 256–274, 2009.

Marc Lanctot, Kate Larson, Yoram Bachrach, Luke Marris, Zun Li, Avishkar Bhoopchand, Thomas Anthony, Brian Tanner, and Anna Koop. Evaluating agents using social choice theory. *arXiv preprint arXiv:2312.03121*, 2023.

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

Smitha Milli, John Miller, Anca D Dragan, and Moritz Hardt. The social cost of strategic classification. In *Proceedings of the conference on fairness, accountability, and transparency*, pp. 230–239, 2019.

Rui Min, Tianyu Pang, Chao Du, Qian Liu, Minhao Cheng, and Min Lin. Improving your model ranking on chatbot arena by vote rigging. *arXiv preprint arXiv:2501.17858*, 2025.

Kenny Peng and Nikhil Garg. Monoculture in matching markets. *Advances in Neural Information Processing Systems*, 37:81959–81991, 2024.

Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *International conference on machine learning*, pp. 7599–7609. PMLR, 2020.

Manish Raghavan. Competition and diversity in generative ai. *arXiv preprint arXiv:2412.08610*, 2024.

Brian Roberson. The colonel blotto game. *Economic Theory*, 29(1):1–24, 2006.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P Lalor, Robin Jia, and Jordan Lee Boyd-Graber. Evaluation examples are not equally informative: How should that change nlp leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4486–4503, 2021.

Shivalika Singh, Yiyang Nan, Alex Wang, Daniel D'souza, Sayash Kapoor, Ahmet Üstün, Sanmi Koyejo, Yuntian Deng, Shayne Longpre, Noah A Smith, et al. The leaderboard illusion. *arXiv preprint arXiv:2504.20879*, 2025.

Stergios Skaperdas. Contest success functions. *Economic theory*, 7(2):283–290, 1996.

Gordon Tullock. Efficient rent seeking. In *Efficient rent-seeking: Chronicle of an intellectual quagmire*, pp. 3–16. Springer, 2001.

Milan Vojnović. *Contest theory: Incentive mechanisms and ranking methods*. Cambridge University Press, 2015.

A Win Rate and Elo

We discuss the relationship between the win-rate objective in our model and Bradley–Terry/Elo-style rating systems. Recall that developer i's objective depends on its pairwise probability of beating other developers across categories. In category k, the probability that developer i beats developer i' is $\sigma\left(F_i(\mathbf{x}_i)_k - F_{i'}(\mathbf{x}_{i'})_k\right)$. When σ is logistic, this corresponds exactly to a Bradley–Terry model within each category, where $F_i(\mathbf{x}_i)_k$ is the latent performance of developer i in category k. Elo uses the same logistic relationship between differences in latent scores and pairwise win probabilities, up to a rescaling of the scores.

Our model aggregates these pairwise win probabilities across categories. For category weights $\mathbf{w}$, the overall win probability of developer i against developer i' is

$$\sum_{k=1}^{K} w_k \sigma\left(F_i(\mathbf{x}_i)_k - F_{i'}(\mathbf{x}_{i'})_k\right).$$

Thus, the objective in our model corresponds to developers maximizing a linear function of their category-level win rates. This directly captures settings in which models are evaluated according to overall win rate, or according to a weighted aggregation of category-level performance.

This is distinct from fitting a single latent Bradley–Terry rating to each model using comparisons pooled across categories. In general,

$$\sum_{k=1}^{K} w_k \sigma(z_k) \neq \sigma\left(\sum_{k=1}^{K} w_k z_k\right),$$

since a weighted average of logistic functions is not itself generally logistic unless z_k is constant. Thus, when production and performance are inherently category-specific, there need not exist a single Bradley–Terry score whose implied pairwise probabilities coincide with the weighted win probabilities in our model. A leaderboard based on a single Bradley–Terry/Elo score therefore imposes an additional one-dimensional representation of model performance. Our model instead directly represents category-level win probabilities and developers' incentives to improve their weighted aggregate win rate.

B Situations Where an Equilibrium Doesn't Exist

We note that an equilibrium is not guaranteed to exist if σ is close to the step function. For example, suppose that

$$f_i(x_i) = x_i, \qquad c(x_i) = \alpha x_i, \qquad \sigma_\tau(z) = \frac{1}{1+e^{-\tau z}}, \qquad \tau > 4\alpha.$$

Here, $u_i(x_i; x_j) = \sigma_\tau(x_i - x_j) - \alpha x_i$, where τ controls how sharply the evaluator distinguishes between developers; as $\tau \to \infty$, σ_τ approaches a step function.

We first show that no symmetric candidate equilibrium, $x_1 = x_2 = x$ exist when $\tau > 4\alpha$. At such a profile,

$$\frac{\partial u_i}{\partial x_i} = \sigma_\tau'(0) - \alpha = \frac{\tau}{4} - \alpha > 0.$$

Thus, holding the opposing developer's investment fixed, each developer can increase its utility by slightly increasing its own investment. We note that investments above $\bar{x}$ cannot be optimal, and so no symmetric profile can be a Nash equilibrium. This is because, beyond $\bar{x}$, each developer achieves negative utility, and so would prefer to deviate.

Next, we demonstrate that an asymmetric equilibrium cannot exist either. Suppose without loss of generality that $x_1 > x_2$. When a developer is behind, we have $\sigma_\tau''(x_2 - x_1) > 0$, so its utility is strictly convex in its own investment. As a result, the losing developer cannot have an interior optimum strictly below its competitor. Therefore, any asymmetric equilibrium must, up to relabeling, take the form $(x, 0)$ for some $x > 0$. Since x is optimal for the winning developer against zero, it satisfies

$$u_1'(x; 0) = \sigma_\tau'(x) - \alpha = 0.$$

Moreover, since $\tau > 4\alpha$, we have $u_1'(0;0) = \tau/4 - \alpha > 0$. Since $\sigma_\tau'(y)$ is strictly decreasing for $y > 0$, it follows that $u_1'(y;0) > 0$ for every $y \in [0, x)$. Hence

$$u_1(x;0) = \sigma_\tau(x) - \alpha x > u_1(0;0) = \frac{1}{2}.$$

We now show that such a solution cannot be an equilibrium. If developer 2 deviates from 0 to $2x$, its gain is

$$u_2(2x;x) - u_2(0;x) = \sigma_\tau(x) - 2\alpha x - \sigma_\tau(-x) = 2\left(\sigma_\tau(x) - \alpha x - \frac{1}{2}\right) > 0,$$

where the final inequality follows from $u_1(x;0) > 1/2$. Thus, developer 2 strictly prefers to leapfrog developer 1, so $(x, 0)$ cannot be an equilibrium. Therefore, no pure-strategy Nash equilibrium exists. The losing developer strictly prefers to leapfrog the winner, and so $(x, 0)$ cannot be an equilibrium. As a result, no pure-strategy Nash equilibrium exists.

C Monoculture from Different Production Functions

We present an example with $N = 2$ developers, where the two developers have very different production functions, yet they achieve near-identical outputs.

Example 2. *Consider a scenario with $N = K = 2$, $\sigma(z)$ as the sigmoid function, and $c(\|\mathbf{x}_i\|_1) = \frac{0.15}{2}\|\mathbf{x}_i\|_1^2$. Let $F_i(\mathbf{x}_i) = A_i \log(\mathbf{x}_i + \mathbf{1})$, where the logarithm is applied componentwise. Let*

$$A_1 = \begin{pmatrix} 1.0 & 0.3 \\ 0.2 & 0.9 \end{pmatrix}, \qquad A_2 = \begin{pmatrix} 0.4 & 0.8 \\ 0.7 & 0.1 \end{pmatrix}, \qquad \mathbf{w} = \begin{pmatrix} 0.35 & 0.65 \end{pmatrix}.$$

Then numerically solving for the equilibrium yields

$$x_1^* \approx (0.115,\ 0.601), \qquad x_2^* \approx (0.613,\ 0),$$

and the corresponding outputs are

$$\boldsymbol{\theta}_1^* \approx (0.250,\ 0.445), \qquad \boldsymbol{\theta}_2^* \approx (0.191,\ 0.335).$$

Here, $H_{1,2}(\mathbf{x}^*) = 0.99997$, and so the outputs are near-parallel while the production functions are very different.

