IFN Working Paper No. 1575, 2026

Repricing by Replacement: Pricing Closed Language Models Against a Free Alternative

Martin Ljunge

Research Institute of Industrial Economics
P.O. Box 55665
SE-102 15 Stockholm, Sweden
info@ifn.se
www.ifn.se

Repricing by Replacement:

Pricing Closed Language Models Against a Free Alternative

Martin Ljunge*

November 23, 2026

Abstract

For 8 months Anthropic sold Claude Opus 4.1 at 3 times the price of the more capable Opus 4.5. Unlike chips or cars, closed language models rarely get cheaper with age. In validated list prices of OpenAI, Anthropic, Google, and xAI, the price of a given capability fell 45 percent a year, yet posted prices were cut only 8 times. Launches account for 54 percent of the decline and withdrawals for 32. Per task, closed models cost about 4.0 times as much as equally capable open-weight models, the free alternative. A product-line model with menu costs explains repricing by replacement.

Keywords: versioning, competing with free, menu costs, hedonic prices, large language models.
JEL codes: L11, L13, L15, L86, M21, O33.

1 Introduction

When Anthropic launched Claude Opus 4.5, it priced the new model at 25 dollars per million output tokens, the units of text a model writes. The older Opus 4.1 cost 75 dollars. The new version was the more capable of the two, yet Anthropic sold both at those prices for 8 months, until it withdrew the older one. Usage did not wait that long. On the OpenRouter marketplace (OpenRouter, 2026; Aubakirova et al., 2026), Opus 4.1 handled 0.8 percent of the two versions' tokens in the 22 weeks of usage records in which both were listed. In many

*Research Institute of Industrial Economics (IFN), Stockholm. Email: martin.ljunge@ifn.se. I thank Anders Sandrews Stiftelse (the Anders Sandrew Foundation) and Jan Wallanders och Tom Hedelius stiftelse (the Jan Wallander and Tom Hedelius Foundation, grant P23-0186) for financial support. The funders had no influence on the research.

product markets a product gets cheaper over its life, as chips do when better chips arrive (Aizcorbe and Kortum, 2005) and cars over their model year (Copeland et al., 2011). As a rule, a closed language model keeps its price until it leaves the list.

The pattern holds across the closed providers, the firms that sell access to their models and keep the weights to themselves. We follow the list prices of 102 model versions sold by OpenAI, Anthropic, Google, and xAI in 121 weekly snapshots from July 2024 to October 2026. Over that period the price of a given capability fell 45 percent a year, while the price of a version that stayed on sale fell 4 percent (Figure 1, top panel). The providers cut the price of a version already on sale only 8 times. A listed version's price was cut in 0.9 percent of the months in which it could have been, while Nakamura and Steinsson (2008) report that regular consumer prices change in 9 to 12 percent of months. When a provider launched a version that beat its previous best, the price of the previous best was cut within a week in 1 of 46 cases. The providers reprice by replacement instead, and a replacement has two dates. At the launch a new version sells capability for less, and launches account for 54 percent of the decline. The withdrawal comes months later. When the old price leaves the list the measured price of capability falls again, and withdrawals account for another 32 percent. For 21 of the 29 versions withdrawn, the provider already listed a cheaper version of at least the same measured capability, so the withdrawal offered buyers nothing new. It ended the old price. A buyer therefore collects the decline only by moving, and one who stays on a named model keeps paying the old price until the provider withdraws it.

One source of pressure on these prices is public and dated. Open-weight models have their trained parameters published for anyone to run, and we call them the free alternative. Independent hosts run them on their own hardware and sell access by the token, so the free alternative has a posted price. On benchmark tasks a closed model costs about 4.0 times as much as an open model of equal capability. The releases of open models are dated, and their capability is scored on the same benchmarks as the closed models'. We can therefore ask a question about competition that is seldom answerable. When the free alternative improves, which of an incumbent's posted prices are worth changing? The rare cuts tend to fall on versions near the best open model's capability, which we call the open floor. On average the list prices of top versions show no detectable response.

The prices come from a community-maintained file of list prices for every major API. We validate the file against archived captures of the providers' own pricing pages, where 94.2 percent of 258 model-date checks agree, trace every price change to the provider's records, and correct the file's errors. We join the 102 closed versions to Epoch's Capabilities Index (ECI), which scores open and closed models on one scale, and add the hosted prices of 91 open-weight models on 65 hosting providers, the Artificial Analysis index with the tokens

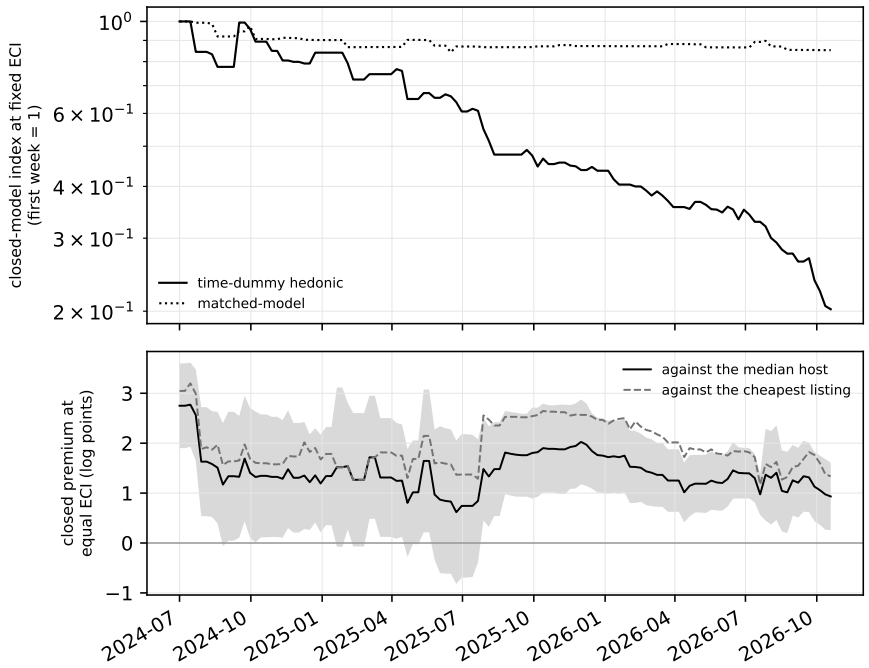

Figure 1: Closed prices at equal capability, and the closed premium over the free alternative. The two panels make different comparisons. Top, closed models over time: price indexes for closed models, weekly, log scale. The time-dummy hedonic index follows the price of a given capability across the versions on sale, and the matched-model index the price of versions that stay on sale (Section 4). Bottom, closed against open models: the closed premium at equal ECI (log points) among models no more than ten ECI points below the best open model, against the median host (solid, with 95 percent confidence bands) and against the cheapest listing (dashed; Section 6.1). Figure E.1 in Online Appendix E adds the hosted open-weight indexes.

each model uses on a fixed task set, the providers' retirement records, and OpenRouter's usage by version.

Table 1 collects four findings, three of them measured and one resting on few events. First, the quality-adjusted decline in closed prices arrives through the turnover of versions. Launches and withdrawals together account for 86 percent of the decline. On the Artificial Analysis index, over a shorter window, the decline is slower at 12 percent a year for the same providers, again with almost no repricing. Second, closed models sell at a large price premium over the free alternative at equal capability (Figure 1, bottom panel). Per token the premium shows no clear trend on the host prices of the community file, the main source, and it narrows on those of Artificial Analysis. It is a price premium, not a markup, since the closed providers' serving cost is unobserved and the premium may also pay for attributes that the capability index does not measure. Third, on few events, the timing of the rare cuts

Table 1: Price adjustment through the product line: the main findings

Finding	Main evidence	Interpretation
Prices adjust through launches and withdrawals	The price of a fixed capability falls 45% a year (12% on the Artificial Analysis index, same providers); a given model's list price falls 4%	Measured. Launches and withdrawals carry 86% of it here, and less on a chained index (29% a year, 7% for continuing versions)
Closed models cost a multiple of open models at equal capability	Closed models cost 4.2× as much per token at the median host, with a trend of −0.14 log points a year (0.28). In benchmark data that count tokens they cost 6.0× per token and 4.0× per task	Measured. A price premium, not a markup, with no clear trend on the main source
Posted cuts are rare and tend to follow frontier open releases	8 provider cuts in 7 decisions in 121 weeks, one cut per 501 closed model-weeks at risk, and one tariff change at the end of a preview. The decision rate after frontier open releases exceeds the rate before by 0.042 a week (two-sided $p = 0.096$; after the 11 major releases alone, $p = 0.281$), with no such rise after closed launches	Suggestive, on 7 decisions, three of them OpenAI's. The cuts fall on young versions ($p = 0.0001$) and mostly near the open floor (suggestive). The causal share of open competition is not identified
Versions are withdrawn, not repriced	50 of 52 retirement announcements fall below the open floor, every one on a version that a cheaper hosted open model matched or beat; rate ratio 8.1	Measured. Not separable from age: the floor overtakes nearly every version within a year, and the rate ratio at equal age is 1.7

Notes: The rows draw on Sections 4, 6.1, 6.2, and 6.3, in that order. The p-values of the third row redraw the dates of the providers' cut decisions. Measured marks a descriptive estimate, and Suggestive an estimate in the predicted direction that is imprecise or not robust. Online Appendix F lists the paper's claims with their design and strength.

fits a product-line model with costly repricing, and their location fits it in part. The rate of cut decisions is higher after open-weight releases at the frontier than before them (two-sided $p = 0.096$ over 7 decisions), but not after open releases far below the open floor, new closed frontier models, or closed launches near the open floor. Fourth, versions are withdrawn rather than repriced. Of the 52 retirement announcements, 50 fall below the open floor, but by then the floor has overtaken nearly every version, so the data cannot separate being overtaken from being old.

Pricing against a free alternative is not peculiar to AI. Software vendors compete with open-source projects, publishers with free content, and branded drug makers with generics, but there the free alternative's quality is hard to measure and changes slowly. The language-model market shows the incumbent's response in weekly data, and Online Appendix F sets it beside two benchmark markets, processors and prescription drugs.

We interpret the results through a model of a product line sold to buyers who can always use the free alternative, an outside option whose value rises with the buyer's taste for quality. The provider's profit then separates into one term for each cutoff between adjacent versions,

and only the lowest cutoff involves the free alternative. When posted prices are free to move, an improvement of the free alternative lowers every price by the same absolute amount. A cost of changing a posted price, a menu cost, changes this for legacy prices, the posted prices of versions already on sale. The cuts fall on the lowest versions that keep buyers, never on a version while a lower one keeps its price and its buyers. With uniformly distributed types the problem has an exact solution, in which the cuts shrink up the line and reach the top only after large improvements. The menu cost also shapes launches. At a launch, a legacy version can keep selling beside the new version at its old price, lose its buyers to a new version priced against the free alternative's current quality, or have its price reset. When repricing is costly, adjustment runs through new versions and withdrawals, and a version left without buyers stays listed while withdrawing it costs more than keeping it.

The paper contributes to the study of rare price changes and of adjustment through the replacement of products (Nakamura and Steinsson, 2008; Bils, 2009; Argente et al., 2018). In the LLM market quality is measured on one scale for every product, closed or open, so the part of the decline that new versions carry can be priced, and the improvements of the free alternative are publicly dated. Zhu (2026) and Wang (2026) also find that new models carry most of the fall in inference prices and of the gain in consumer surplus. This paper adds a verified record of list prices, the shares of the decline that arrive at the launch and at the withdrawal, and the release date, capability score, and hosted price of the open competitors. The paper also brings evidence to the theory of versioning and of competition with free and open-source products (Casadesus-Masanell and Ghemawat, 2006; Economides and Katsamakas, 2006; Casadesus-Masanell and Zhu, 2010; Bhargava and Choudhary, 2008; Johnson and Myatt, 2003), whose mechanisms have rarely been observed with prices, quality, and product turnover measured together. It measures the price of the free alternative and the incumbent's premium over it, per token and per task, with a correction for token intensity that the descriptive literature on falling inference prices does not make (Epoch AI, 2025; Gundlach et al., 2025; Demirer et al., 2026). The model is the upgrades structure of Johnson and Myatt (2003) with the free alternative in the entrant's place. What is new in it is the exact loss from stale prices, the block structure of cuts under a menu cost, and the two candidate launch prices that a stale legacy price leaves a provider. Its predictions differ from those of the token-menu model of Bergemann et al. (2026), in which the proprietary leader serves the highest types as a monopolist would.

The paper establishes where the adjustment happens and measures its size. It does not measure how much of the decline the free alternative causes. The evidence that open releases prompt cuts rests on the timing of 7 cut decisions and is suggestive (Table 5), and it is silent on launches and withdrawals, which carry most of the decline.

2 Related Literature

Price rigidity, product turnover, and obsolescence. Posted prices change rarely (Nakamura and Steinsson, 2008), and a large share of price adjustment takes place through product replacement (Bils, 2009; Argente et al., 2018; Aghion et al., 2019). With a fixed cost of changing a price a firm adjusts in discrete steps (Sheshinski and Weiss, 1977), and a multiproduct firm that pays one cost to reprice all its goods changes them together (Midrigan, 2011; Alvarez and Lippi, 2014). In our model each price change costs the same and the free alternative presses on one end of the line, so the cuts start at the bottom. A product's sales decline over most of its life as newer products, including the firm's own, make it obsolete (Argente et al., 2024). Durable-goods and software monopolists may retire or upgrade products to manage obsolescence (Bulow, 1986; Ellison and Fudenberg, 2000). In the LLM market a retirement is a forced migration of the provider's own customers. We show that retirements fall almost only on versions the free alternative has overtaken and that the distance to the provider's own newer versions does not predict them, although neither can be separated from the version's age.