D Experimental Details

D.1 Dataset Details

Leaderboards Our primary longitudinal data come from LM Arena. We construct a panel covering 422 models from 59 organizations and 28 categories, using the "lmarena-ai/leaderboard-dataset". Of these 28 categories, we focus on 8 (coding, math, hard prompts, instruction following, creative writing, longer query, multi-turn, and english) because these categories are consistently available since 2024. This data comes from March 2024 through September 2026, and additionally contains an overall rating. The snapshots contain category-level Elo ratings estimated from pairwise votes using a Bradley–Terry model, along with rating variances and vote counts. We use this panel to measure developers' capability profiles, changes across model releases, and whether subsequent improvements are associated with dimensions in which developers previously lagged their competitors.

OpenRouter To measure realized model usage, we use daily OpenRouter traffic data from January 2025 through September 2026. The data cover 408 models over 620 days. We aggregate token shares to the model and developer levels to study how changes in the evaluated model frontier correspond to changes in realized model usage.

D.2 Name Matching Details

For our usage analysis, we match model names between OpenRouter and LM Arena, which use different naming conventions for provider prefixes, routing variants, release dates, and model versions.

We first canonicalize names by removing provider and routing information and matching models based on their alphabetic and numeric components. We then allow matches that differ only in date stamps, and manually verify remaining models with material token volume. We deliberately avoid general fuzzy matching, since incorrect matches would attribute token volume to the wrong capability profile. Overall, we match 213 of 408 OpenRouter models, covering 71.3% of pooled token volume and an average of 76.7% of monthly token volume. Models without a defensible LMArena counterpart, including pre-release codenames and models not separately evaluated by LMArena, are left unmatched. We treat unmatched token volume as falling outside any threshold of the top-rated model, making our estimates of usage near the frontier conservative.

D.3 Noise Correction

The leaderboard reports an Elo rating $\hat{\theta}_{ikt} = \theta_{ikt} + \varepsilon_{ikt}$, together with an estimated sampling variance $v_{ikt} = \mathrm{Var}(\varepsilon_{ikt})$. Because the outgoing model's rating enters both sides of our regression, sampling noise mechanically induces a positive relationship between the measured gap and subsequent improvement. In particular,

$$\Delta\hat{\theta}_{ik} = \hat{\theta}^{\mathrm{new}}_{ik} - \hat{\theta}^{\mathrm{old}}_{ik}, \qquad \hat{g}_{ik} = \hat{\theta}^{\mathrm{riv}}_{ik} - \hat{\theta}^{\mathrm{old}}_{ik},$$

so $\varepsilon^{\mathrm{old}}_{ik}$ enters both variables with a negative sign. Assuming mean-zero rating errors that are independent across models and releases conditional on the true ratings,

$$\mathrm{Cov}\left(\Delta\hat{\theta}_{ik}, \hat{g}_{ik}\right) = v^{\mathrm{old}}_{ik}, \qquad \mathrm{Var}\left(\hat{g}_{ik} - g_{ik}\right) = v^{\mathrm{old}}_{ik} + v^{\mathrm{riv}}_{ik}.$$

Sampling noise can bias toward a positive catch-up effect.

We remove this bias using an errors-in-variables correction. Let Z denote the design matrix for the developer-category and category-quarter fixed effects, and let

$$M = I - Z(Z^\top Z)^{-1}Z^\top$$

denote the corresponding residual-maker. Thus, multiplying by M removes the variation explained by these fixed effects.

After computing residuals, the regression coefficient is the covariance between the measured competitive gap and subsequent improvement, divided by the variance of the measured gap. Rating noise contributes to both quantities. We therefore subtract its expected contribution before computing the coefficient:

$$\hat{\beta}_c = \frac{\hat{g}^\top M\Delta\hat{\theta} - d_{\mathrm{fac}}\, n\bar{v}^{\mathrm{old}}}{\hat{g}^\top M\hat{g} - d_{\mathrm{fac}}\, n\left(\bar{v}^{\mathrm{old}} + \bar{v}^{\mathrm{riv}}\right)},$$

where $\bar{v}^{\mathrm{old}}$ and $\bar{v}^{\mathrm{riv}}$ are the panel-average reported sampling variances for the outgoing model and its rival, respectively, and d_{fac} accounts for the degrees of freedom consumed by the within transformation. The correction has little effect on our estimate: the baseline estimate is $\hat{\beta} = 0.487$, compared with $\hat{\beta}_c = 0.480$. This limited change reflects that variation in the measured competitive gaps is substantially larger than the reported sampling noise.

D.4 Statistical Comparisons for RQ1

In Figure 2 (left), we compare this distribution to a permutation null that independently permutes developers within each category, preserving each category's marginal distribution of residuals while destroying any developer-specific pattern of relative strength. We find little difference between the observed and null distributions: the null has a mean cosine similarity of -0.051, compared with -0.048 in the observed data. While this difference is statistically significant ($p < 0.001$, paired t-test), it is negligible in magnitude.

In Figure 2 (right) we average the similarities between pairs of developers across snapshots. Four groupings are particularly visible in the similarity matrix, which we summarize here (each contains additional developers beyond those highlighted): models from IBM and Microsoft are stronger in math, and relatively weaker in multi-turn interactions and creative writing; models from NVIDIA and Amazon are stronger in English and coding, and weaker in creative writing; models from Google and Meta are stronger in creative writing and multi-turn interactions, and weaker in coding and math; and models from OpenAI and Anthropic are stronger in instruction following, longer queries, and weaker in English.

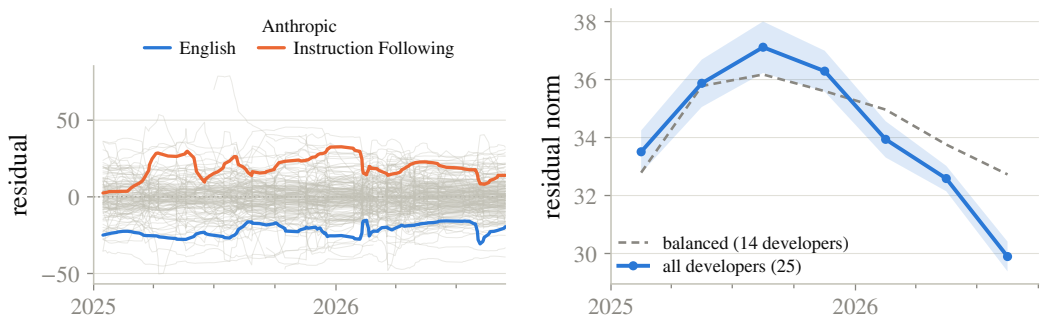

Figure 7: Specialization of developers over time. Left: most developer-category residuals are noisy, but a few persist across time. For example, Anthropic's score in English and its score in instruction following hold in sign across time steps. Right: magnitude of residuals over time.

D.5 Statistical Comparisons for RQ2

We define a flagship changeover as any change in the identity of a developer's model with the highest overall rating. We correct all estimates for measurement error using Arena's reported rating variances, removing the mechanical covariance between the competitive gap and subsequent improvement induced by their shared baseline rating (details in Appendix D.3). We restrict this analysis to developers with at least four qualifying flagship changeovers, each with mature ratings (at least 56 days post-release) for the incoming model. This leaves 12 developers in our analysis. Here, $g_{i,k,t}$ is computed as the difference between a model and the best competitor in that category among the 12 developers, at the time that the new model was released (the last snapshot before changing over).

E Additional Experiments

E.1 Persistence of Relative Strengths

We next ask whether these relative strengths persist over time. Figure 7 (left) shows that many developer–category residuals fluctuate around zero, but some exhibit stable advantages or disadvantages. For example, Anthropic persistently underperforms in English, while Cohere overperforms in creative writing. Of the 104 developer–category pairs in the panel, 29 have residuals of the same sign in every quarter; all 29 are statistically significant after applying the Benjamini–Hochberg procedure at a 5% false-discovery rate. These include results such as Cohere's poor performance on math, Anthropic's poor performance in English, and Cohere's good performance in creative writing.

We additionally ask whether developers have become more or less specialized over time. We measure specialization using the L2 norm of each developer's residual vector, $\|r_{i,\cdot,t}\|_2$, at each time step t, which captures the magnitude of category-specific specialization. Whereas pairwise residual cosine similarity measures whether developers specialize in similar directions, the residual norm measures the magnitude of a developer's category-specific specialization.

Figure 7 (right) shows that this magnitude decreases over time, indicating that developers' capability profiles become less specialized over time. We find a similar pattern for both the balanced set of 14 developers that appear in every quarter and the full set of developers, including those entering in 2025 and 2026, reducing concerns that the trend is driven by changes in developer composition.

E.2 Confounds in $\hat{\beta}$

We test whether the competitive catch-up relationship is robust to alternative comparator developers, release years, and sample splits. In Table 1, we first recompute $\hat{\beta}$ using various comparators; in particular we use the best rival, second-best rival, and top-3 average to define $g_{i,k}$. The resulting estimates are similar, ranging from 0.480 to 0.521. We find a similar pattern when splitting by release year: both 2025 and 2026 have $\hat{\beta} > 0.45$, showing that the positive relationship is present in both

Table 1: We account for confounds when computing $\hat{\beta}$, and find that $\hat{\beta}$ remains positive throughout.

		$\hat{\beta}$ (s.e.)	Events	N
Baseline	all events	+0.480*** (0.117)	106	848
Comparator choice	best rival	+0.480*** (0.117)	106	848
	second-best rival	+0.485*** (0.120)	106	848
	top-3 average	+0.521*** (0.117)	106	848
Release year	2025	+0.455*** (0.114)	60	480
	2026	+0.849*** (0.096)	41	328
Sample composition	events A	+0.500*** (0.155)	53	424
	events B	+0.331*** (0.105)	53	424
Event-level heterogeneity	release-event fixed effects	+0.949*** (0.115)	106	848
Regression to the mean	measured spurious slope	+0.099 (0.038)	20 splits	116,257
	correction-implied slope	+0.139	–	–

periods. Here, the remaining events are in 2024, where there were only 5 events. We additionally test whether the result is driven by a subset of release events by splitting events into two alternating subsamples and computing $\hat{\beta}$ separately on each. We find that $\hat{\beta} > 0.33$ in both halves of the split. We additionally include a release-event fixed effect δ_e, which absorbs event-specific factors shared across categories, such as differences in the overall magnitude of improvement across releases. The resulting estimate remains positive and increases to $\hat{\beta} = 0.949$. The catch-up relationship persists when identification comes from within-release variation across categories rather than comparisons across release events.

We finally consider mechanical regression to the mean, since noise in the outgoing model's rating enters both the measured gap and subsequent improvement. Because the historical leaderboard provides only rating snapshots, we cannot re-estimate $\hat{\beta}$ using independent ratings for each release event; instead, we use raw Arena battles to estimate the magnitude of this mechanical effect. We retrieve 116,257 battles, randomly split them in half, and fit Bradley–Terry ratings separately on each half to obtain two independent estimates, $\hat{\theta}^A$ and $\hat{\theta}^B$, of each model's rating. We regress $\hat{\theta}^B - \hat{\theta}^A$ on the gap computed from half A alone. Because the two estimates measure the same underlying model performance, any systematic positive relationship should arise from sampling variation in $\hat{\theta}^A$, since the two estimates are constructed from independent random halves of the same set of battles. These battles cover only April–July 2025 and three of our eight categories (plus the overall category), so we use this exercise to estimate the magnitude of the bias rather than to re-estimate $\hat{\beta}$. Averaged over 20 random splits, the spurious slope is 0.099, and the errors-in-variables correction predicts 0.139 in the same exercise. As a result, measurement error appears too small to account for the observed competitive catch-up relationship. Therefore, persistent comparative advantage coexists with a systematic pattern associated with convergence; developers retain different strengths, but disproportionately improve the dimensions on which they lag.

F Proofs of Equilibria and Monoculture Results

F.1 Common Cost Function

Proposition 2 (Common Cost Functions)**.** *Suppose each developer i has continuous production function F_i and cost function c_i, where each $c_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a strictly increasing bijection with $c_i(0) = 0$. Let c be any other such cost function. Then, for each developer i, there exists a production function $\tilde{F}_i$ and a bijection between action vectors $\mathbf{x}_i \leftrightarrow \mathbf{z}_i$ such that*

$$c_i\big(\|\mathbf{x}_i\|_1\big) = c\big(\|\mathbf{z}_i\|_1\big) \qquad \textit{and} \qquad F_i(\mathbf{x}_i) = \tilde{F}_i(\mathbf{z}_i).$$

Consequently, the game defined by $(F_i, c_i)_{i=1}^N$ is equivalent under reparameterization to the game defined by $(\tilde{F}_i, c)_{i=1}^N$: corresponding strategy profiles induce identical payoffs, and their Nash equilibria are in bijective correspondence.

Proof. Define the scalar map

$$g_i(T) := c_i^{-1}\big(c(T)\big), \qquad T \geq 0.$$

Since c_i and c are strictly increasing bijections of $\mathbb{R}_{\geq 0}$, so is g_i. In particular, g_i^{-1} exists and is itself a strictly increasing bijection of $\mathbb{R}_{\geq 0}$. By construction,

$$c_i\big(g_i(T)\big) = c(T) \qquad \text{for all } T \geq 0. \tag{1}$$

Equivalently, substituting $g_i^{-1}(T)$ for T gives

$$c\big(g_i^{-1}(T)\big) = c_i(T) \qquad \text{for all } T \geq 0. \tag{2}$$

For $\mathbf{x}_i \neq \mathbf{0}$ define the *reference-unit* vector

$$\mathbf{z}_i := \frac{g_i^{-1}\big(\|\mathbf{x}_i\|_1\big)}{\|\mathbf{x}_i\|_1}\,\mathbf{x}_i,$$

and set $\mathbf{z}_i = \mathbf{0}$ when $\mathbf{x}_i = \mathbf{0}$. Thus $\mathbf{x}_i$ is rescaled, keeping its direction fixed, so that $\|\mathbf{z}_i\|_1 = g_i^{-1}(\|\mathbf{x}_i\|_1)$. Applying g_i to both sides of this identity gives

$$g_i\big(\|\mathbf{z}_i\|_1\big) = \|\mathbf{x}_i\|_1.$$

Moreover, by (2),

$$c\big(\|\mathbf{z}_i\|_1\big) = c\big(g_i^{-1}(\|\mathbf{x}_i\|_1)\big) = c_i\big(\|\mathbf{x}_i\|_1\big),$$

so cost is exactly preserved.

For $\mathbf{z} \neq \mathbf{0}$, define

$$\tilde{F}_i(\mathbf{z}) := F_i\left(\frac{g_i\big(\|\mathbf{z}\|_1\big)}{\|\mathbf{z}\|_1}\,\mathbf{z}\right),$$

and let $\tilde{F}_i(\mathbf{0}) = F_i(\mathbf{0})$. Evaluating at $\mathbf{z} = \mathbf{z}_i$ for $\mathbf{x}_i \neq \mathbf{0}$ using $g_i\big(\|\mathbf{z}_i\|_1\big) = \|\mathbf{x}_i\|_1$ gives,

$$\tilde{F}_i(\mathbf{z}_i) = F_i\left(\frac{\|\mathbf{x}_i\|_1}{\|\mathbf{z}_i\|_1} \cdot \frac{g_i^{-1}(\|\mathbf{x}_i\|_1)}{\|\mathbf{x}_i\|_1}\,\mathbf{x}_i\right) = F_i(\mathbf{x}_i),$$

where we used $\|\mathbf{z}_i\|_1 = g_i^{-1}(\|\mathbf{x}_i\|_1)$. If $\mathbf{x}_i = \mathbf{0}$, then $\mathbf{z}_i = \mathbf{0}$ and the equality follows directly from $\tilde{F}_i(\mathbf{0}) = F_i(\mathbf{0})$. Thus $F_i(\mathbf{x}_i) = \tilde{F}_i(\mathbf{z}_i)$ as required.