Versioning and product lines. In the theory of second-degree price discrimination with quality, a seller offers a menu and the buyers who value quality more choose the better and dearer versions (Mussa and Rosen, 1978). Together with the theory of vertical differentiation (Shaked and Sutton, 1982), it underlies the analysis of versioning information goods (Shapiro and Varian, 1998; Bhargava and Choudhary, 2008; Jones and Mendelson, 2011) and of damaged goods (Deneckere and McAfee, 1996). Later work shows how marginal, usage, and versioning costs shape a line of versions (Chellappa and Mehra, 2018) and how software is priced nonlinearly when buyers cannot vary their usage (Xin and Sundararajan, 2020). Johnson and Myatt (2003) show that an incumbent may answer entry by adding fighting brands, cheap versions aimed at the entrant's customers, at the bottom of its line or by pruning it, and Johnson and Myatt (2018) show how cost differences between firms decide whether their lines compete head to head or separate by quality. Our model puts in the entrant's place a free outside option whose value rises with the buyer's type, so that the participation constraint, the condition that a buyer prefers the menu to its outside option, varies with the type, as in Jullien (2000). Here the free alternative's quality is below the lowest version's, so the constraint binds only at the bottom of the line and the free alternative moves only the lowest cutoff. The closest paper is Bergemann et al. (2026), who model LLM pricing as menus of token budgets and, in an extension, a proprietary leader facing an open-source fringe that sells tokens at marginal cost. There, low types buy only from the

fringe, intermediate types are offered quantities that deter top-up from the fringe, and high types are served as under monopoly. In our model buyers choose among versions of different quality, the free alternative is a substitute rather than a source of extra tokens, and the top of the line is shielded from it in level only by costly repricing, sticky legacy prices at launch, or a closed rival between them that keeps buyers.

Competing with free and open products. How proprietary firms compete with open-source or free rivals has been studied mostly in theory or in single firms. The work covers demand-side learning (Casadesus-Masanell and Ghemawat, 2006), platforms with complementary applications (Economides and Katsamakas, 2006), the restrictiveness of licences (August et al., 2021, 2018), why developers and firms contribute to open code (Lerner and Tirole, 2002; Kumar et al., 2011), and the share of free content over the demand cycle (Lambrecht and Misra, 2017). Closest to our results, an incumbent facing an ad-sponsored rival may reconfigure its business model rather than cut price (Casadesus-Masanell and Zhu, 2010), and competition from free but less usable open-source software can lead a proprietary vendor to raise both quality and price (Zhou and Choudhary, 2022). The LLM market adds measured quality for both sides, a sequence of dated quality jumps of the free alternative, and a product line whose versions are launched, repriced, and retired in public.

Generic entry and incumbent response. After generic entry, brand-name drug prices fall little (Caves et al., 1991) or even rise (Grabowski and Vernon, 1992; Frank and Salkever, 1997), while generics take a growing share of prescriptions (Frank and Salkever, 1997), a pattern that market segmentation between loyal and price-sensitive buyers explains (Frank and Salkever, 1992; Regan, 2008). In microprocessors, Goettler and Gordon (2011) estimate how competition from AMD shapes Intel's innovation and prices. Online Appendix F compares both markets with the LLM market.

The economics of LLM markets. Demirer et al. (2026) describe the business-to-business market for inference with OpenRouter data and report that open-source models are about 90 percent cheaper than comparable closed ones. Fradkin (2025) documents substitution and multihoming after model releases, Aubakirova et al. (2026) describe usage on the same platform, and Korinek and Vipra (2025) discuss scale economies and concentration. Descriptive work shows that the price of a fixed benchmark score falls by an order of magnitude a year or more (Epoch AI, 2025; Appenzeller, 2024; Gundlach et al., 2025). Zhu (2026) builds quality-adjusted indexes of inference prices and finds, as we do, that matched-model indexes record little of a decline that hedonic indexes put at about half a log point a year. Wang (2026)

recovers providers' markups from a demand system on OpenRouter usage and attributes most of the gain in consumer surplus to new models rather than to price cuts on existing ones. The markups are inferred under prices re-optimized each month, and our records show how rarely the providers changed a continuing version's list price in those months. Measurement follows the hedonic literature for semiconductors and computers (Pakes, 2003; Aizcorbe, 2006; Byrne et al., 2018; Flamm, 2018; Nordhaus, 2007).

3 Data

List prices. litellm, an open-source library that routes developers' requests to the providers' interfaces, maintains a public file of list prices per token for every major API, with a dated commit history. We take one snapshot per week from 1 July 2024 to 19 October 2026, 121 in all. Closed models are those listed under OpenAI, Anthropic, Google (Gemini), and xAI with a chat or responses mode and positive prices. Prices are per million output tokens, the text a model writes; the text it reads, the input tokens, is billed separately. A blended price of three parts input to one part output, a common mix of reading and writing, gives similar results where we report it. Closed models of other developers, such as Alibaba's Qwen-Max line and Mistral's, are outside the sample. Each listing is matched to a model version: a listing whose name carries a release date is matched to the version released within two weeks of that date, and an undated name, an alias that a provider moves from one version to the next, follows the providers' documented alias moves (Online Appendix D).

Validation and verified price events. A community file can be wrong, so we validate it twice. First, we compare it with archived captures of the providers' own pricing pages from the Wayback Machine, and 94.2 percent of 258 model-date checks agree on the output price. Second, every change in a model's price in the weekly panel is checked against the provider's pages and the commit that made it. Of 20 price events, 8 are provider cuts, 2 are temporary promotional cuts, one is an alias that moved to a newer version, one is a change of tariff when a model left preview (Section 4), and 8 are errors in the file: corrections of wrong launch prices, typos, and stale entries left after a model's retirement. The file's dates are not reliable either. It never recorded one of the 8 cuts, which we found from the provider's page, and it entered three others 5 to 15 weeks after the provider's change. We correct the errors, date each provider cut from the provider's own announcement or pricing page, and drop listings after a model's shutdown. The clean panel has 102 closed models and 4,835 closed model-weeks. Of these, 4,006 are at risk of a cut: the weeks after a version's first week in the panel and before any retirement announcement, which removes the 728 weeks that

53 versions spent between their announcement and their shutdown, one of them announced before the panel began.

Capability. Epoch's Capabilities Index (October 2026 export) gives one score per model version, its release date, its developer, and whether its weights are open. For each model-week the *open leader* is the best open-weight model released by that date, and the *open floor* is the open leader's ECI, the capability a buyer can have without a closed provider. A closed model's *lead over the open floor* is its ECI minus the floor, negative below it. The *closed frontier* is the ECI of the best closed model of any developer, and a model's *distance to the closed frontier* is the frontier minus its own ECI. An open-weight release is a *frontier release* if its ECI is no more than one point below the prior open leader's, and a *major* release if it raised the open floor by at least 1.5 points. From January 2024 to October 2026 there are 25 frontier releases and 11 major ones, 8 of them from Chinese developers. We record each open model's licence, total and active parameters (the parameters used on each token, which set the cost of serving a model), and country. The index puts every model, open or closed, on one scale, so a point means the same across models and dates.

The price of the free alternative. The same file lists open-weight models on many hosts. We match the hosted entries to the open-weight models in the ECI, excluding batch tiers, fine-tunes, and distillations. We drop 58 listings that appear for a single week and vanish, and we screen out 613 listings priced more than twenty times above or below the median of the other hosts of the same model in the same week, mostly a units error in one host's prices (Online Appendix F). This leaves 30,016 price observations on 91 open-weight models across 65 hosts. The cheapest host for each model and week, the cheapest listing, gives the best price of the free alternative, and the median host its typical price.

A second capability measure and tokens per task. Artificial Analysis publishes, for several hundred models, an Intelligence Index computed on a fixed set of evaluations, the tokens each model uses to run it, prices at the provider's own API and at each third-party host, open-weight status, licence, and active parameters. We parse 60 archived captures (snapshots) of its models page between January 2024 and October 2026 and use the 52 that hold the whole page, 11,162 model-capture records. The index is rescaled when the evaluation set changes. We detect 5 versions of the index from the data and compare models only within a capture, where all models face the same tasks. The cost of a task is the tokens a model uses on the index times its prices, at the provider's own price for closed models and at the cheapest benchmarked host for open-weight models.

Retirements. A retirement, which the providers call a deprecation, is the announced withdrawal of a version from sale. The provider announces it first and shuts the version down later, and the shutdown is the withdrawal that ends a replacement. We date a retirement by its announcement. The deprecation pages of OpenAI, Anthropic, Google (Gemini API and Vertex AI), Azure OpenAI, Amazon Bedrock, Mistral, and Cohere, together with xAI's model listings, give 645 retirement records with the date of the announcement, the shutdown date, and the named successor. Matched to the panel, 52 of the 100 closed models at risk received a retirement announcement in the sample. For the 50 with a shutdown date, that date is a median of 5 months after the announcement. Online Appendix D gives the rules.

Data availability. All inputs are public: the litellm file's commit history, archived pages from the Internet Archive, Epoch's Capabilities Index, the archived Artificial Analysis and OpenRouter pages, the Medicaid State Drug Utilization Data, and Wikipedia's processor lists. A replication package that rebuilds every table, figure, and number in the text from these inputs will be made available upon journal publication. OpenRouter has permitted the use of its public rankings data, which the paper credits to OpenRouter (OpenRouter, 2026; Aubakirova et al., 2026).

4 Adjustment Through Versions

A hedonic index prices a fixed bundle of characteristics over time, so it tracks the price of capability rather than the price of any one model. We estimate time-dummy hedonic regressions of the log list price on the ECI, a reasoning-model indicator fixed for each version, the log context window (the amount of text a model can take in at once), and provider and week effects. Each week effect is the average, over the versions on the list that week, of the log price net of what the characteristics predict. Every listed version has the same weight, whatever its usage. A matched-model index replaces the characteristics with model fixed effects, so it follows the price of a version that stayed on sale (Pakes, 2003; Aizcorbe, 2006).

Table 2 and the top panel of Figure 1 report the results. The price of a given capability falls 45 percent a year (log slope -0.600, standard error 0.141), with nearly the same rate at monthly frequency, 43 percent a year excluding the premium "pro" tiers, and 46 percent on blended prices. The price of a version that stayed on sale falls 4 percent a year, so the matched-model index misses 94 percent of the decline.

To see where the decline comes from, we follow the versions that enter and leave the price list each week, in an exact accounting of the hedonic index's annual log trend (Online Appendix F). A version that enters below the average lowers the index, and so does a version

that leaves from above it. The 85 launches after the panel's opening week entered below comparable listed versions and account for 54 percent of the decline. The withdrawals are the 29 versions whose last listing fell within six weeks of their shutdown. They removed relatively expensive versions and account for 32 percent. Of the 52 retirement announcements of Section 3, 27 ended in such a withdrawal before the panel closes. Price changes of continuing versions account for 12 percent, and other entries and exits account for the rest. A withdrawal lowers the index without putting a new price in front of any buyer. For 21 of the 29 versions withdrawn, the same provider already listed a cheaper version with at least the withdrawn version's ECI. The cheaper offer was on the list already, and the withdrawal took the old price off it. The model of Section 5 explains why adjustment runs through versions when repricing is costly. It does not predict these shares.

The size of the decline depends on the measure of capability and on the index, but its route does not. On the Artificial Analysis index, which is built from recent, reasoning-heavy evaluations, closed prices of all developers on the index fall 26 percent a year at fixed capability between March 2025 and July 2026, and those of the four providers 12 percent, while a given closed model's list price does not fall. A chained Fisher-type imputation index in the spirit of Aizcorbe (2006) and Byrne et al. (2018), which imputes the price of an entering or exiting version from the hedonic function and gives every version equal weight, falls 29 percent a year, against 7 percent for continuing versions alone (Table 2 and Online Appendix C). A dollar buys more capability each year, mostly through new versions.

List prices are sticky. Within a model, list prices rarely change. Beyond the 8 provider cuts, the clean panel has only 2 other price cuts, the temporary promotions. In a within-model regression of the log price on the lead over the open floor and a trend, the coefficient on the lead is 0.004 (0.003). One tariff changed. When Google took Gemini 2.5 Flash out of preview in June 2025, it replaced the preview's two output prices by one. The preview had cost 0.60 dollars per million tokens with the model's reasoning mode switched off and more with it on, and the single price is 2.50 dollars (Online Appendix C).

Launch prices. At a given capability, new versions are launched at lower prices over time. Across the 87 launches, the closed models first listed within forty-five days of their release (85 of them after the opening week), the log launch price rises by 0.227 per ECI point (0.026) and changes by −3.19 log points a year at given capability (0.45). This slope is far steeper than the hedonic index's. Among launches capability rises steadily with the date, so the launch regression holds capability fixed only by extrapolating, and its slope should not be read as a rate of decline (Online Appendix C). Without holding capability fixed, the log

Table 2: Quality-adjusted price indexes: closed models and the free alternative

Index	Change per year	Reading
Closed models (litellm list prices, ECI), July 2024 – October 2026		
Time-dummy hedonic, weekly	−45%	log slope −0.600 (0.141); monthly −44%
excluding pro tiers; blended price	−43%; −46%	
Matched model	−4%	log slope −0.039; share of the decline it misses 94%
accounting: launches; withdrawals; continuing versions	−0.325; −0.189; −0.072 log points	shares of the annual log trend 54%; 32%; 12%
Chained Fisher-type imputation, equal weights	−29%	pooled imputations; adjacent-period −35%; monthly −28%
base basket; current basket	−26%; −32%	continuing versions alone −7%
Closed models of all developers (Artificial Analysis index and prices), March 2025 – July 2026		
Time-dummy hedonic, chained across index versions	−26%	four providers −12%
Matched model	+0.5%	providers' own list prices
Hosted open-weight models (the free alternative)		
Cheapest host, ECI	−36%	ECI coefficient 0.037 (0.008)
with log active parameters	−18%	elasticity 0.67 (0.14)
all hosts, host effects	−25%	
Matched model, cheapest host	−31%	mixes host entry with the file's coverage (Online Appendix E)
Matched model and host	−2%	a given host's price of a given model
Matched model, cheapest benchmarked host (Artificial Analysis)	+2%	index coefficient 0.003 (0.006), active-parameter elasticity 0.68 (0.10)

Notes: Log price per million output tokens on week effects and ECI, a reasoning indicator, log context window, and provider effects (closed); on week effects and ECI, with log active parameters or host effects as stated (hosted). In the closed regression a point of ECI adds 0.149 log points (0.023) and a reasoning model −1.50 (0.55). The accounting's share for continuing versions exceeds what the matched-model index implies because the two weight the same price changes differently. Annual changes are $100(e^b - 1)$ for the OLS slope b of the week effects on time in years; standard errors of the slope in parentheses, from the covariance of the week effects clustered by model. The Artificial Analysis rows use the captures that hold the whole page. The price file marks reasoning models only from April 2025, so a version counts as a reasoning model if the file marks it, or its generally available version, as one at any date (72 of the 102 versions). The Artificial Analysis hedonic index is estimated within each index version and chained across versions by the median matched-model price change.

launch price rises by 0.31 a year, so a provider can raise through a new version a price it almost never raises on the list. The same holds per task (Online Appendix F).