Finally, the map $\mathbf{x}_i \mapsto \mathbf{z}_i$ is a bijection of $\mathbb{R}_{\geq 0}^K$, with inverse

$$\mathbf{x}_i = \frac{g_i(\|\mathbf{z}_i\|_1)}{\|\mathbf{z}_i\|_1}\mathbf{z}_i$$

for $\mathbf{z}_i \neq \mathbf{0}$, and $\mathbf{x}_i = \mathbf{0}$ when $\mathbf{z}_i = \mathbf{0}$.

Applying this construction to every developer i, corresponding strategy profiles $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ and $(\mathbf{z}_1, \ldots, \mathbf{z}_N)$ induce identical outputs and costs for every developer, and hence identical payoffs. Since the correspondence is a bijection for each developer, best responses are preserved under the reparameterization, and therefore the Nash equilibria of the two games are in bijective correspondence. □

F.2 Proof of Theorem 1

Theorem 1 (Equilibrium Existence). *Suppose that $g_{i,k}(\mathbf{x}_i) = [F_i(\mathbf{x}_i)]_k$ is twice continuously differentiable and c convex. Let $\rho_\sigma := \sup_{z\in\mathbb{R}} \frac{\sigma''(z)}{\sigma'(z)}$. There is a pure-strategy Nash equilibrium if*

$$-\nabla^2 g_{i,k}(\mathbf{x}_i) \succeq \rho_\sigma \nabla g_{i,k}(\mathbf{x}_i) \nabla g_{i,k}(\mathbf{x}_i)^\top \qquad \forall\, \mathbf{x}_i,\ \forall\, i, k.$$

Proof. Since σ is bounded in $[0,1]$ and $\mathbf{w} \in \Delta^{K-1}$, the total reward of any developer is at most 1. Moreover, choosing $\mathbf{x}_i = \mathbf{0}$ gives nonnegative utility. Therefore, any action satisfying

$$c(\|\mathbf{x}_i\|_1) > 1$$

cannot be a best response. Let $\bar{x} = c^{-1}(1)$. We may therefore restrict each developer's strategy set to

$$\mathcal{X}_i = \{\mathbf{x}_i \in \mathbb{R}_+^K : \|\mathbf{x}_i\|_1 \le \bar{x}\},$$

which is compact and convex. Since any action outside $\mathcal{X}_i$ yields strictly negative utility, while $\mathbf{x}_i = \mathbf{0}$ yields nonnegative utility, no action outside $\mathcal{X}_i$ can be a profitable deviation. Thus any Nash equilibrium of the restricted game is also a Nash equilibrium of the original game. We note that u_i is continuous because it is built from the addition and multiplication of continuous functions in σ, F_i, and c.

It remains to show that $u_i(\mathbf{x}_i, \mathbf{x}_{-i})$ is concave in $\mathbf{x}_i$. For $z_{ij,k} = g_{i,k}(\mathbf{x}_i) - g_{j,k}(\mathbf{x}_j)$,

$$\nabla^2_{\mathbf{x}_i} \sigma(z_{ij,k}) = \sigma'(z_{ij,k}) \nabla^2 g_{i,k}(\mathbf{x}_i) + \sigma''(z_{ij,k}) \nabla g_{i,k}(\mathbf{x}_i) \nabla g_{i,k}(\mathbf{x}_i)^\top.$$

Since $\sigma'(z) > 0$ and

$$\frac{\sigma''(z)}{\sigma'(z)} \le \rho_\sigma,$$

the assumed curvature condition implies

$$\nabla^2_{\mathbf{x}_i} \sigma(z_{ij,k}) \preceq 0.$$

Thus every pairwise reward term is concave in $\mathbf{x}_i$. Their nonnegative weighted sum is therefore concave, and since c is convex,

$$-c(\|\mathbf{x}_i\|_1)$$

is concave as well. Hence $u_i(\cdot, \mathbf{x}_{-i})$ is concave (and so quasi-concave) on $\mathcal{X}_i$ for every $\mathbf{x}_{-i}$.

Finally, the utilities are continuous and the strategy sets $\mathcal{X}_i$ are nonempty, compact, and convex. Therefore, by the Debreu–Fan–Glicksberg equilibrium existence theorem, the game admits a pure-strategy Nash equilibrium. □

F.3 Proof of Lemma 1

Lemma 1 (Symmetric Developers). *Suppose the conditions of Theorem 1 hold. If $F_i = F_{i'}$, then there exists a pure-strategy Nash equilibrium $\mathbf{x}^*$ such that $\mathbf{x}_i^* = \mathbf{x}_{i'}^*$ and $H_{i,i'}(\mathbf{x}^*) = 1$. Moreover, if $F_i = F$ for every developer i, then there exists a symmetric pure-strategy Nash equilibrium satisfying $\mathbf{x}_1^* = \cdots = \mathbf{x}_N^*$, and at this equilibrium, $\theta_1^* = \cdots = \theta_N^*$, with $H(\mathbf{x}^*) = 1$.*

Proof. By Theorem 1, we may restrict each developer's action space to a compact convex set $\mathcal{X}_i$, and each payoff u_i is continuous and concave in the developer's own action. Hence each best-response correspondence is nonempty, convex-valued, and upper hemicontinuous.

First suppose that $F_i = F_{i'}$. Consider the subset of strategy profiles

$$\mathcal{X}^{i,i'} = \left\{ \mathbf{x} \in \prod_{j=1}^N \mathcal{X}_j : \mathbf{x}_i = \mathbf{x}_{i'} \right\}.$$

At any profile $\mathbf{x} \in \mathcal{X}^{i,i'}$, developers i and i' face identical optimization problems: they have the same production function and cost function, and, because $\mathbf{x}_i = \mathbf{x}_{i'}$, each faces the same actions and outputs from all other developers. Therefore,

$$\mathrm{BR}_i(\mathbf{x}_{-i}) = \mathrm{BR}_{i'}(\mathbf{x}_{-i'}).$$

Since these best-response sets are identical, nonempty, and convex, we can define a reduced best-response correspondence on $\mathcal{X}^{i,i'}$ that requires developers i and i' to select the same element of their common best-response set, while every other developer selects from its usual best-response set. Equivalently, parameterize $\mathcal{X}^{i,i'}$ by a common action $\mathbf{y} = \mathbf{x}_i = \mathbf{x}_{i'}$ and the actions $(\mathbf{x}_\ell)_{\ell \neq i,i'}$. The resulting correspondence is nonempty and convex-valued. Moreover, it is upper hemicontinuous because each developer's best-response correspondence is upper hemicontinuous and the restriction to profiles satisfying $\mathbf{x}_i = \mathbf{x}_{i'}$ is continuous. Thus, by Kakutani's fixed-point theorem, there exists a fixed point $\mathbf{x}^* \in \mathcal{X}^{i,i'}$. At this fixed point every developer is playing a best response in the original game, and, by construction, $\mathbf{x}_i^* = \mathbf{x}_{i'}^*$.