The cost of staying put. The other side of sticky list prices is what a buyer who stays on a named model pays. For the 31 closed models launched in the sample and listed for at least a year, the list price a year after the first listing was a median of 10 times the price of the cheapest closed model then listed with at least the same ECI, and 3.3 times the cheapest such model of the same provider. At launch the median ratio was 1.0. The comparison takes

the ECI as the measure of capability and ignores the attributes that separate tiers. The two indexes of Table 2 give the counterpart at fixed characteristics. A named model's price falls 4 percent a year, and the price the list puts on the same characteristics falls 45 percent. After a year, relative to launch, the named model therefore costs about 1.8 times what the list then charges for its characteristics.

What buyers use. Usage moves with the versions. On OpenRouter, a marketplace that routes developers' requests to closed and open models (OpenRouter, 2026; Aubakirova et al., 2026), archived pages report the tokens of every model in 78 weeks between July 2024 and May 2026. Of the four providers' tokens on versions in our panel, 72 percent went to versions under six months old, which make up 40 percent of the listed version-weeks, and so did 88 percent of spending at list prices. Within Anthropic's Opus line, Opus 4.1 handled 0.8 percent of the two versions' tokens while both were listed. Usage had moved to the cheaper version long before the old price left the list, and that is how the decline reaches buyers.

5 A Product Line Against a Free Alternative

At a launch a provider prices the new version and must decide what to do with the old one. Leaving the old price alone keeps the old margin on the buyers who do not move. Cutting it may win buyers from the open models, but it also draws buyers from the provider's other versions, and changing a posted price is costly. The model first asks which legacy prices are worth changing when the free alternative improves, and then lets the provider launch a better version. Appendix A proves the results stated here, and Online Appendix B works through an example and states and proves the further results, including those that Proposition 4 collects.

5.1 Setup

The model is a product line in the sense of Mussa and Rosen (1978), with the free alternative in the place that the entrant's product takes in Johnson and Myatt (2003): a seller offers versions of rising quality at rising prices, buyers sort themselves, and those who value quality more buy the better versions. Two features carry the argument. First, the free alternative is an option open to every buyer, and its quality is below that of the cheapest closed version, so the buyers closest to switching to it are those at the bottom of the line. In the data the best open model soon scores above most closed versions, and what then matters for a version is the best open model hosted at no more than its price (Remark B.3). Second, changing a

posted price is costly, so the seller may leave a price where the free alternative has made it wrong. Throughout, a legacy version is one already on sale, and its price is stale once the free alternative has improved since it was set.

A unit mass of buyers has types $\theta \in [0, \bar{\theta}]$ with distribution F and a continuously differentiable density $f > 0$. A buyer's type is its value of quality: a buyer with high-stakes tasks has a high type. Each buyer has a task to complete. A buyer of type θ who completes it with a model of quality q at price p per task obtains $\theta q - p$. The free alternative has quality q_o and costs c_o per task to use, the price at which competing hosts serve it. A closed provider sells versions $j = 1, \ldots, J$ of qualities $q_o < q_1 < \cdots < q_J$, set by its training runs, at posted prices $p_1, \ldots, p_J$, with marginal costs $c_1 \leq \cdots \leq c_J$. Write $\Delta q_1 = q_1 - q_o$, $\Delta c_1 = c_1 - c_o$, and $\Delta q_k = q_k - q_{k-1}$, $\Delta c_k = c_k - c_{k-1}$ for $k \geq 2$, and let $h(\theta) = (1 - F(\theta))/f(\theta)$ be the inverse hazard rate, the mass of buyers above θ relative to the density at θ.

Assumption 1 (Monotone hazard)**.** *h is nonincreasing.*

Assumption 2 (Increasing incremental cost)**.** $-h(0) < \Delta c_1/\Delta q_1 < \Delta c_2/\Delta q_2 < \cdots < \Delta c_J/\Delta q_J < \bar{\theta}$.

The expression $\theta - h(\theta)$ is the virtual valuation, the marginal revenue from extending a version to a buyer of type θ. It is what that buyer pays less what lowering the cutoff gives up on every buyer above it. Under Assumption 1 it rises strictly from $-h(0)$ to $\bar{\theta}$, so Assumption 2 places each version's incremental cost per unit of quality strictly inside its range, which is what gives every version a segment of buyers. Because every buyer completes the task, the free alternative is the outside option of every type, and its value $\theta q_o - c_o$ rises with the type. Remark B.3 in Online Appendix B adds the option of not using a model, and several open models, of which the one that matters for the marginal buyer need not be the most capable. A buyer of type θ prefers version 1 to the free alternative if $\theta \geq \theta_1 \equiv (p_1 - c_o)/\Delta q_1$ and version k to version $k-1$ if $\theta \geq \theta_k \equiv (p_k - p_{k-1})/\Delta q_k$. The cutoffs are the types at which a buyer is indifferent between two neighboring options, and they divide the buyers into segments, one per option: the types below θ_1 use the free alternative, those between θ_1 and θ_2 buy version 1, and so on up the line. We call the menu active if every option has buyers, $0 < \theta_1 < \cdots < \theta_J < \bar{\theta}$. Prices can then be written in terms of the cutoffs as $p_j = c_o + \sum_{k \leq j} \theta_k \Delta q_k$.

Lemma 1 (Separability)**.** *If the menu is active, the provider's profit is*

$$\Pi = \sum_{k=1}^{J} \big(1 - F(\theta_k)\big)\big(\theta_k \Delta q_k - \Delta c_k\big). \tag{1}$$

Each buyer above cutoff θ_k pays the increment $\theta_k \Delta q_k$ and costs the increment Δc_k, so profit is a sum of one term per cutoff, and only the first term involves the free alternative. A buyer who moves up from version 1 to version 2 pays for the quality step between them, and an improvement of the free alternative leaves that step unchanged. What it changes is the price of the first step, from the free alternative to version 1, which every closed price contains. Everything that follows rests on this separability.

5.2 Which legacy prices change

Proposition 1 (Floor pricing)**.** *Under Assumptions 1–2, the optimal cutoffs solve* $(\theta_k - h(\theta_k))\Delta q_k = \Delta c_k$ *for* $k = 1, \ldots, J$*, and the optimal menu is active. The lowest version is priced above the free alternative by* $p_1 - c_o = \theta_1(q_1 - q_o)$*, the quality gap valued at the marginal buyer's type. Only* θ_1 *depends on* (q_o, c_o)*.*

Proposition 2 (Parallel shift)**.** *While Assumption 2 holds, an improvement of the free alternative changes every optimal price by the same amount,*

$$\frac{\partial p_j}{\partial q_o} = \frac{\theta_1 h'(\theta_1) - h(\theta_1)}{1 - h'(\theta_1)} \equiv -s < 0 \qquad \text{for all } j, \tag{2}$$

and leaves the cutoffs $\theta_2, \ldots, \theta_J$ *unchanged. A fall in the hosting cost passes through to every price at the rate* $\partial p_j / \partial c_o = -h'(\theta_1)/(1 - h'(\theta_1)) \in [0, 1)$*. The proportional effect,* s/p_j*, is largest for the cheapest version.*

With uniform types on $[0, 1]$, $h(\theta) = 1 - \theta$, the optimal cutoffs are $\theta_k = \frac{1}{2}(1 + \Delta c_k / \Delta q_k)$, and $p_1 = \frac{1}{2}(q_1 - q_o + c_1 + c_o)$ is linear in (q_o, c_o), so every price falls by exactly half of any improvement of the free alternative and by half of any fall in the hosting cost, as long as the menu stays active. A fall common to every closed version's cost moves every price by half of it as well, so a parallel shift does not reveal its source. The buyers who switch to the free alternative are the marginal buyers of version 1. In this frictionless benchmark every price falls by the same amount, so the top version's price falls least in proportion. Prices would rise when the free alternative improves only if the hazard rate fell steeply at the marginal buyer, the local counterpart of the generic-entry paradox of brand-name prices that rise after generic entry (Section 2), which Assumption 1 rules out (Remark B.1). A closed rival at a fixed price between the free alternative and the provider's top insulates the versions above it in level, as long as the rival keeps buyers (Remark B.2). Figure 2 draws a line of three versions.

Suppose now that changing a posted price costs $\kappa > 0$ per version, a menu cost. Every change then costs the same, but the profit lost by leaving a price unchanged differs along the

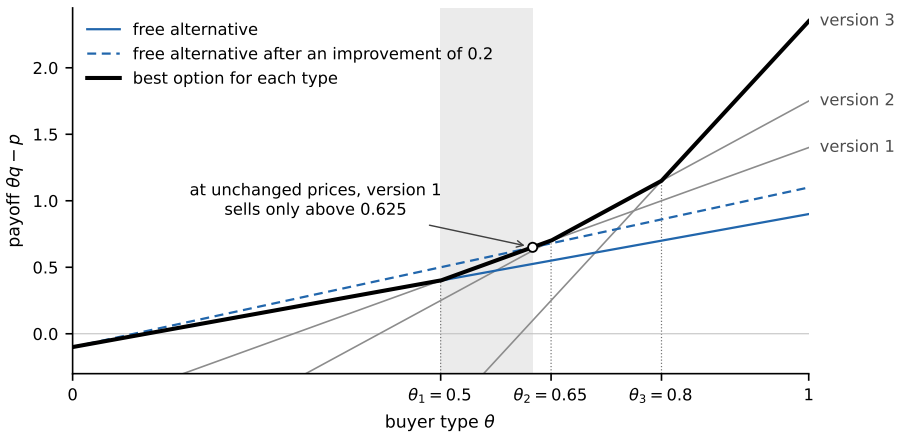

Figure 2: The product line against the free alternative, in the worked example of Online Appendix B.1. Each line is an option's payoff $\theta q - p$ to a buyer of type θ, at the optimal prices of Proposition 1 with uniform types, and the thick line is each type's best option. The cutoffs θ_1, θ_2, and θ_3 divide the types among the free alternative and the three versions. The dashed line is the free alternative after its quality rises by 0.2. At unchanged prices it takes the buyers of version 1 with types between 0.5 and 0.625 (shaded) and no others, because the higher cutoffs depend only on closed prices. Cutting every price by 0.1 restores all three cutoffs (Proposition 2).

line. As in other markets, the cost is managerial and reputational rather than the reprinting of a price list (Zbaracki et al., 2004; Anderson and Simester, 2010). A cut on a listed version can be read in public as a sign that the version is losing ground, and enterprise contracts can carry negotiated terms and price protection that a change in the list price reopens. A cut presented with a new lineup at a launch escapes both, which is why Section 5.3 treats launches as the cheap occasion for repricing. Start from the optimal menu of Proposition 1. The free alternative then improves by $\Delta > 0$, once and unexpectedly, and the provider compares the menu costs of repricing with the profit that the old prices lose in one period. Types are uniform on $[0, 1]$, which makes the problem exact. The first statement of part (b) below holds for any distribution that satisfies Assumption 1, and Proposition B.7 gives the local counterpart of the other parts. Write $\ell_k = q_k - q_o - \Delta$ for the lead of version k over the improved free alternative. The formulas compare cuts that keep every version bought, as cuts do when the improvement is small. A deep cut on a higher version can instead leave a lower one without buyers at its stale price, which can pay when the lower version adds little to the line.

Proposition 3 (Where repricing happens)**.** *Let the provider cut the price of version j by $d_j \geq 0$, with $d_j = 0$ if it leaves the price unchanged, and write $d_0 = \Delta/2$. (a) Among cuts*

that keep every version bought, the profit lost relative to the new optimal menu is exactly

$$\mathcal{L}(d) = \frac{(d_0 - d_1)^2}{\ell_1} + \sum_{k=2}^{J} \frac{(d_{k-1} - d_k)^2}{\Delta q_k}. \tag{3}$$

Leaving every price unchanged loses $\Delta^2/(4\ell_1)$, which rises as the lowest version's lead over the free alternative shrinks. (b) For any improvement that leaves the free alternative below the lowest closed version, and any distribution that satisfies Assumption 1, an optimal set of cuts includes no version without buyers and no version above an uncut version that keeps buyers, so the versions cut are those with buyers below the lowest uncut version that keeps buyers. Among cuts that keep every version bought they form a block $\{1, \ldots, k\}$ at the bottom of the line, or there are none. With uniform types, the best cuts for a block $k < J$ are

$$d_i = \frac{\Delta}{2}\Big(1 - \frac{\ell_i}{\ell_{k+1}}\Big), \qquad i \le k, \tag{4}$$

which shrink up the block and leave a loss $R_k = \Delta^2/(4\ell_{k+1})$. Cutting the whole line lowers every price by $\Delta/2$ and leaves no loss, $R_J = 0$. The provider chooses the block $k \in \{0, \ldots, J\}$ that minimizes $k\kappa + R_k$. (c) Among such cuts, version k is cut only if $\ell_k \le \Delta^2/(4\kappa)$, so versions far above the free alternative keep their prices. The top version is cut, together with every other, if $J\kappa \le \Delta^2/(4\ell_J)$, and only if $\kappa \le \Delta^2/(4\ell_J)$. The smallest and the largest optimal blocks grow weakly with the improvement and shrink weakly with the menu cost. (d) No price changes unless the improvement exceeds a threshold Δ^*, which falls as the lowest version's lead closes, holding the steps between versions fixed. With a single version whose lead before the improvement is $\bar{\ell}_1 = q_1 - q_o$, $\Delta^* = 2(\sqrt{\kappa(\kappa + \bar{\ell}_1)} - \kappa) \approx 2\sqrt{\kappa\bar{\ell}_1}$, provided the version still has buyers at that improvement.

Cuts fall on the lowest versions that keep buyers, never on a version while a lower one keeps its price and its buyers. A provider can therefore pass over a version that has lost its buyers, and whether it has is seen in sales, not in list prices. Within a partial block the cut is largest on the lowest version and shrinks up the line. Only when the whole line is cut, which needs an improvement large relative to the menu cost and the top's lead, does every price fall by the same amount, the parallel shift of Proposition 2. A fall common to the closed versions' costs also moves only the first term of (1), so under a menu cost it brings the same cuts at the bottom of the line. The formulas apply equally to the improvement accumulated since the provider last set its prices: after several open releases without a cut, Δ is the sum of their steps, provided the line's prices were all set at the same date, so the case for a cut grows with every release left unanswered. A provider that sells several lines, each to its own

buyers, faces the free alternative at the bottom of each, and the proposition applies line by line. In the data a provider's tiers (small, standard, and pro) are the candidates for such lines.