Now suppose that $F_i = F$ for every i. Now consider the fully symmetric subset

$$\mathcal{X}^{\mathrm{sym}} = \left\{ \mathbf{x} \in \prod_{j=1}^{N} \mathcal{X}_j : \mathbf{x}_1 = \cdots = \mathbf{x}_N \right\}.$$

Here, every developer faces the same optimization problem, and therefore faces the same best response problem. We can choose the same best response for each developer under this situation. As above, the resulting correspondence maps $\mathcal{X}^{\mathrm{sym}}$ into itself and satisfies the conditions of Kakutani's fixed-point theorem. Therefore, there exists a fixed point with $\mathbf{x}_i$ constant, which implies $\boldsymbol{\theta}_i$ constant, and consequently $H = 1$. □

F.4 Proof of Lemma 2

Lemma 2 (Similarity in Production Functions)**.** *Suppose the conditions of Theorem 1 hold, while the production functions are L-Lipschitz, and the pseudogradient G is μ-strongly monotone, which ensures equilibrium stability to perturbations:*

$$(G(\mathbf{x}) - G(\mathbf{x}'))^\top (\mathbf{x} - \mathbf{x}') \geq \mu \|\mathbf{x} - \mathbf{x}'\|_2^2 \qquad \forall \mathbf{x}, \mathbf{x}'.$$

If developers i and i' have similar production functions in both value and gradient:

$$\sup_{\mathbf{x}_i \in \mathcal{X}_i} \|F_i(\mathbf{x}_i) - F_{i'}(\mathbf{x}_i)\|_2 \leq \epsilon, \qquad \sup_{\mathbf{x}_i \in \mathcal{X}_i} \|\nabla g_{i,k}(\mathbf{x}_i) - \nabla g_{i',k}(\mathbf{x}_i)\|_2 \leq \epsilon \quad \forall k.$$

then there exists an equilibrium $\mathbf{x}^*$ that satisfies

$$\|\mathbf{x}_i^* - \mathbf{x}_{i'}^*\|_2 \leq O\left(\frac{\epsilon}{\mu}\right), \qquad \|F_i(\mathbf{x}_i^*) - F_{i'}(\mathbf{x}_{i'}^*)\|_2 \leq O\left(\epsilon\left(1 + \frac{L}{\mu}\right)\right).$$

Moreover, if $\|\boldsymbol{\theta}_i^*\|_2, \|\boldsymbol{\theta}_{i'}^*\|_2 \geq m > 0$, then their pairwise monoculture satisfies

$$H_{i,i'}(\mathbf{x}^*) \geq 1 - O\left(\frac{\epsilon^2}{m^2}\left(1 + \frac{L}{\mu}\right)^2\right).$$

Proof. Let $G(\mathbf{x})$ be the pseudogradient for the original game with production functions F_i and $F_{i'}$ and let $\widetilde{G}(\mathbf{x})$ be the pseudogradient for the counterfactual game where F_i is replaced with $F_{i'}$ (while everything else stays constant).

We show that G and $\widetilde{G}$ are close in L2 norm. For each developer ℓ, the corresponding block of the pseudogradient is

$$G_\ell(\mathbf{x}) = -\nabla_{\mathbf{x}_\ell} u_\ell(\mathbf{x}).$$

Since

$$u_\ell(\mathbf{x}) = \frac{1}{N-1} \sum_{r \neq \ell} \sum_{k=1}^{K} w_k \sigma(F_\ell(\mathbf{x}_\ell)_k - F_r(\mathbf{x}_r)_k) - c(\|\mathbf{x}_\ell\|_1),$$

we have

$$\nabla_{\mathbf{x}_\ell} u_\ell(\mathbf{x}) = \frac{1}{N-1} \sum_{r \neq \ell} \sum_{k=1}^{K} w_k \sigma'(F_\ell(\mathbf{x}_\ell)_k - F_r(\mathbf{x}_r)_k) \nabla_{\mathbf{x}_\ell} F_\ell(\mathbf{x}_\ell)_k - \nabla_{\mathbf{x}_\ell} c(\|\mathbf{x}_\ell\|_1).$$

Since the strategy space is compact and the production functions are continuous, the arguments of σ lie in a compact interval. Since σ is twice continuously differentiable, σ' and σ'' are bounded on this interval, so σ' is Lipschitz; moreover, since the production functions are continuously differentiable on the compact strategy space, their gradients are uniformly bounded. Replacing F_i with $F_{i'}$ affects only terms involving developer i. For $\ell = i$, both the argument of σ' and the production-function gradient change; for $\ell \neq i$, only the comparison against developer i changes. Using $\|F_i(\mathbf{x}_i) - F_{i'}(\mathbf{x}_i)\|_2 \leq \epsilon$, $\|\nabla F_i(\mathbf{x}_i)_k - \nabla F_{i'}(\mathbf{x}_i)_k\|_2 \leq \epsilon$, and the Lipschitz continuity of σ', we can bound these differences term-by-term to obtain

$$\sup_{\mathbf{x}\in\mathcal{X}} \|G(\mathbf{x}) - \widetilde{G}(\mathbf{x})\|_2 \leq C\epsilon,$$

for some constant $C > 0$ independent of ϵ.

Next, let $\mathbf{x}^*$ denote the equilibrium of the original game. Because F_i and $F_{i'}$ both satisfy the conditions of Theorem 1, the counterfactual game also satisfies these conditions. By Lemma 1, there exists an equilibrium $\widetilde{\mathbf{x}}^*$ of the counterfactual game such that $\widetilde{\mathbf{x}}^*_i = \widetilde{\mathbf{x}}^*_{i'}$. We select such an equilibrium $\widetilde{\mathbf{x}}^*$. Since each developer's utility is concave and differentiable in its own strategy, the first-order optimality conditions characterize equilibrium. That is, because $\mathbf{x}^*$ and $\widetilde{\mathbf{x}}^*$ are equilibria of the original and counterfactual games, respectively, we have that

$$G(\mathbf{x}^*)^\top(\widetilde{\mathbf{x}}^* - \mathbf{x}^*) \geq 0, \qquad \widetilde{G}(\widetilde{\mathbf{x}}^*)^\top(\mathbf{x}^* - \widetilde{\mathbf{x}}^*) \geq 0.$$

Let $d = \mathbf{x}^* - \widetilde{\mathbf{x}}^*$. Adding the two inequalities gives $(G(\mathbf{x}^*) - \widetilde{G}(\widetilde{\mathbf{x}}^*))^\top d \leq 0$. Recall that $G(\mathbf{x})$ is strongly monotone, and so

$$\left(G(\mathbf{x}) - G(\mathbf{x}')\right)^\top (\mathbf{x} - \mathbf{x}') \geq \mu\|\mathbf{x} - \mathbf{x}'\|_2^2.$$

Setting $\mathbf{x} = \mathbf{x}^*$ and $\mathbf{x}' = \widetilde{\mathbf{x}}^*$ gives

$$(G(\mathbf{x}^*) - G(\widetilde{\mathbf{x}}^*))^\top d \geq \mu\|d\|_2^2.$$

Invoking $(G(\mathbf{x}^*) - \widetilde{G}(\widetilde{\mathbf{x}}^*))^\top d \leq 0$ gives that

$$\mu\|d\|_2^2 \leq (G(\mathbf{x}^*) - G(\widetilde{\mathbf{x}}^*))^\top d \leq (\widetilde{G}(\widetilde{\mathbf{x}}^*) - G(\widetilde{\mathbf{x}}^*))^\top d \leq C\epsilon\|d\|_2,$$

where the final inequality follows from Cauchy–Schwarz and the bound on the difference between the pseudogradients. Rearranging and solving for d gives that

$$\|\mathbf{x}^* - \widetilde{\mathbf{x}}^*\|_2 \leq \frac{C\epsilon}{\mu}.$$

Since $\widetilde{\mathbf{x}}^*_i = \widetilde{\mathbf{x}}^*_{i'}$,

$$\|\mathbf{x}^*_i - \mathbf{x}^*_{i'}\|_2 \leq \|\mathbf{x}^*_i - \widetilde{\mathbf{x}}^*_i\|_2 + \|\mathbf{x}^*_{i'} - \widetilde{\mathbf{x}}^*_{i'}\|_2 \leq O(\|\mathbf{x}^* - \widetilde{\mathbf{x}}^*\|_2) = O\left(\frac{\epsilon}{\mu}\right).$$