5.3 Launches: coexistence, replacement, or a reset

New versions arrive at dates and qualities set by training, which the model takes as given. When a new version is launched, a legacy version can keep selling beside it at its stale price (coexistence), be left without buyers by a new version priced to take them (replacement), or have its price cut along with the launch (a reset). Replacement in the model is complete, in that the legacy version keeps no buyers although it can stay listed, while in the data it is a matter of degree. With uniform types, consider one legacy version of quality q and cost $c \geq c_o$ whose price p was optimal when the free alternative stood at $q_o - A$, under the cost ordering of Assumption 2 at that date, so that $A \geq 0$ is the improvement accumulated since, which plays the role of Δ. The provider launches a version of quality $Q = q + v$ and cost C at a price it chooses freely, and can keep the legacy version at its stale price, reprice it at a one-time cost $\kappa_L < \kappa$, or withdraw it at a one-time cost X, the cost of migrating its users and ending its contracts. A listed legacy version costs m a period, and the line has D discounted periods left. Write $\ell = q - q_o$ for its lead over the current free alternative, $a = c - c_o$, and $r = (C - c)/v$, and let

$$L = \frac{A^2}{4\ell}, \qquad B = \frac{v(r\ell - a)^2}{4\ell(\ell + v)} \tag{5}$$

Here L is the profit lost per period by keeping the legacy price stale. B is the legacy version's value to the line, the profit of the repriced two-version menu less that of the new version sold alone. It is what the provider loses by withdrawing the legacy version once both prices are reset. The value B falls as the free alternative closes in, because the segment the legacy version serves, between the free alternative and the new version, narrows, and it vanishes when the legacy version's cost per unit of quality, a/ℓ, reaches the new version's incremental ratio r, the point at which the legacy version would drop out of an optimal menu anyway.

Proposition 4 (Launches: coexistence, replacement, or a reset)**.** *(a) With uniform types and the legacy price in place, the launch price is one of two candidates. Each is the new version's stand-alone price, the price it would carry if sold alone against the free alternative, and the two differ in the date at which the free alternative's quality is taken. Under coexistence the new version is priced at* $P_S = \frac{1}{2}(Q - q_o + A + C + c_o)$*, against the free alternative's quality when the legacy price was set, and the legacy version keeps buyers. Under replacement it*

is priced at $P_R = \frac{1}{2}(Q - q_o + C + c_o)$*, against the current quality, and the legacy version is left without buyers at its stale price.* $P_S - P_R = A/2$*, and neither price depends on the legacy version's quality or cost. Replacement is infeasible at* $A = 0$*, feasible from a threshold* $A_R > 0$*, and chosen from a higher threshold* A^**, at which the launch price drops by* $A^*/2$*. (b) Under the conditions of Proposition B.6, the provider chooses among the feasible options the one with the smallest loss per period: coexistence,* $L + m$*; keeping a displaced legacy version listed,* $B + m$*; resetting its price,* $\kappa_L/D + m$*; or withdrawing it,* $B + X/D$*.* L *rises and* B *falls as the free alternative improves, so a reset is chosen only for moderate improvements, and a displaced version stays listed while* $m < X/D$*.*

A provider whose repricing cost is high relative to the loss from a stale price and to the legacy version's value adjusts through new versions and withdrawals, and its posted prices stay where they were set. This is the pattern of Section 4. The proposition combines Propositions B.3 and B.6 of Online Appendix B, which also give the thresholds, extend the reset to a whole line (Proposition B.4), and treat a new version launched into a sticky menu at the top or the bottom of the line (Proposition B.2). Uniform types are needed for the two candidates to be the stand-alone prices at two dates and for the formulas in (5), but not for the two-candidate structure of the launch price, which holds for any distribution that satisfies Assumption 1 (Remark B.5). With uniform types, neither regime passes the predecessor's own quality or cost through to the launch price, so a predecessor's price alone does not reveal which regime a launch is in. The order of the two prices says more, since a legacy version keeps buyers beside a better one only if the better one costs more. A legacy version left without buyers stays listed if withdrawing it costs more than listing it for the rest of the line's life. With constant costs the provider withdraws it at once or never. A withdrawal some months after the launch, as in the data, needs an exit cost that falls as contracts end, which the model leaves out. Withdrawal also follows directly from the free alternative. An open model at least as good as a version and no more expensive takes all of its buyers at its posted price, and if serving the version costs at least the open model's hosted price, no cut restores them profitably (Proposition B.5).

Together the propositions give the predictions we take to the data.

P1. *Adjustment through versions.* While the free alternative's improvements stay below the thresholds of Proposition 3, continuing versions keep their prices, and launches and withdrawals carry the adjustment (Proposition 4).

P2. *Floor pricing and the premium.* The cheapest closed version is priced above the free alternative by the quality gap at the marginal buyer's valuation (Proposition 1). In the

model a premium at equal measured capability is the price of attributes the capability index does not measure, so Section 6.1 measures the premium and does not test it.

P3. *Timing and location of cuts.* Cuts follow large improvements of the free alternative, or large accumulations of them since the last cut. They fall at the bottom of a line, in the order of Proposition 3, and not on versions far above the free alternative or on versions that an open model dominates, at least as good and no more expensive, and that cost at least its hosted price to serve (Proposition B.5). A line with more of its life ahead is cut after smaller improvements (Proposition B.1).

P4. *Withdrawal rather than repricing.* A version that an open model dominates, and that costs at least the hosted price to serve, is withdrawn rather than repriced. Withdrawal can come earlier, when a new version replaces it, or later, while withdrawing it costs more than keeping it listed (Propositions 4 and B.5).

P5. *Launches and the top of the line.* The top version's list price responds to the free alternative only after an improvement large relative to the menu cost and the top's lead (Proposition 3). In a sticky menu, a new top is priced off the old top and a new bottom version against the free alternative (Proposition B.2). A new top that replaces its predecessor, or a launch that resets the line, is priced against the free alternative's current quality, and only a reset cuts posted legacy prices (Propositions 4 and B.4).

6 The Free Alternative and the Product Line

Section 4 showed that prices adjust through versions. This section asks what the free alternative has to do with it, taking predictions P2 to P5 to the data, and the answers are mixed. The premium over the free alternative is large, and the few posted cuts tend to follow frontier open releases. Retirement announcements concentrate below the open floor but cannot be told from age, launch prices show no detectable response to open gains, and versions that an open model dominates stay in use.

6.1 The price of the free alternative and the closed premium

The free alternative's own price decline is hard to pin down. The cheapest hosted price of an open-weight model of given ECI falls 36 percent a year in the litellm listings, but this index is fragile. Hosted prices scale with an open model's size, measured by its active parameters, rather than with what it can do, and the litellm file adds cheap hosts late (Table 2 and

Online Appendices C and E). We therefore treat the hosted price as the free alternative's cost of use at a point in time, and not as a reliable index of its decline.

Per token. We pool closed models and hosted open-weight models no more than ten ECI points below the open floor and regress the log price on the ECI, week effects, and a closed indicator for each week. The week-specific closed coefficient is the premium at equal capability. We take the open model's price at the median host, which does not hinge on any single listing, and at its cheapest listing. Against the median host the premium averages 1.43 log points (0.24), a multiple of 4.2, with a trend of −0.14 log points a year (0.28; Table 3 and the bottom panel of Figure 1) that the data cannot tell from zero. Against the cheapest listing it averages 1.95 log points, a multiple of 7.0, with a trend of −0.02 a year (0.29). On the host prices of Artificial Analysis it narrows by 0.74 log points a year (0.37; Online Appendix F).

Per task. The Artificial Analysis data give the same comparison per token and per task in one sample. Within each capture, among models no more than ten index points below the index's open leader, a closed model costs 6.0 times as much per token and 4.0 times as much per task (1.79 and 1.40 log points, with standard errors of 0.20 and 0.19). The two differ because open models write 1.9 times as many tokens per index run (−0.66 log points for closed relative to open, 0.13; Figure 3). The correction, 0.40 log points, is smaller than the output-token ratio alone would give. The bill for a run also has an input part, each model's input tokens at its input price, on which the models differ less (−0.18 log points, 0.14), and the per-token premium is on a blended price with a fixed mix of the two. The 4.2 against the median host above comes from litellm output prices, other hosts, and the full period, and the 6.0 here from blended prices at the cheapest benchmarked host and the closed models of every developer on the page, so the two are separate estimates.

Attributes beyond capability. Equal capability is not equal service. Models and hosts differ in speed, latency, context window, and numerical precision, and the premium could pay for those, but it does not shrink when they are held fixed (Table F.4 in Online Appendix F). In the Artificial Analysis data, holding output speed and latency fixed raises the premium from 1.80 to 2.33 log points per token (0.23), because the closed models are faster and faster models are cheaper at equal capability. The remainder also pays for attributes we cannot measure, such as reliability, rate limits, tool support, and data-handling and indemnity terms, and open models score worse on private benchmarks than on public ones, which suggests that they are tuned to the public benchmarks from which the indexes are built (Edwards and

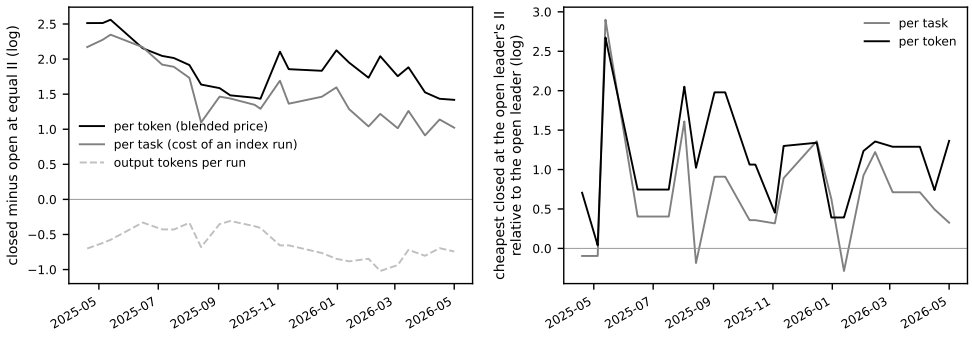

Figure 3: Per token and per task. Left: closed minus open at equal Artificial Analysis index, within each capture, among models no more than ten points below the index's open leader: log blended price per token, log cost of an index run, and log output tokens per run. Right: the cheapest closed model at or above the open leader's index, relative to the open leader at its cheapest host, per task and per token. II on the axes is the Artificial Analysis Intelligence Index.

Emberson, 2026). If the unmeasured attributes favor the closed models on net, the measured premium overstates the part that market power supports, and benchmark tuning adds to the overstatement. The premium is an upper bound on the closed providers' markup over their own serving cost only if that cost is at least what hosts charge for an open model of equal capability, which the data cannot check.

6.2 Where and when posted prices are cut

Figure 4 sets each provider's line against the open floor. Of the 87 versions launched in the sample, 71 percent entered at or above the floor. Each open release that beats the previous open leader raises the floor and pushes every lead down a step, so the curves descend as staircases, and 79 percent of version-weeks lie below the floor. The steps come from the floor rising, not from any change in price. The few price cuts (▼) sit near the floor, and retirements (×) are announced almost only below it.

Proposition 3 predicts that the rare cuts follow large improvements of the free alternative and fall on the versions nearest it. The cuts do cluster after frontier open releases, but the evidence rests on 7 decisions, three of them OpenAI's. The 8 provider cuts are 7 decisions, since OpenAI cut two posted prices on one day. The test asks whether a provider was more likely to cut a posted price soon after a frontier open release, one that reached or came within a point of the open leader. We compare the weekly rate of cut decisions in the eight weeks after each release with the eight weeks before, pooled over the 22 releases whose windows reach the panel. The window is our choice and not the model's. Releases come in clusters

Table 3: The closed premium over the free alternative at equal capability

Object	Estimate	Reading
A. Per token, hedonic premium (litellm, ECI); models no more than ten points below the open floor		
Median host: mean; 2024; 2025; 2026	1.43; 1.58; 1.45; 1.30	multiple 4.2
Median host: trend per year	−0.14 (0.28)	monthly −0.16 (0.26)
Cheapest listing: mean; 2024; 2025; 2026	1.95; 1.94; 2.02; 1.86	multiple 7.0
Cheapest listing: trend per year	−0.02 (0.29)	on Artificial Analysis host prices −0.74 (0.37)
B. Per token and per task (Artificial Analysis), within capture; 24 captures, 705 model-captures		
Per token; per task	1.79 (0.20); 1.40 (0.19)	multiples 6.0 and 4.0
Output tokens per index run, closed relative to open	−0.66 (0.13)	open models use 1.9 times as many

Notes: Log points, with standard errors clustered by model in parentheses. Trends are OLS slopes of the weekly premium on time in years, with standard errors from the covariance of the weekly premiums clustered by model; a line fitted through the weekly estimates as if they were data would give the median-host trend a standard error of 0.05. Panel A pools closed models and each open model's median independent host or cheapest listing (screened litellm prices) with ECI at least the open leader's minus ten; the regression has week effects, a common ECI slope (0.113), and a closed indicator for each week. Panel B uses the captures with token counts, published from April 2025, and within each, models of any developer no more than ten index points below the index's open leader (the four providers' closed models alone give a per-task multiple of 4.2); the cost per task is each model's input and output token counts for an index run at the provider's own input and output prices (closed) or the cheapest benchmarked host's (open). Table F.5 in Online Appendix F reports the premium by segment, by source, at the open leader, and at launch and over a model's life.

and their windows overlap, and 103 of the 121 weeks lie within eight weeks after some frontier release, so inference keeps the calendar of releases and redraws the date of each cut decision uniformly over the panel's weeks. The rate after a release exceeds the rate before by 0.042 decisions a week (two-sided $p = 0.096$). Counting the 8 cuts instead gives a difference of 0.060 a week ($p = 0.054$; Table 4), against a base rate of 0.066 cuts a week. The 11 major releases alone, the large improvements the model speaks to most directly, give a two-sided p of 0.281 (Table 4 and Figure 5), because for 2 of the cuts the nearest release was not a major one. It closed less than half a point on the best closed model.

The result rests on OpenAI's three decisions. Leaving out any one of them raises the two-sided p of the cut count from 0.054 to between 0.14 and 0.31, the largest without the cuts that followed Kimi K3. Leaving out any other decision keeps it at or below 0.058. OpenAI said that two of the three, o3's and GPT-5.6 Terra and Luna's, passed on lower serving costs of its own (Table F.10). Such statements rest on private information that outsiders cannot check and may be selective, so they can bias as well as inform (Milgrom, 1981). Online Appendix F gives the variants and shows that cuts do not precede releases.

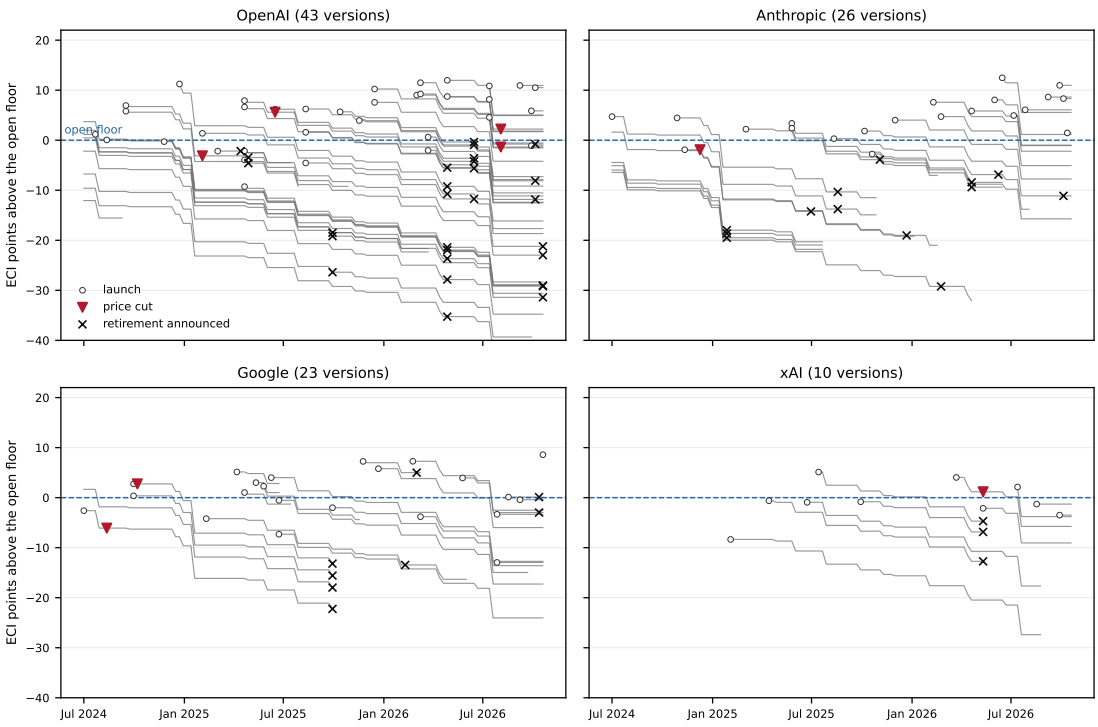

Figure 4: The product line against the open floor. Each curve is a closed version's lead over the open floor (ECI points, negative below it), weekly, from its first listing to its last, by provider. The dashed line is the open floor, and each open release that raises it moves every curve down a step. Circles (○): first listings of versions released in the sample; triangles (▼): provider price cuts; crosses (×): retirement announcements, each at the first week on or after the provider's date.

Three placebo event sets show no rise: open releases more than five points below the open floor (-0.008, $p = 0.863$), new closed frontier models, the releases of any developer that raise the closed frontier (-0.007, $p = 0.833$), and closed launches of any developer near the open floor (-0.008, $p = 0.67$).

Which price was cut. The model's sharpest restriction is on the order of cuts within a line. A version is not cut while a lower one keeps its price and its buyers. Whether the 8 cuts obey it depends on what counts as a line (Table F.9 in Online Appendix F). Take first all of a provider's versions as one line. Then 4 of the 8 cuts break the order, because a lower active version of the same provider kept its price. A lower version counts as active if it was listed that week, scored above every open model hosted at no more than its own price, and had no version of its own provider at least as capable and no dearer. This screen uses prices and capability, while the theory's condition is that the version keeps buyers. Take instead each tier as a line. The tiers are small, standard, and pro, read from the version's name

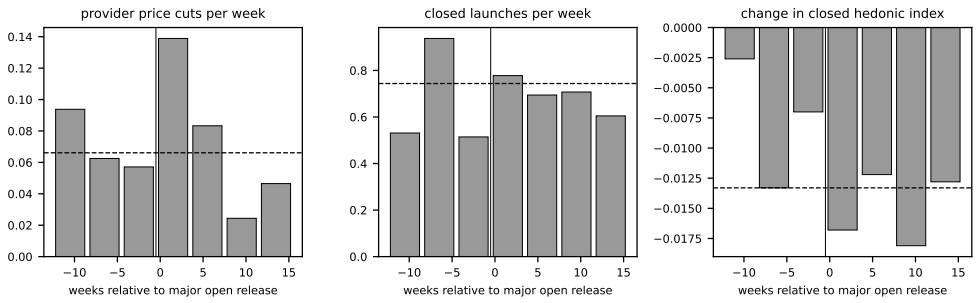

Figure 5: Event time around the 11 major open-weight releases (the timing test pools all 25 frontier releases): provider price cuts per week, closed launches per week, and the weekly change in the closed hedonic index, in four-week bins. Dashed lines: sample means.

and fixed before we looked at the cuts. Then 2 cuts break the order. In both, the version that kept its price belongs to another product family, a non-reasoning small model beside the reasoning o1-mini and a coding model beside GPT-5.6 Terra, a distinction we drew only after reading the cuts. One cut, the first in the sample, fell on a version that an open model already dominated in capability and price, which pays only if the version costs less to serve than the open model to host (Proposition B.5).

Where the cuts fall, how they scale with the size of the release, and which versions they hit are tested in Online Appendix F, with one primary test for each claim and p-values adjusted for the positions and definitions we examined (Table F.8).

Location. Of the 8 cuts, 3 fell on versions zero to five ECI points above the open floor, which account for 14 percent of version-weeks. Nearness to the floor is our stand-in for the bottom of a line. In the model-week panel, the weekly cut probability of those versions rises by 0.0092 (0.0054) from before to after a major release, against 0.0009 (0.0029) for versions below the floor and 0.0003 (0.0023) for versions far above it. The one-sided randomization p of the first is 0.034, and 0.10 and 0.16 after the adjustment of Romano and Wolf (2005) for the positions and contrasts we examined (Table 4), so the location of cuts is suggestive rather than established. Against the price-adjusted floor of Remark B.3, the best open model hosted at no more than the version's own price in the week, 5 of the 8 cuts fall zero to five points above it (one-sided $p = 0.092$), and three of the four cuts that were below the open floor were on versions still cheaper than every open model that had overtaken them (Online Appendix F).

Size of the release. The three releases that closed most of the gap to the best closed model, Llama 3.1 405B, DeepSeek-R1, and Kimi K3, were each followed within 20 days by

cuts of 20 to 80 percent. Across the frontier releases, the cuts that followed each rise with a proxy for the loss in Proposition 3 (one-sided permutation $p = 0.044$), a relation that rests on these three releases.

Age. The versions cut are young: the 8 cuts hit models a median of 1.6 months after their launch, against 8.0 months for the listed models, and a cut model was younger than 86 percent of the closed models listed in its week ($p = 0.0001$). The same holds within week and position. This fits the remaining-life mechanism of Proposition B.1, although that result is about the life left to a whole line and not to one version.

The provider's own frontier. The versions cut were close to their provider's best. In the week of the cut they sat a median of 3.5 ECI points below their provider's best listed version, against 12.3 for the closed model-weeks in the panel (permutation $p = 0.016$), and 3 of the 8 fell on the provider's best version, which in each case was also the lowest active version of its tier. The cuts did not fall on the versions that the provider's own newer models had left far behind. They fell on versions near the provider's frontier that the open floor had reached. The model predicts cuts there after an open improvement, and equally after a common fall in closed costs (Section 5).

6.3 When versions are retired

Proposition B.5 predicts that a version that an open model dominates in capability and price loses its buyers at its posted price and, if serving it costs at least the open model's hosted price, is withdrawn rather than repriced, so retirements should concentrate below the open floor. They do, but the data cannot tell whether the overtaking or the age brings the retirement (Online Appendix C).

The concentration itself is not in doubt. Of the 52 announcements, 50 concern versions below the floor, and an exact conditional test, which takes the number of announcements as given and asks how likely their split between the two sides of the floor is under equal rates, puts the ratio of the announcement rates below and above the floor at 8.1 (95 percent interval 2.1 to 69; $p < 0.001$). Every version retired below the floor was also dominated in price: an open model of at least equal ECI was hosted at no higher output price, the condition under which Proposition B.5 has the version lose all its demand at its posted price. The versions below the floor that no cheaper open model matched account for 6.3 model-years and were never retired ($p = 0.009$). In discrete-time hazards of a retirement announcement on 4,065 model-weeks of 100 closed models, with controls for age, the distance to the closed frontier, tier, provider, and a trend, being below the open floor raises the hazard by a factor of 5.1 (a

Table 4: Price cuts around open-weight releases

Object	Estimate	Reading
A. Provider cuts per week, weeks 0–7 after minus weeks 8–1 before (base rate 0.066)		
Frontier open releases (25)	0.060	one-sided 0.021; two-sided 0.054; by decisions 0.042, two-sided 0.096
Major open releases (11)	0.051	one-sided 0.159; two-sided 0.281
from Chinese developers (8)	0.054	one-sided 0.175
Placebo: open releases far below the open floor (33)	−0.008	two-sided 0.863
Placebo: new closed frontier models (27)	−0.007	two-sided 0.833
Distributed lag, frontier; major	0.061 (0.037); 0.070 (0.054)	binned leads and lags, trend, Newey–West
B. By position (model-weeks; post-minus-pre contrast in the weekly cut probability after major releases)		
Below the open floor; 0–5 above; 5 or more above	0.0009; 0.0092; 0.0003	(0.0029); (0.0054); (0.0023), week clusters
Near minus far, week fixed effects	0.0094 (0.0063)	
Near: randomization p; adjusted	0.034; 0.10; 0.16	one-sided; Romano–Wolf over three positions and over twelve contrasts
Cut decisions on models within three ECI points of a major release in the prior 60 days	2 of 7	13% of model-weeks exposed; one-sided $p = 0.24$

Notes: Provider cuts are the verified permanent cuts, dated from the providers' announcements and pricing pages (o1-mini's from the file's commit). Frontier releases are open-weight releases no more than one ECI point below the prior open leader's, January 2024 – October 2026; major ones raised the open floor by at least 1.5 points. Windows are clipped to the panel's weeks, so 22 frontier and 9 major releases contribute. The p-values of panel A keep the calendar of releases and redraw the date of each of the 7 cut decisions uniformly over the panel's weeks, 20,000 times; one decision cut two posted prices and carries both cuts. Panel B regressions include position, provider, and tier indicators, age, and a trend, and its randomization also keeps the release calendar and redraws the dates of the cut decisions. The exposure row counts decisions, with a one-sided binomial p.

coefficient of 1.63) with age entered linearly, while a version's distance to the provider's own best version does not predict its retirement (Table F.18 and Table F.19, panel C, in Online Appendix F).

The free alternative overtakes nearly every closed version early: 98 percent of the versions launched at or above the floor were overtaken within a year, at a median age of 4.6 months, and retirements are announced at a median age of 12 months. With an age spline, which fits better, or age bands in place of the linear term, the coefficient on being below the floor falls to 1.08 and 0.73 (standard errors 0.90 and 0.89), and within age bands the exact rate ratio is 1.7 (0.3 to 18; $p = 0.91$). Retirement also comes late. The 31 versions that an open release pushed below the floor before their retirement was announced waited a median of 34 weeks for the announcement. In the model a version without buyers stays listed that long

only if withdrawing it is costly (Proposition B.6).

6.4 Launches and the top of the line

Launches rarely reset legacy prices. The posted price of the predecessor, the provider's best version in the week before, was cut within a week of the launch in 1 of the 46 cases, which fall on 39 launch dates, so launches adjust prices through new versions. In 28 of the cases the predecessor's retirement was announced later, a median of 250 days after the launch.

Within a model, the list price of each provider's top version shows no detectable response to the free alternative, as Proposition 3 predicts when the improvements are small relative to the menu cost and the top's lead, although 3 of the cuts fell on a provider's best version that was also the lowest active version of its tier (Section 6.2). With model fixed effects, a point of open-leader ECI moves the top version's log price by 0.0023 (0.0018), and a point of the rivals' best closed model's ECI by 0.0018 (0.0022; Table F.18).

Launch prices respond no more to recent open gains next to the floor than well above it, whereas a launch into a sticky menu would respond more next to it (Proposition B.2). Given the prices of their predecessors, new tops respond to the closed frontier's gain over the ninety days before the launch, by -0.125 log points per point (0.053; permutation $p = 0.036$), and to the open leader's gain by -0.037 (0.039; $p = 0.37$), a response not distinguishable from zero. Replacement or a reset predicts a response to the open leader's gains and coexistence does not, but that response is not detectable here, and the model does not speak to the closed frontier. Since the predecessor's price alone does not reveal the regime either (Section 5), the launch regimes are identified only in part (Online Appendix F). The Opus pair of the introduction, the older and the newer version of one family, is a case that the order of prices settles. Opus 4.5 was more capable and cheaper per token, so if the cost per task ranks the same way, in the model Opus 4.1 was displaced at the launch, and the small share of tokens it kept until its withdrawal is the legacy demand of Remark B.4.

6.5 Do overtaken versions stay in use?

Usage moves with the versions (Section 4), but it does not all move to the free alternative. In the OpenRouter tables, 41 percent of the four providers' tokens, and 23 percent of their spending at list prices, went to versions that an open model matched or beat in capability while its median host charged no more. Weighted by tokens, the posted output price of those versions was a median of 2.4 times the open model's. The share is 33 percent without Google's Flash models and 17 percent when the open model must lead by five ECI points. In the model such a version loses all its buyers at its posted price (Proposition B.5). That

such versions stay in use points to attributes the capability index does not measure, or to costs of switching that the marketplace does not remove (Remark B.4). Online Appendix F gives the details and reports tests of the open share and of daily usage around open releases, which have little power.