Finally, to bound $H_{i,i'}(\mathbf{x}^*)$, let

$$\begin{aligned}
\delta &= \|\boldsymbol{\theta}^*_i - \boldsymbol{\theta}^*_{i'}\|_2 \\
&= \|F_i(\mathbf{x}^*_i) - F_{i'}(\mathbf{x}^*_{i'})\|_2 \\
&\leq \|F_i(\mathbf{x}^*_i) - F_i(\mathbf{x}^*_{i'})\|_2 + \|F_i(\mathbf{x}^*_{i'}) - F_{i'}(\mathbf{x}^*_{i'})\|_2 \\
&\leq L\|\mathbf{x}^*_i - \mathbf{x}^*_{i'}\|_2 + \epsilon \\
&= O\left(\epsilon\left(1 + \frac{L}{\mu}\right)\right).
\end{aligned}$$

Next, recall that

$$H_{i,i'}(\mathbf{x}^*) = \frac{\langle \boldsymbol{\theta}^*_i, \boldsymbol{\theta}^*_{i'}\rangle}{\|\boldsymbol{\theta}^*_i\|_2\|\boldsymbol{\theta}^*_{i'}\|_2}.$$

Let $\widehat{\boldsymbol{\theta}}^*_i = \boldsymbol{\theta}^*_i/\|\boldsymbol{\theta}^*_i\|_2$ and $\widehat{\boldsymbol{\theta}}^*_{i'} = \boldsymbol{\theta}^*_{i'}/\|\boldsymbol{\theta}^*_{i'}\|_2$. Since both normalized vectors have unit norm,

$$\|\widehat{\boldsymbol{\theta}}^*_i - \widehat{\boldsymbol{\theta}}^*_{i'}\|_2^2 = 2 - 2\langle\widehat{\boldsymbol{\theta}}^*_i, \widehat{\boldsymbol{\theta}}^*_{i'}\rangle = 2(1 - H_{i,i'}(\mathbf{x}^*)).$$

If $\|\boldsymbol{\theta}_i^*\|_2, \|\boldsymbol{\theta}_{i'}^*\|_2 \geq m$, then, adding and subtracting $\boldsymbol{\theta}_{i'}^*/\|\boldsymbol{\theta}_i^*\|_2$ and applying the triangle inequality,

$$\begin{aligned}\left\| \frac{\boldsymbol{\theta}_i^*}{\|\boldsymbol{\theta}_i^*\|_2} - \frac{\boldsymbol{\theta}_{i'}^*}{\|\boldsymbol{\theta}_{i'}^*\|_2} \right\|_2 &= \left\| \frac{\boldsymbol{\theta}_i^* - \boldsymbol{\theta}_{i'}^*}{\|\boldsymbol{\theta}_i^*\|_2} + \boldsymbol{\theta}_{i'}^* \left(\frac{1}{\|\boldsymbol{\theta}_i^*\|_2} - \frac{1}{\|\boldsymbol{\theta}_{i'}^*\|_2} \right) \right\|_2 \\ &\leq \frac{\|\boldsymbol{\theta}_i^* - \boldsymbol{\theta}_{i'}^*\|_2}{\|\boldsymbol{\theta}_i^*\|_2} + \|\boldsymbol{\theta}_{i'}^*\|_2 \left| \frac{1}{\|\boldsymbol{\theta}_i^*\|_2} - \frac{1}{\|\boldsymbol{\theta}_{i'}^*\|_2} \right| \\ &= \frac{\|\boldsymbol{\theta}_i^* - \boldsymbol{\theta}_{i'}^*\|_2}{\|\boldsymbol{\theta}_i^*\|_2} + \frac{|\|\boldsymbol{\theta}_i^*\|_2 - \|\boldsymbol{\theta}_{i'}^*\|_2|}{\|\boldsymbol{\theta}_i^*\|_2} \\ &\leq \frac{2\|\boldsymbol{\theta}_i^* - \boldsymbol{\theta}_{i'}^*\|_2}{m} = \frac{2\delta}{m}.\end{aligned}$$

Therefore,

$$2(1 - H_{i,i'}(\mathbf{x}^*)) \leq \frac{4\delta^2}{m^2},$$

and hence

$$H_{i,i'}(\mathbf{x}^*) \geq 1 - \frac{2\delta^2}{m^2} = 1 - O\left(\frac{\epsilon^2}{m^2} \left(1 + \frac{L}{\mu} \right)^2 \right).$$

□

F.5 Proof of Lemma 3

Lemma 3 (Specialized Production Functions)**.** *Consider two developers i and i' with disjoint sets $S_i, S_{i'} \subseteq [K]$. Suppose that, for each $j \in \{i, i'\}$, there exists $\gamma_j \in [0, 1]$ such that*

$$\left\| F_j(\mathbf{x}_j)_{S_j^c} \right\|_2 \leq \gamma_j \left\| F_j(\mathbf{x}_j)_{S_j} \right\|_2 \qquad \forall \mathbf{x}_j \in \mathcal{X}_j.$$

Then any equilibrium $\mathbf{x}^*$ with $\|\boldsymbol{\theta}_i^*\|_2, \|\boldsymbol{\theta}_{i'}^*\|_2 > 0$ satisfies

$$H_{i,i'}(\mathbf{x}^*) \leq \frac{\gamma_i + \gamma_{i'}}{\sqrt{1 + \gamma_i^2}\sqrt{1 + \gamma_{i'}^2}}.$$

Proof. By assumption, each developer's production function primarily produces output in its corresponding set of dimensions. That is,

$$\left\| F_i(\mathbf{x}_i)_{S_i^c} \right\|_2 \leq \gamma_i \left\| F_i(\mathbf{x}_i)_{S_i} \right\|_2.$$

Evaluating this condition at the equilibrium $\mathbf{x}^*$, where $\boldsymbol{\theta}_i^* = F_i(\mathbf{x}_i^*)$, gives

$$\left\| \boldsymbol{\theta}_{i,S_i^c}^* \right\|_2 \leq \gamma_i \left\| \boldsymbol{\theta}_{i,S_i}^* \right\|_2, \qquad \left\| \boldsymbol{\theta}_{i',S_{i'}^c}^* \right\|_2 \leq \gamma_{i'} \left\| \boldsymbol{\theta}_{i',S_{i'}}^* \right\|_2.$$

Next, we analyze the geometry of this situation to show that $\boldsymbol{\theta}_i^*$ and $\boldsymbol{\theta}_{i'}^*$ are nearly orthogonal. Let α_i denote the angle between $\boldsymbol{\theta}_i^*$ and its projection onto the dimensions S_i, $\boldsymbol{\theta}_{i,S_i}^*$. Because the components on S_i and S_i^c are orthogonal,

$$\tan(\alpha_i) = \frac{\|\boldsymbol{\theta}_{i,S_i^c}^*\|_2}{\|\boldsymbol{\theta}_{i,S_i}^*\|_2} \leq \gamma_i,$$

and therefore $\alpha_i \leq \arctan(\gamma_i)$. Analogously, if $\alpha_{i'}$ denotes the angle between $\boldsymbol{\theta}_{i'}^*$ and its projection onto $S_{i'}$, then $\alpha_{i'} \leq \arctan(\gamma_{i'})$.