7 Implications and Limitations

For providers and buyers. The closed providers cut the price of capability without, as a rule, cutting a price. A new version is the occasion to lower the schedule, and an old version is withdrawn rather than marked down. The premium is large (Section 6.1), and a model that leads every open model also commands a price for its lead. A closed version's lead over the free alternative is brief, since the free alternative overtakes 98 percent of the closed versions launched at or above it within a year, which fits a provider that lowers its schedule at the next launch rather than repricing a version whose lead is gone. For buyers the decline is on offer at each launch, and a buyer collects it only by moving. A buyer who stays on a named model pays last year's price for last year's capability. Contracts and budgets indexed to the current version capture the decline, and those tied to a model name do not. The comparison with the free alternative should be made per task rather than per token, because open models use about 1.9 times as many output tokens for the same work. A commitment to a named version also has an end. The retirements in the sample were announced at a median age of 12 months, with the shutdown set a median of 5 months after the notice, so a buyer should budget for a migration.

For policy. The free alternative appears to discipline the bottom of the closed product line, the versions it approaches or overtakes, and it makes the capability it matches cheaper for those who use it. In the data the list prices of the versions at the top show no detectable response to the free alternative or to the closed rivals, and the top is repriced at launches instead, mostly by new versions rather than by cuts of old ones. Policies that support open-weight models can therefore lower what users pay for capability below the frontier, a gain that the incumbents' price lists do not show. Whether they also lower the launch prices of new frontier versions the data cannot yet tell (Section 6.4).

What the next release should show. If the next frontier open release raises the open floor, the model predicts that any posted cuts will fall near the new floor, at the bottom of a tier's line. A cut on a version while a lower version of the same tier and product family keeps its price and its buyers would contradict the model, and so would the repricing of a

version that an open model dominates in capability and price and that costs at least the hosted price to serve, although the data cannot check that cost. A release followed by no cut at all would not, since the threshold in Proposition 3 rises with the lead of the versions nearest the floor.

Alternative explanations. Other forces could produce parts of the pattern (Table 5). Falling serving costs would bring cuts at any time, but a cost index built from the hosted prices of older open models does not move around frontier releases. That index cannot see a provider's own efficiency gains, which OpenAI cited for two of its cut decisions in statements that outsiders cannot check. Competition among the closed providers would time cuts to closed launches, but neither new closed frontier models nor closed launches near the open floor are followed by a rise in the cut rate, and the list price of a provider's top version responds to the rivals' best closed model no more than to the open leader. New tops do respond at launch to the closed frontier's recent gains, the provider's own included (Section 6.4). Fast technical progress with costly repricing alone would produce adjustment through versions, and we do not claim otherwise. A fall common to the closed providers' costs would also bring cuts at the bottom of the active line (Section 5), so the location of cuts cannot tell it from the free alternative, but it would not produce cuts that follow open releases rather than closed ones. Switching costs could also keep legacy prices up, since a provider can harvest installed buyers who would pay to move while it courts new ones with new versions (Klemperer, 1987; Farrell and Klemperer, 2007). Usage does move with the versions (Section 4), but the 41 percent of tokens that went to versions an open model dominates fit switching costs, or attributes the capability index does not measure, and the model treats switching costs only in Remark B.4. Retirement announcements fall on the versions a cheaper open model has overtaken, and a version's distance to its provider's own best version does not predict them, but age confounds that comparison. Unmeasured quality would make the premium a payment for service rather than a price of market power, but holding the measured attributes fixed does not shrink it (Section 6.1).

Limitations. List prices are not transacted prices. Enterprise discounts and committed-use contracts are unobserved. The posted batch discounts are fixed proportions of list prices, while the cached-input discounts deepen now and then, a second margin of repricing (Online Appendix F). Capability is a single index per model, built from public benchmarks, and several attributes remain unmeasured (Section 6.1). Hosted endpoints of the same model differ in latency, context, and reliability (Li et al., 2026), and two sources disagree on how the hosted price evolves after release, so whether the premium narrows depends on the source

Table 5: What separates the free alternative from technical progress and closed competition

Evidence	Estimate	Reading
Cuts follow frontier open releases, not new closed frontier models	post-minus-pre cut rate 0.060 a week after open releases against −0.007 after new closed frontier models; difference 0.067, one-sided $p = 0.045$	Suggestive, on 7 decisions. Closed competition does not appear to time the cuts
Cuts do not follow closed launches near the open floor	post-minus-pre cut rate −0.008 a week after 73 closed launches of any developer within five points of the open floor, against 0.060 after frontier open releases; difference 0.068, one-sided $p = 0.032$; frontier open releases minus other developers' launches, by provider, 0.018 (one-sided $p = 0.031$)	Suggestive, on 7 decisions. The closed events that compete with the versions cut do not appear to time the cuts either (Table F.9)
Cuts after major releases by Chinese developers	0.054 cuts a week, one-sided $p = 0.175$	Not distinguishable from chance. The releases least likely to be timed against US list prices
Bigger open steps bring more cuts	dose slope $p = 0.044$, on three releases	Suggestive, on few events
Hosted prices of older open models, and the providers' cuts on products other than language models, do not move with the releases	serving-cost control leaves the cut-count statistic at 0.063 (two-sided $p = 0.041$); non-language products −0.008 (two-sided $p = 0.43$)	Suggestive. A provider's own efficiency gains, which OpenAI cited for two decisions, would escape both
Retirements fall on versions a cheaper open model overtook, and the distance to the provider's own versions does not predict them	50 of 52 below the open floor, all price-dominated; undominated versions never retired ($p = 0.009$); own-frontier distance −0.025 (0.032)	Measured. Not separable from age

Notes: Each row is weak alone, and the rows share the same 8 cuts and 52 retirements. The timing p-values redraw the dates of the cut decisions and keep the event calendars, the dose slope permutes its proxy across releases, and the test on the providers' products other than language models redraws the release weeks. Together the rows point away from closed competition as the source of the cuts, and the timing and the dose point to the free alternative rather than to a common fall in costs. The location of the cuts and the retirements do not separate the two, since such a fall would also cut at the bottom of the line, and retirements cannot be separated from age. None of them identifies the free alternative's share of the decline in quality-adjusted prices.

(Section 6.1). Price cuts and major open releases are few, so the event-time evidence rests on randomization inference with modest power. The model takes the dates and qualities of new versions as given, so it explains how launches are priced and which versions are repriced or withdrawn, but not when new ones are launched. It has no contracts, and it treats switching costs only in a remark. The model predicts that an open improvement matters more for a version launched next to the free alternative, and we find no such difference in launch prices.

The data separate replacement and resets from coexistence only in part. The tests of launch prices were designed after the launches were split by position, they rest on 46 launches, and serving costs are unobserved. Release timing is not random, and a shock to the cost of serving frontier models alone would escape the serving-cost and non-language checks of Table 5. The usage data cover one marketplace of developers, the release tests have little power, and the closed models are those of four US developers, so closed models of Chinese and European developers are outside the sample.

8 Conclusion

A developer who looks up last year's model name finds last year's price. A developer who chooses again from the same provider's list finds more capability for less. Across the four closed providers, launches account for 54 percent of the decline in the price of capability, and the withdrawal of old, relatively expensive versions for 32 percent. A replacement thus has two dates, many months apart. Usage moves at the launch, and the old price stays on the list until the withdrawal. The free alternative has a price of its own, set more by a model's size than by what it can do, and it sits far below closed prices at equal capability. The gap is smaller per task than per token.

An improving free alternative gives a provider a reason to revisit its posted prices, and a product line with costly repricing says which. A cut, when it pays, falls on the lowest version that still keeps buyers, and the versions above it can be left alone. The few posted cuts fit that order only when lines are read narrowly, by tier and product family, and on 7 decisions they tend to follow frontier open releases. Retirements are announced at about a year of age, most of them after a cheaper open model has overtaken the version, and the data cannot separate the overtaking from the age. The data show where prices adjust. They do not show how much of the decline the free alternative causes.

As long as repricing stays costly, we expect the list prices of named models to stay where they are and the price of capability to keep falling through launches and withdrawals. Should repricing become cheap, through automated price schedules or contracts no longer tied to list prices, posted cuts would appear, starting at the bottom of the line and reaching the top last. The same margins are likely to operate wherever a free or open alternative improves over time, and the LLM market, with its posted prices and measured quality, is a laboratory in which to watch them.

References

Aghion, P., Bergeaud, A., Boppart, T., Klenow, P. J., and Li, H. (2019). Missing growth from creative destruction. *American Economic Review*, 109(8):2795–2822.

Aizcorbe, A. (2006). Why did semiconductor price indexes fall so fast in the 1990s? A decomposition. *Economic Inquiry*, 44(3):485–496.

Aizcorbe, A. and Kortum, S. (2005). Moore's law and the semiconductor industry: A vintage model. *Scandinavian Journal of Economics*, 107(4):603–630.

Álvarez, E. and Barry, A. (2026). Updating the Epoch Capabilities Index methodology to use a Bayesian model. Report, Epoch AI, 4 November 2026. epoch.ai/publications/eci-bayesian-methodology.

Alvarez, F. and Lippi, F. (2014). Price setting with menu cost for multiproduct firms. *Econometrica*, 82(1):89–135.

Anderson, E. T. and Simester, D. I. (2010). Price stickiness and customer antagonism. *Quarterly Journal of Economics*, 125(2):729–765.

Appenzeller, G. (2024). Welcome to LLMflation: LLM inference cost is going down fast. Andreessen Horowitz, November 2024.

Argente, D., Lee, M., and Moreira, S. (2018). Innovation and product reallocation in the US. *Journal of Monetary Economics*, 93:1–20.

Argente, D., Lee, M., and Moreira, S. (2024). The life cycle of products: Evidence and implications. *Journal of Political Economy*, 132(2):337–390.

Aubakirova, M., Atallah, A., Clark, C., Summerville, J., and Midha, A. (2026). State of AI: An empirical 100 trillion token study with OpenRouter. arXiv:2601.10088.

August, T., Chen, W., and Zhu, K. (2021). Competition among proprietary and open-source software firms: The role of licensing in strategic contribution. *Management Science*, 67(5):3041–3066.

August, T., Shin, H., and Tunca, T. I. (2018). Generating value through open source: Software service market regulation and licensing policy. *Information Systems Research*, 29(1):186–205.

Bergemann, D., Bonatti, A., and Smolin, A. (2026). Menu pricing of large language models. Cowles Foundation Discussion Paper 2502, Yale University.

Bhargava, H. K. and Choudhary, V. (2008). Research note—When is versioning optimal for information goods? *Management Science*, 54(5):1029–1035.

Bils, M. (2009). Do higher prices for new goods reflect quality growth or inflation? *Quarterly Journal of Economics*, 124(2):637–675.

Bulow, J. (1986). An economic theory of planned obsolescence. *Quarterly Journal of Economics*, 101(4):729–749.

Byrne, D. M., Oliner, S. D., and Sichel, D. E. (2018). How fast are semiconductor prices falling? *Review of Income and Wealth*, 64(3):679–702.

Casadesus-Masanell, R. and Ghemawat, P. (2006). Dynamic mixed duopoly: A model motivated by Linux vs. Windows. *Management Science*, 52(7):1072–1084.

Casadesus-Masanell, R. and Zhu, F. (2010). Strategies to fight ad-sponsored rivals. *Management Science*, 56(9):1484–1499.

Caves, R. E., Whinston, M. D., and Hurwitz, M. A. (1991). Patent expiration, entry, and competition in the U.S. pharmaceutical industry. *Brookings Papers on Economic Activity: Microeconomics*, 1991:1–66.

Chellappa, R. K. and Mehra, A. (2018). Cost drivers of versioning: Pricing and product line strategies for information goods. *Management Science*, 64(5):2164–2180.

Copeland, A., Dunn, W., and Hall, G. (2011). Inventories and the automobile market. *RAND Journal of Economics*, 42(1):121–149.

Demirer, M., Fradkin, A., and Tadelis, N. (2026). The emerging market for intelligence: How firms buy and sell AI. *Journal of Economic Perspectives*, 40(3):23–46.

Deneckere, R. J. and McAfee, R. P. (1996). Damaged goods. *Journal of Economics & Management Strategy*, 5(2):149–174.

Economides, N. and Katsamakas, E. (2006). Two-sided competition of proprietary vs. open source technology platforms and the implications for the software industry. *Management Science*, 52(7):1057–1071.

Edwards, J. and Emberson, L. (2026). Open models lag state-of-the-art closed models by 4 months. Data insight, Epoch AI, May 2026.

Ellison, G. and Fudenberg, D. (2000). The neo-Luddite's lament: Excessive upgrades in the software industry. *RAND Journal of Economics*, 31(2):253–272.

Epoch AI (2025). LLM inference prices have fallen rapidly but unequally across tasks. Data insight, Epoch AI, March 2025.

Farrell, J. and Klemperer, P. (2007). Coordination and lock-in: Competition with switching costs and network effects. In Armstrong, M. and Porter, R. H., editors, *Handbook of Industrial Organization*, volume 3, pages 1967–2072. Elsevier, Amsterdam.

Flamm, K. (2018). Measuring Moore's law: Evidence from price, cost, and quality indexes. NBER Working Paper 24553.

Fradkin, A. (2025). Demand for LLMs: Descriptive evidence on substitution, market expansion, and multihoming. arXiv:2504.15440.

Frank, R. G. and Salkever, D. S. (1992). Pricing, patent loss and the market for pharmaceuticals. *Southern Economic Journal*, 59(2):165–179.

Frank, R. G. and Salkever, D. S. (1997). Generic entry and the pricing of pharmaceuticals. *Journal of Economics & Management Strategy*, 6(1):75–90.

Goettler, R. L. and Gordon, B. R. (2011). Does AMD spur Intel to innovate more? *Journal of Political Economy*, 119(6):1141–1200.

Grabowski, H. G. and Vernon, J. M. (1992). Brand loyalty, entry, and price competition in pharmaceuticals after the 1984 Drug Act. *Journal of Law and Economics*, 35(2):331–350.

Gundlach, H., Lynch, J., Mertens, M., and Thompson, N. (2025). The price of progress: Price performance and the future of AI. arXiv:2511.23455, revised March 2026.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.

Johnson, J. P. and Myatt, D. P. (2003). Multiproduct quality competition: Fighting brands and product line pruning. *American Economic Review*, 93(3):748–774.

Johnson, J. P. and Myatt, D. P. (2018). The determinants of product lines. *RAND Journal of Economics*, 49(3):541–573.

Jones, R. and Mendelson, H. (2011). Information goods vs. industrial goods: Cost structure and competition. *Management Science*, 57(1):164–176.