Since $S_i \cap S_{i'} = \emptyset$, vectors supported on S_i and $S_{i'}$ are orthogonal. Therefore, the angle between $\boldsymbol{\theta}_i^*$ and $\boldsymbol{\theta}_{i'}^*$ is at least

$$\angle(\boldsymbol{\theta}_i^*, \boldsymbol{\theta}_{i'}^*) \geq \frac{\pi}{2} - \alpha_i - \alpha_{i'} \geq \frac{\pi}{2} - \arctan(\gamma_i) - \arctan(\gamma_{i'}).$$

Since $\gamma_i \in [0, 1]$, this lower bound is nonnegative. It follows that

$$\begin{aligned}H_{i,i'}(\mathbf{x}^*) = \cos\left(\angle(\boldsymbol{\theta}_i^*, \boldsymbol{\theta}_{i'}^*)\right) &\leq \cos\left(\frac{\pi}{2} - \arctan(\gamma_i) - \arctan(\gamma_{i'}) \right) \\ &= \sin(\arctan(\gamma_i) + \arctan(\gamma_{i'})) \leq \frac{\gamma_i + \gamma_{i'}}{\sqrt{1 + \gamma_i^2}\sqrt{1 + \gamma_{i'}^2}}.\end{aligned}$$

□

F.6 Proof of Theorem 2

Theorem 2 (Directional Characterization of Monoculture and Specialization)**.** *Suppose that the conditions of Theorem 1 hold. A profile of directions $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N) \in (S_+^{K-1})^N$ is induced by a pure-strategy Nash equilibrium with nonzero outputs if and only if $\mathbf{D}$ is locally supportable.*

A pure-strategy equilibrium with monoculture exists if and only if $(\mathbf{d}, \ldots, \mathbf{d})$ is locally supportable for some $\mathbf{d} \in S_+^{K-1}$, and a pure-strategy equilibrium with complete specialization exists if and only if there is a locally supportable profile $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_N)$ with $\mathbf{d}_i^\top \mathbf{d}_{i'} = 0$ for all $i \neq i'$.

Proof. First suppose that $D = (\mathbf{d}_1, \ldots, \mathbf{d}_N)$ is induced by a pure-strategy Nash equilibrium $\mathbf{x}^*$ with nonzero outputs. Let $\alpha_i = \|F_i(\mathbf{x}_i^*)\|_2$ and $\mathbf{d}_i = \frac{F_i(\mathbf{x}_i^*)}{\|F_i(\mathbf{x}_i^*)\|_2}$, so that $F_i(\mathbf{x}_i^*) = \alpha_i \mathbf{d}_i$ for every developer i.

For each developer i and capability k, define its marginal evaluation weight at $\mathbf{x}^*$ as

$$p_{i,k}^* = \frac{w_k}{N-1} \sum_{j \neq i} \sigma'\big(F_i(\mathbf{x}_i^*)_k - F_j(\mathbf{x}_j^*)_k\big)\,.$$

Since $F_i(\mathbf{x}_i^*) = \alpha_i \mathbf{d}_i$, this is exactly

$$p_{i,k}^* = p_{i,k}(D, \boldsymbol{\alpha}) = \frac{w_k}{N-1} \sum_{j \neq i} \sigma'(\alpha_i d_{i,k} - \alpha_j d_{j,k})\,.$$

The gradient of developer i's utility with respect to its investment is therefore

$$\nabla_{\mathbf{x}_i} u_i(\mathbf{x}^*) = \nabla F_i(\mathbf{x}_i^*)^\top \mathbf{p}_i(D, \boldsymbol{\alpha}) - c'(\|\mathbf{x}_i^*\|_1)\mathbf{1}.$$

Because $\mathbf{x}^*$ is an equilibrium, the first-order optimality conditions imply

$$\nabla F_i(\mathbf{x}_i^*)^\top \mathbf{p}_i(D, \boldsymbol{\alpha}) \leq c'(\|\mathbf{x}_i^*\|_1)\mathbf{1},$$

with equality in every coordinate ℓ such that $x_{i,\ell}^* > 0$. Thus D is locally supportable.

Conversely, suppose that D is locally supportable. Then there exist $\boldsymbol{\alpha} \in \mathbb{R}_{++}^N$ and $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ such that

$$F_i(\mathbf{x}_i) = \alpha_i \mathbf{d}_i$$

and

$$\nabla F_i(\mathbf{x}_i)^\top \mathbf{p}_i(D, \boldsymbol{\alpha}) \leq c'(\|\mathbf{x}_i\|_1)\mathbf{1},$$

with equality whenever $x_{i,\ell} > 0$.

At this profile,

$$p_{i,k}(D, \boldsymbol{\alpha}) = \frac{w_k}{N-1} \sum_{j \neq i} \sigma'(F_i(\mathbf{x}_i)_k - F_j(\mathbf{x}_j)_k)\,,$$

and hence the conditions above are exactly the first-order optimality conditions for developer i's original payoff $u_i(\mathbf{x}_i, \mathbf{x}_{-i})$.

By the curvature argument in the proof of Theorem 1, $u_i(\cdot, \mathbf{x}_{-i})$ is concave on $\mathbb{R}_+^K$. Therefore, these first-order conditions are sufficient for optimality so $\mathbf{x}_i$ is a best response to $\mathbf{x}_{-i}$ for every developer i. It follows that $\mathbf{x}$ is a pure-strategy Nash equilibrium, and its normalized output directions are exactly D.

Finally, complete monoculture occurs if and only if

$$\mathbf{d}_1 = \cdots = \mathbf{d}_N,$$

so the first characterization follows by setting $D = (\mathbf{d}, \ldots, \mathbf{d})$. Similarly, because output vectors are nonnegative, complete specialization occurs if and only if

$$\mathbf{d}_i^\top \mathbf{d}_j = 0 \qquad \forall i \neq j.$$

The second characterization therefore follows directly. □

F.7 Proof for Proposition 1

Proposition 1 (Catch-Up and Long-Run Crowding). *Let $\boldsymbol{\theta}_i^t = F_i(\mathbf{X}_i^t)$ denote developer i's capability vector at time t, and define its category-specific gap to the frontier as $g_{i,k}^t = \max_j \theta_{j,k}^t - \theta_{i,k}^t$. Let $\mathbf{g}^t \in \mathbb{R}_+^{NK}$ stack these gaps across developers and categories at timestep t, and let*

$$\Delta\boldsymbol{\theta}^t = \left(\boldsymbol{\theta}_1^{t+1} - \boldsymbol{\theta}_1^t, \ldots, \boldsymbol{\theta}_N^{t+1} - \boldsymbol{\theta}_N^t\right) \in \mathbb{R}^{NK}.$$

Suppose each F_i is continuously differentiable with

$$\max_{\ell \in [K]} \left\| \frac{\partial F_i(\mathbf{x})}{\partial x_\ell} \right\|_2 \to 0 \qquad \text{as } \|\mathbf{x}\|_1 \to \infty, \quad \forall i.$$

Suppose there exists $\beta > 0$ such that, for sufficiently large t, $\left\langle \mathbf{g}^t, \Delta\boldsymbol{\theta}^t \right\rangle \geq \beta \|\mathbf{g}^t\|_2^2$. Then we have that $\|\mathbf{g}^t\|_2 \to 0$, and $\max_{i'} \theta_{i',k}^t - \theta_{i,k}^t \to 0$ for all developers i and categories k.

Proof. Recall that at each time t, developer i chooses $\mathbf{x}_{i,t}$ to maximize the win-rate, minus an unbounded cost function. The win-rate is at most 1, while the cost function is increasing and unbounded, so there exists $B < \infty$ such that $c(B) > 1$. Therefore, at any equilibrium, $\|\mathbf{x}_{i,t}\|_1 \leq B$ for every i, t.

Next, we deal with two cases on whether the cumulative investments $\{\mathbf{X}_i^t\}_{t \geq 1}$ are bounded or not, and show that in either case $\|F_i(\mathbf{X}_i^{t+1}) - F_i(\mathbf{X}_i^t)\|_2 \to 0$. First, suppose that $\{\mathbf{X}_i^t\}_{t \geq 1}$ is bounded. Because $\mathbf{X}_i^t$ is the sum of t non-negative numbers, if sum converges, then the individual terms must go to zero; $\sum_t \|\mathbf{x}_{i,t}\|_1 < \infty$, and so $\|\mathbf{x}_{i,t}\|_1 \to 0$. Because $\|\mathbf{x}_{i,t}\|_1 \to 0$, then $\|\mathbf{X}_i^{t+1} - \mathbf{X}_i^t\|_1 \to 0$. Because F_i is continuously differentiable, its Jacobian is bounded on the compact set containing $\{\mathbf{X}_i^t\}_{t \geq 1}$, and so

$$\|F_i(\mathbf{X}_i^{t+1}) - F_i(\mathbf{X}_i^t)\|_2 \to 0.$$

Instead, suppose that $\{\mathbf{X}_i^t\}_{t \geq 1}$ is unbounded, so $\|\mathbf{X}_i^t\|_1 \to \infty$. We know that $\|X_i^{t+1} - X_i^t\|_1 \leq B$. We can then use the mean-value inequality to show that

$$\|F_i(\mathbf{X}_i^{t+1}) - F_i(\mathbf{X}_i^t)\|_2 \leq B \sup_{\|\mathbf{z}\|_1 \geq \|\mathbf{X}_i^t\|_1} \max_{\ell \in [K]} \left\| \frac{\partial F_i(\mathbf{z})}{\partial z_\ell} \right\|_2.$$

As $t \to \infty$, $\|\mathbf{X}_i^t\|_1 \to \infty$, and by assumption, we have that $\left\| \frac{\partial F_i(\mathbf{z})}{\partial z_\ell} \right\|_2 \to 0$. As a result, $\|F_i(\mathbf{X}_i^{t+1}) - F_i(\mathbf{X}_i^t)\|_2 \to 0$, so $\|\Delta\boldsymbol{\theta}^t\|_2 \to 0$.

We next use the catch-up property. By Cauchy–Schwarz, we have that

$$\beta \|\mathbf{g}^t\|_2^2 \leq \left\langle \mathbf{g}^t, \Delta\boldsymbol{\theta}^t \right\rangle \leq \|\mathbf{g}^t\|_2 \|\Delta\boldsymbol{\theta}^t\|_2.$$

If $\|\mathbf{g}^t\|_2 > 0$, dividing by $\|\mathbf{g}^t\|_2$ gives

$$\|\mathbf{g}^t\|_2 \leq \frac{1}{\beta} \|\Delta\boldsymbol{\theta}^t\|_2.$$

Because $\|\Delta\boldsymbol{\theta}^t\|_2 \to 0$ and β is a constant, we have that $\|\mathbf{g}^t\|_2 \to 0$. By definition of $\mathbf{g}$ this implies that

$$g_{i,k}^t = \max_{i'} \theta_{i',k}^t - \theta_{i,k}^t \to 0 \qquad \text{for every } i, k.$$

□

F.8 Sufficient Conditions for Catch-Up

To introduce sufficient conditions for catch-up, we first introduce the idea of a common investment component. For any pair of developers i, i' define their common investment component $\mathbf{d}_{i,i',t} = \min\{\mathbf{x}_{i,t+1}, \mathbf{x}_{i',t+1}\}$ componentwise, and let $\mathbf{r}_{j,t} := \mathbf{x}_{j,t+1} - \mathbf{d}_{i,i',t}, j \in \{i, i'\}$, denote developer j's unmatched investment. The key idea is that when production functions have sufficiently strong diminishing relative returns and different developers share a sufficiently large common investment component, we can lower bound β.

Proposition 3. *Suppose each F_i is continuously differentiable, coordinatewise nondecreasing, and L-Lipschitz with respect to ℓ_1. Suppose further that there exists $\kappa > 0$ such that, for every pair of developers i, i', category k, investment states $\mathbf{x}, \mathbf{y}$, and investment coordinate ℓ, whenever $F_i(\mathbf{x})_k \leq F_{i'}(\mathbf{y})_k$,*

$$\frac{\partial F_i(\mathbf{x})_k}{\partial x_\ell} - \frac{\partial F_{i'}(\mathbf{y})_k}{\partial y_\ell} \geq \kappa \left(F_{i'}(\mathbf{y})_k - F_i(\mathbf{x})_k\right).$$

Suppose additionally that, there exist $a > 0$ and $0 \leq \rho < 1$ for any i, i', k, t with $\theta^t_{i,k} \leq \theta^t_{i',k}$ such that $\|\mathbf{d}_{i,i',t}\|_1 \geq a$ and

$$L\|\mathbf{r}_{i',t}\|_1 \leq \rho \left(1 - e^{-\kappa\|\mathbf{d}_{i,i',t}\|_1}\right) \left(\theta^t_{i',k} - \theta^t_{i,k}\right).$$

Then the directed pairwise performance gap contracts by at least $\beta = (1-\rho)(1-e^{-\kappa a}) > 0$.

Proof. Fix i, i', k, t with $\theta^t_{i',k} \geq \theta^t_{i,k}$, and write $\mathbf{d} = \mathbf{d}_{i,i',t}$. Define

$$h(s) = F_{i'}(\mathbf{X}^t_{i'} + s\mathbf{d})_k - F_i(\mathbf{X}^t_i + s\mathbf{d})_k, \qquad s \in [0,1].$$

Whenever $h(s) \geq 0$, we have that

$$\frac{\partial F_i(\mathbf{X}^t_i + s\mathbf{d})}{\partial x_\ell} - \frac{\partial F_{i'}(\mathbf{X}^t_{i'} + s\mathbf{d})_k}{\partial y_\ell} \geq \kappa \left(F_{i'}(\mathbf{X}^t_{i'} + s\mathbf{d})_k - F_i(\mathbf{X}^t_i + s\mathbf{d})_k\right) = \kappa h(s).$$

Then by the chain rule, we have that

$$\begin{aligned} h'(s) &= \sum_\ell d_\ell \left(\frac{\partial F_{i'}(\mathbf{X}^t_{i'} + s\mathbf{d})_k}{\partial x_\ell} - \frac{\partial F_i(\mathbf{X}^t_i + s\mathbf{d})_k}{\partial y_\ell}\right) \\ &\leq -\kappa h(s) \sum_\ell d_\ell = -\kappa\|\mathbf{d}\|_1 h(s). \end{aligned}$$

Using properties of derivatives, we get

$$\frac{d}{ds}\log h(s) \leq -\kappa\|\mathbf{d}\|_1.$$

We can then bound $h(1)$ by integration from 0 to 1 (or until $h(s)$ is negative) to get

$$[h(1)]_+ \leq e^{-\kappa\|\mathbf{d}\|_1} h(0) = e^{-\kappa\|\mathbf{d}\|_1}(\theta^t_{i',k} - \theta^t_{i,k}).$$

Now write $\mathbf{x}_{j,t+1} = \mathbf{d} + \mathbf{r}_{j,t}$ for $j \in \{i, i'\}$. Because F_i is coordinatewise non-decreasing, $\mathbf{r}_{i,t}$ can only reduce the gap, while we can bound the difference through Lipschitzness

$$\theta^{t+1}_{i',k} - \theta^{t+1}_{i,k} \leq e^{-\kappa\|\mathbf{d}\|_1}(\theta^t_{i',k} - \theta^t_{i,k}) + L\|\mathbf{r}_{i',t}\|_1.$$

We then bound the Lipschitz term by $\rho\left(1 - e^{-\kappa\|\mathbf{d}\|_1}\right)\left(\theta^t_{i',k} - \theta^t_{i,k}\right)$ to get

$$\theta^{t+1}_{i',k} - \theta^{t+1}_{i,k} \leq \left[1 - (1-\rho)\left(1 - e^{-\kappa\|\mathbf{d}\|_1}\right)\right](\theta^t_{i',k} - \theta^t_{i,k}) \leq (1-\beta)(\theta^t_{i',k} - \theta^t_{i,k}),$$

for $\beta = (1-\rho)(1-e^{-\kappa a}) > 0$.

□