Jullien, B. (2000). Participation constraints in adverse selection models. *Journal of Economic Theory*, 93(1):1–47.

Klemperer, P. (1987). Markets with consumer switching costs. *Quarterly Journal of Economics*, 102(2):375–394.

Korinek, A. and Vipra, J. (2025). Concentrating intelligence: Scaling and market structure in artificial intelligence. *Economic Policy*, 40(121):225–256.

Kumar, V., Gordon, B. R., and Srinivasan, K. (2011). Competitive strategy for open source software. *Marketing Science*, 30(6):1066–1078.

Lambrecht, A. and Misra, K. (2017). Fee or free: When should firms charge for online content? *Management Science*, 63(4):1150–1165.

Lerner, J. and Tirole, J. (2002). Some simple economics of open source. *Journal of Industrial Economics*, 50(2):197–234.

Li, H., He, Z., Liu, X., Xu, Y., Liu, D., Ma, J., Wu, L., Wu, Y., Tang, X., and Shi, T. (2026). When is the same model not the same service? A measurement study of hosted open-weight LLM APIs. arXiv:2605.02821.

Midrigan, V. (2011). Menu costs, multiproduct firms, and aggregate fluctuations. *Econometrica*, 79(4):1139–1180.

Milgrom, P. R. (1981). Good news and bad news: Representation theorems and applications. *Bell Journal of Economics*, 12(2):380–391.

Milgrom, P. and Shannon, C. (1994). Monotone comparative statics. *Econometrica*, 62(1):157–180.

Mussa, M. and Rosen, S. (1978). Monopoly and product quality. *Journal of Economic Theory*, 18(2):301–317.

Nakamura, E. and Steinsson, J. (2008). Five facts about prices: A reevaluation of menu cost models. *Quarterly Journal of Economics*, 123(4):1415–1464.

Nordhaus, W. D. (2007). Two centuries of productivity growth in computing. *Journal of Economic History*, 67(1):128–159.

OpenRouter (2026). LLM rankings: token volumes by model. openrouter.ai/rankings, captures in the Internet Archive from January 2024 to October 2026.

Pakes, A. (2003). A reconsideration of hedonic price indexes with an application to PC's. *American Economic Review*, 93(5):1578–1596.

Regan, T. L. (2008). Generic entry, price competition, and market segmentation in the prescription drug market. *International Journal of Industrial Organization*, 26(4):930–948.

Romano, J. P. and Wolf, M. (2005). Stepwise multiple testing as formalized data snooping. *Econometrica*, 73(4):1237–1282.

Schmidheiny, K. and Siegloch, S. (2023). On event studies and distributed-lags in two-way fixed effects models: Identification, equivalence, and generalization. *Journal of Applied Econometrics*, 38(5):695–713.

Shaked, A. and Sutton, J. (1982). Relaxing price competition through product differentiation. *Review of Economic Studies*, 49(1):3–13.

Shapiro, C. and Varian, H. R. (1998). Versioning: The smart way to sell information. *Harvard Business Review*, 76(6):106–114.

Sheshinski, E. and Weiss, Y. (1977). Inflation and costs of price adjustment. *Review of Economic Studies*, 44(2):287–303.

Wang, Z. (2026). Who captures AI deflation? Quality-adjusted prices, markups, and pass-through in AI services. SSRN Working Paper 6426025, Warwick Business School, November 2026 version.

Xin, M. and Sundararajan, A. (2020). Nonlinear pricing of software with local demand inelasticity. *Information Systems Research*, 31(4):1224–1239.

Zbaracki, M. J., Ritson, M., Levy, D., Dutta, S., and Bergen, M. (2004). Managerial and customer costs of price adjustment: Direct evidence from industrial markets. *Review of Economics and Statistics*, 86(2):514–533.

Zhou, Z. Z. and Choudhary, V. (2022). Impact of competition from open source software on proprietary software. *Production and Operations Management*, 31(2):731–742.

Zhu, L. Y. (2026). The price of intelligence: A quality-adjusted price index for AI services. arXiv:2608.29843, version 2, 2 November 2026.

A Proofs of the Main Results

Every proposition has also been checked numerically, computing demand from the buyers' choices rather than from the formulas below. Online Appendix B describes these checks and states and proves the further results of the model.

Proof of Lemma 1. If the menu is active, version j is bought by types in $[\theta_j, \theta_{j+1})$, with $\theta_{J+1} = \bar{\theta}$. Write $p_j = c_o + \sum_{k \le j} \theta_k \Delta q_k$ and $c_j = c_o + \sum_{k \le j} \Delta c_k$. Revenue is $\sum_j p_j[F(\theta_{j+1}) - F(\theta_j)] = c_o(1 - F(\theta_1)) + \sum_k \theta_k \Delta q_k (1 - F(\theta_k))$, since the increment $\theta_k \Delta q_k$ is paid by every buyer above θ_k. Cost is $c_o(1 - F(\theta_1)) + \sum_k \Delta c_k (1 - F(\theta_k))$ by the same argument. The difference is (1).

Proof of Proposition 1. Each term $T_k(\theta) = (1 - F(\theta))(\theta \Delta q_k - \Delta c_k)$ has derivative $T_k'(\theta) = -f(\theta)[(\theta - h(\theta))\Delta q_k - \Delta c_k]$. Under Assumption 1, $\theta - h(\theta)$ is strictly increasing, so T_k is single-peaked with its maximum where $(\theta_k - h(\theta_k))\Delta q_k = \Delta c_k$. Under Assumption 2 the maximizers lie strictly inside $(0, \bar{\theta})$ and increase with k, so the menu they define is active. No menu in which a version has no buyers does better, since its profit is the sum of the same terms at weakly increasing cutoffs (proof of Proposition 3(b)), so the cutoffs are optimal. The price of the lowest version follows from the definition of θ_1.

Proof of Proposition 2. Only θ_1 and Δq_1 depend on q_o, so $\partial p_j / \partial q_o = -\theta_1 + \Delta q_1 \, \partial \theta_1 / \partial q_o$ for every j. Differentiating the first-order condition, $(1 - h'(\theta_1))\Delta q_1 \, \partial \theta_1 / \partial q_o = \theta_1 - h(\theta_1)$. Substituting gives (2). Its numerator is negative because $h(\theta_1) > 0$ for $\theta_1 < \bar{\theta}$ and $h' \le 0$, and its denominator is at least one. For the hosting cost, $\partial p_j / \partial c_o = 1 + \Delta q_1 \, \partial \theta_1 / \partial c_o$ and $(1 - h'(\theta_1))\Delta q_1 \, \partial \theta_1 / \partial c_o = -1$, so $\partial p_j / \partial c_o = -h'(\theta_1)/(1 - h'(\theta_1))$. Remark B.1 follows from the sign of the numerator of (2) when the denominator is positive. With uniform types, $p_1 = c_o + \frac{1}{2}(\Delta q_1 + \Delta c_1) = \frac{1}{2}(q_1 - q_o + c_1 + c_o)$ and the higher cutoffs do not involve (q_o, c_o),

so the rates of one half hold exactly while Assumption 2 holds. In the pruning example of Online Appendix B, $\theta_1 = \frac{1}{2}(1 + 0.1/(1 - q_o))$ reaches $\theta_2 = 0.7$ at $q_o = 0.75$.

Proof of Proposition 3. After the improvement, write $\omega_1 = \ell_1$ and $\omega_k = \Delta q_k$ for $k \geq 2$, the quality steps between neighboring options. With uniform types each term of (1), $T_k(\theta) = (1 - \theta)(\theta\omega_k - \Delta c_k)$, is a quadratic with leading coefficient $-\omega_k$ and maximum at $\theta_k^* = \frac{1}{2}(1 + \Delta c_k/\omega_k)$, so $T_k(\theta_k^*) - T_k(\theta) = \omega_k(\theta - \theta_k^*)^2$ exactly. The new optimal prices are the old ones less $\Delta/2$ (Proposition 2), so at the prices $p_j - d_j$ the cutoffs are $\theta_1 = \theta_1^* + (d_0 - d_1)/\ell_1$ and $\theta_k = \theta_k^* + (d_{k-1} - d_k)/\Delta q_k$ for $k \geq 2$, and Lemma 1 gives (3) as long as the menu stays active. At unchanged prices the loss is $\Delta^2/(4\ell_1)$, which falls with ℓ_1. Suppose the provider changes the prices of a run of versions $j, \ldots, j'$ with $j \geq 2$ while version $j - 1$ and, if $j' < J$, version $j' + 1$ keep theirs. The run's cuts enter (3) only through the terms $k = j, \ldots, j' + 1$, whose sum is zero when the cuts are zero, since $d_{j-1} = d_{j'+1} = 0$. The run recovers nothing and costs its menu costs. The versions cut therefore form a bottom block or there are none. For a block $k < J$, write $e_i = d_{i-1} - d_i$ for $i \leq k + 1$. The e_i sum to $d_0 - d_{k+1} = \Delta/2$, and minimizing $\sum_i e_i^2/\omega_i$ subject to that sum gives $e_i = (\Delta/2)\omega_i/\ell_{k+1}$, since $\sum_{i \leq k+1} \omega_i = \ell_{k+1}$. Hence $d_i = (\Delta/2)(1 - \ell_i/\ell_{k+1})$ and $R_k = \Delta^2/(4\ell_{k+1})$. For $k = J$, cuts of $\Delta/2$ make every term zero. In every block solution the cutoffs $1, \ldots, k+1$ rise by the same amount, $\Delta/(2\ell_{k+1}) \leq \Delta/(2\ell_1)$, above their new optima and the others stay at theirs, so the menu stays active at every price considered if $\theta_1^* > 0$, $\theta_k^* + \Delta/(2\ell_1) < \theta_{k+1}^*$ for $k < J$, and $\theta_J^* + \Delta/(2\ell_1) < 1$. This is the condition on the size of the improvement.

For part (c), if version k belongs to the optimal block $\{1, \ldots, k^*\}$, that block costs no more than $\{1, \ldots, k - 1\}$, so $k^*\kappa + R_{k^*} \leq (k - 1)\kappa + R_{k-1}$, and hence $\kappa \leq (k^* - k + 1)\kappa \leq R_{k-1} = \Delta^2/(4\ell_k)$. The whole line is optimal if $(J - k)\kappa \leq R_k$ for every $k < J$. Since $R_k \geq \Delta^2/(4\ell_J)$ and $J - k \leq J$, the condition $J\kappa \leq \Delta^2/(4\ell_J)$ suffices, and the comparison with $k = J - 1$ shows that $\kappa \leq \Delta^2/(4\ell_J)$ is necessary. Write $\bar{\ell}_k = q_k - q_o = \ell_k + \Delta$ for the lead before the improvement. The objective $k\kappa + R_k$ falls from block k to block $k + 1$ by $R_k - R_{k+1} - \kappa$, where $R_k - R_{k+1} = \frac{\Delta^2}{4} \frac{\bar{\ell}_{k+2} - \bar{\ell}_{k+1}}{(\bar{\ell}_{k+1} - \Delta)(\bar{\ell}_{k+2} - \Delta)}$ for $k + 1 < J$ and $R_{J-1} - R_J = \Delta^2/(4(\bar{\ell}_J - \Delta))$. Both rise with Δ, and the fall decreases with κ. The objective therefore has decreasing differences in the block size and the improvement and increasing differences in the block size and the menu cost, so its smallest and largest minimizers rise weakly with Δ and fall weakly with κ (Milgrom and Shannon, 1994). For part (d), some block beats leaving all prices unchanged if and only if $R_0 - R_k > k\kappa$ for some $k \geq 1$. Each gain $R_0 - R_k$ is zero at $\Delta = 0$ and rises with Δ, so there is a threshold Δ^* below which no price changes. Holding the steps $q_{k+1} - q_1$ fixed, each gain, $\frac{\Delta^2}{4}\left(\frac{1}{\bar{\ell}_1 - \Delta} - \frac{1}{\bar{\ell}_1 + q_{k+1} - q_1 - \Delta}\right)$ for $k < J$ and $\frac{\Delta^2}{4(\bar{\ell}_1 - \Delta)}$ for $k = J$, falls with $\bar{\ell}_1$, so Δ^* falls as the lead closes. With a single version, the cut pays if and only if $\kappa \leq \Delta^2/(4(\bar{\ell}_1 - \Delta))$,

and the positive root of $\Delta^2 + 4\kappa\Delta - 4\kappa\bar{\ell}_1 = 0$ is $\Delta^* = 2(\sqrt{\kappa(\kappa + \bar{\ell}_1)} - \kappa)$.

For the statement before the proposition, consider two versions. Cutting version 2 by $\Delta/2$ prices it at its stand-alone price against the improved free alternative, by the argument of Proposition B.3(b), and leaves version 1 without buyers once version 1's stale cutoff $(p_1 - c_o)/\ell_1$ reaches version 2's stand-alone cutoff, as in Proposition B.3(c). This cut then costs $\kappa + B$, with B of Proposition B.6 for $\ell = \ell_1$, $v = \Delta q_2$, $a = \Delta c_1$, and $r = \Delta c_2/\Delta q_2$, against R_0 without a cut, $\kappa + R_1$ for a cut on version 1 alone, and 2κ for cuts on both, so cutting version 2 alone is best if $B < \kappa$, $B < R_1$, and $\kappa + B < R_0$. With qualities 1 and 1.5, costs 0.2 and 0.5, $q_o = c_o = 0$, and $\Delta = 0.24$, which meet the condition above, $B \approx 0.0086$, $R_1 \approx 0.0114$, and $R_0 \approx 0.0189$, so the provider cuts only version 2 for menu costs between about 0.0086 and 0.0104. The numerical checks search over all cuts on random menus with two to four versions, including menus in which such a cut is best.

For the first statement of part (b), let F satisfy Assumption 1, extended by $F = 0$ below 0 and $F = 1$ above $\bar{\theta}$. The legacy prices were optimal before the improvement, so $p_k - p_{k-1} = \theta_k^* \Delta q_k$ for $k \geq 2$, with $\theta_2^* < \cdots < \theta_J^*$ (Proposition 1). A buyer of type θ takes the option that minimizes $p - \theta q$. Plot each option as the point (q, p), the improved free alternative at $(q_o + \Delta, c_o)$ included. The options with buyers are then vertices of the lower convex hull of these points, and a vertex is bought by the types between the slopes of the hull edges on either side of it. Give each version j the slope $\hat{\theta}_j$ of the hull edge that ends at it from the left or passes below it. An edge that skips versions adds up their steps in quality and cost, so the profit of every menu is $\sum_j T_j(\hat{\theta}_j)$, with T_j the terms of Lemma 1 after the improvement, each single-peaked with its maximum at its optimal cutoff. First, an optimal set of cuts includes no version without buyers, since restoring that version's old price raises its point, changes no buyer's choice, and saves κ. Second, let u be the lowest uncut version with buyers, and suppose that some version above it is cut. Restore the old price of every cut version above u. Only points to the right of u move, and they move up. A line through u's point whose slope σ lies inside u's range of buyers lay below every point and still does, so u remains a vertex with buyers, the hull to its left is unchanged, and so is every term $T_j(\hat{\theta}_j)$ with $j \leq u$. The same line lies below the restored point of version $u+1$, so $\sigma \leq \theta_{u+1}^*$, and the versions above u, all at their old prices, join the hull as a convex chain with slopes $\theta_{u+1}^* < \cdots < \theta_J^*$. Every term with $j > u$ is then at its maximum. Profit does not fall and at least one menu cost is saved, so the original cuts were not optimal. Hence every version cut has buyers and lies below u, and every version with buyers below u is cut, by the definition of u. The two-version example fits this, with version 1 left without buyers and version 2, the lowest version that keeps them, cut.

Proof of Proposition 4. Part (a) restates Proposition B.3(a)–(c) with P_S written as in its part (b), and part (b) restates Proposition B.6(b) to (d), with the monotonicity of L and B from its part (a).

Online Appendix to

Repricing by Replacement:

Pricing Closed Language Models Against a Free Alternative

Martin Ljunge

This online appendix contains a worked example of the model, its further results, and their proofs (Appendix B), the detail behind several passages of the text (Appendix C), the construction of the data (Appendix D), the two sources for the price of the free alternative (Appendix E), and the robustness of every estimate (Appendix F). Tables, figures, and equations carry the letter of their appendix, and Appendix A of the paper proves Lemma 1 and Propositions 1 to 3 of its text, while Proposition 4 collects results proved here.

B Theory: A Worked Example, Further Results, and Proofs

B.1 A worked example

This example takes the model of Section 5 through its steps with numbers, and Figure 2 in the text draws it. Types are uniform on $[0, 1]$. The free alternative has quality $q_o = 1$ and costs $c_o = 0.1$ per task. The provider sells three versions of qualities 2, 3, and 6 at marginal costs 0.1, 0.4, and 2.2. The steps are $\Delta q = (1, 1, 3)$ and $\Delta c = (0, 0.3, 1.8)$, so the cost ratios $\Delta c_k/\Delta q_k$ of 0, 0.3, and 0.6 rise as Assumption 2 requires. Table B.1 collects the numbers.

The optimal line. By Proposition 1 the optimal cutoffs are $\theta_k = \frac{1}{2}(1 + \Delta c_k/\Delta q_k)$, or 0.5, 0.65, and 0.8. The prices climb the ladder $p_j = c_o + \sum_{k \le j} \theta_k \Delta q_k$ to 0.6, 1.25, and 3.65, so the lowest price exceeds the free alternative's by $\theta_1(q_1 - q_o) = 0.5$, the quality gap valued at the marginal buyer's type. Half of the buyers use the free alternative, and the three versions sell to 0.15, 0.15, and 0.2 of them. Profit is 0.4925, which Lemma 1 splits into one term per cutoff, 0.25, 0.1225, and 0.12. Only the first term, $(1-\theta_1)(\theta_1 \Delta q_1 - \Delta c_1) = 0.5 \times 0.5$, involves the free alternative, and every closed buyer contributes to it, not only the buyers of version

1.

An improvement of the free alternative. Let the free alternative's quality rise by $\Delta = 0.2$, to 1.2. Repriced optimally, the line keeps its cutoffs, because $\Delta c_1 = 0$ holds θ_1 at 0.5 and the higher cutoffs do not involve the free alternative. Every price falls by $\Delta/2 = 0.1$, to 0.5, 1.15, and 3.55 (Proposition 2), and profit falls by 0.05, the cut times the half of the buyers who buy a closed version, to 0.4425. At the old prices only the lowest cutoff moves. The lowest version is now bought from $\theta_1 = (0.6 - 0.1)/0.8 = 0.625$, so its buyers with types between 0.5 and 0.625 switch to the free alternative, and no buyer of version 2 or 3 moves. Profit is then 0.43, and the loss against the repriced line is $0.0125 = \Delta^2/(4\ell_1)$, where $\ell_1 = 0.8$ is the lowest version's lead over the improved free alternative, as in Proposition 3(a).

Where the cuts go. The leads over the improved free alternative are $\ell = (0.8, 1.8, 4.8)$. By Proposition 3(b), cutting version 1 alone by $(\Delta/2)(1 - \ell_1/\ell_2) \approx 0.056$ leaves the loss $R_1 = \Delta^2/(4\ell_2) \approx 0.0056$. Cutting the bottom two versions, by about 0.083 and 0.063, leaves $R_2 \approx 0.0021$, and cutting all three by 0.1 leaves none. In each block solution the cutoffs $\theta_1, \ldots, \theta_{k+1}$ sit above their new optima by the same amount, $\Delta/(2\ell_{k+1})$, which is 0.125 with no cut, 0.056 with one, and 0.021 with two, and the loss is the lead ℓ_{k+1} times the square of that gap. The provider chooses the block that minimizes $k\kappa + R_k$. A cut on version 1 pays once the menu cost is below $R_0 - R_1 \approx 0.0069$, adding version 2 pays below $R_1 - R_2 \approx 0.0035$, and adding version 3 pays below $R_2 \approx 0.0021$, so the block grows from the bottom of the line as the menu cost falls (Panel B of Table B.1). Figure B.1 draws the three steps: the loss from the stale price, the cuts in each block, and the block chosen at each improvement and menu cost. In its panel (c), following a horizontal line at a given menu cost from left to right shows the block growing from the bottom of the line as the improvement accumulates, which is how several open releases left unanswered build the case for a cut. The necessary condition of part (c) is not sufficient. At $\kappa = 0.005$ version 2 meets it, since $\ell_2 = 1.8 \leq \Delta^2/(4\kappa) = 2$, but only version 1 is cut.

Cuts that leave a version without buyers. In this example the blocks are also the best of all cuts. Cutting version 2 alone by 0.0625 leaves version 1 without buyers at its

stale price, a replacement within the line, and loses about 0.0121 before its menu cost. That beats leaving every price unchanged only for menu costs below about 0.0004, where cutting all three prices costs less.

Launches. The launch regimes of Proposition 4 involve one legacy version and a new one, and the example after Proposition B.3 works through them, which Figure B.2 draws. Its legacy version has quality 1 and cost 0.2 and was priced at 0.6 against a free alternative of quality 0 and cost 0, and the new version has quality 1.5 and cost 0.4. After small improvements the new version is priced against the legacy one at 0.95, and both sell. From an improvement of about 0.096 the provider replaces the legacy version instead, and the launch price drops to the new version's stand-alone price, half the improvement below 0.95. After an improvement of 0.12 and over one period, with no listing or exit cost, a reset of the legacy price that costs 0.001 beats replacement, and one that costs 0.003 does not (the example after Proposition B.6). Panel (c) of Figure B.2 shows why a reset is chosen only for moderate improvements: the loss from a stale price starts at zero and grows with the square of the improvement, while the legacy version's value to the line shrinks as the free alternative closes in, so a reset pays only in the range where its cost lies below both.

B.2 Further results

This appendix states the results that Section 5 summarizes or sets aside: the pruning of a version before it is overtaken, the generic-entry paradox, a closed rival, several open models, remaining life, launches into a sticky menu, replacement, launches as repricing opportunities, withdrawal, the joint choice among coexistence, replacement, and a reset, and the local counterpart of Proposition 3 for general type distributions. Proposition 4 in the text collects parts (a) to (c) of Proposition B.3 and parts (a) to (d) of Proposition B.6.

Pruning. An improvement can also leave a version without buyers before it overtakes it. If $\Delta c_1 > 0$, the ratio $\Delta c_1/\Delta q_1$ rises as the gap closes, and once it reaches $\Delta c_2/\Delta q_2$ Assumption 2 fails: the optimal cutoffs θ_1 and θ_2 meet, version 1 has no buyers in the optimal menu, and the rest of the line is priced against the free alternative with version 2 at the bottom. With

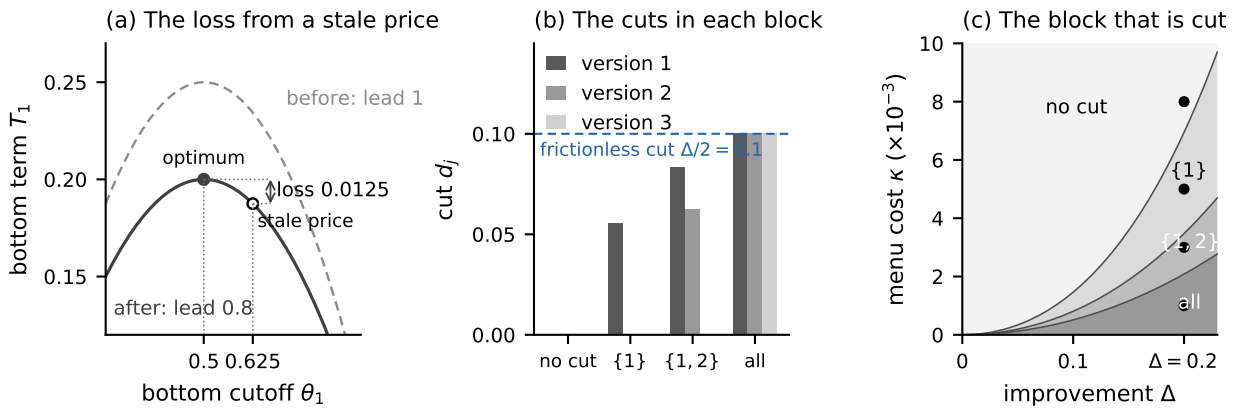

Figure B.1: Repricing after the improvement in the worked example. (a) The bottom term of Lemma 1, $T_1 = (1-\theta_1)\theta_1\ell_1$, before and after the free alternative's quality rises by 0.2. The improvement lowers the term and leaves its peak at 0.5. At the stale price the bottom cutoff sits at 0.625, and the loss is the lead times the squared miss, $0.8 \times 0.125^2 = 0.0125$ (Proposition 3(a)). (b) The cuts on each version in the four bottom blocks of Proposition 3(b). Within a partial block they shrink up the line, and when the whole line is cut every price falls by $\Delta/2$. (c) The block the provider cuts at each improvement and menu cost. The boundaries are the menu costs at which the next version joins the block, $\kappa = R_0 - R_1$, $R_1 - R_2$, and R_2. The dots are menu costs of 0.001, 0.003, 0.005, and 0.008 at the improvement of 0.2: the whole line, versions 1 and 2, version 1, and no cut (Panel B of Table B.1). The formulas hold while every version keeps buyers, up to an improvement of about 0.23.

uniform types, $q_1 = 1$, $q_2 = 2$, $c_o = 0$, $c_1 = 0.1$, and $c_2 = 0.5$, the optimal cutoffs start at 0.55 and 0.7 and meet when q_o reaches 0.75, while version 1 is still better than the free alternative.

Remark B.1 (When the price rises). Without Assumption 1, at an optimum with $1-h'(\theta_1) > 0$, prices rise when the free alternative improves if and only if $h'(\theta_1) > h(\theta_1)/\theta_1$, a hazard rate that falls steeply at the marginal buyer. This is the local counterpart of the generic-entry paradox, in which the buyers who leave for the free alternative are more price sensitive than those who stay. It is a condition on the distribution of types at the margin rather than a model of segmented demand. Under Assumption 1 the paradox does not arise.

Remark B.2 (A closed rival at a fixed price). Suppose a closed rival sells quality q_R between the provider's versions K and $K+1$ at a fixed price p_R, and the provider's best response leaves every option with buyers. The buyers of versions $K+1, \ldots, J$ compare them with the rival, not with the free alternative, so the profit from those versions separates as in

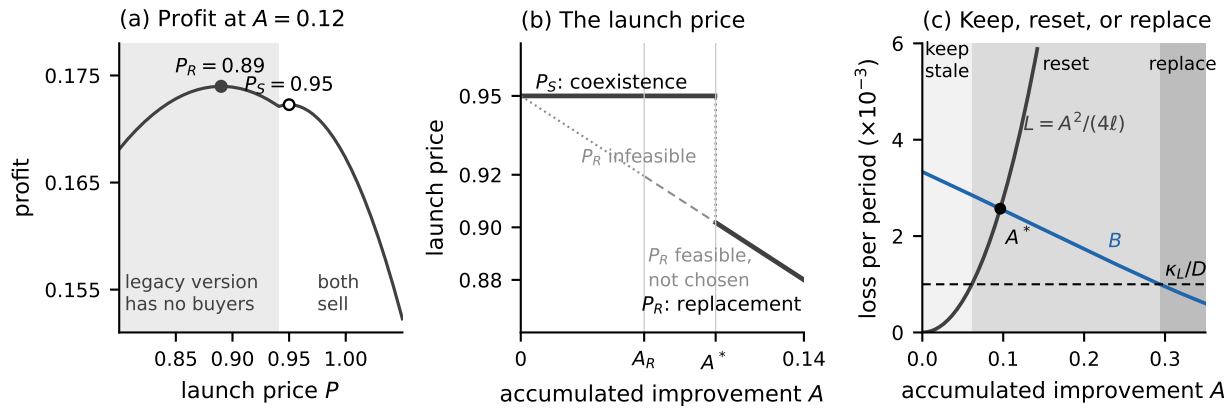

Figure B.2: The launch in the example after Proposition B.3: a legacy version of quality 1 and cost 0.2, priced at 0.6 when the free alternative had quality 0 and cost 0, and a new version of quality 1.5 and cost 0.4. (a) Profit against the launch price after the free alternative has improved by $A = 0.12$, with the legacy price in place. At low launch prices (shaded) the legacy version has no buyers. Each branch has its own peak, the replacement price $P_R = 0.89$, which is chosen, and the coexistence price $P_S = 0.95$. (b) The launch price against the accumulated improvement. Under coexistence it stays at 0.95, because the legacy price fixes the free alternative that the new price refers to. Replacement is infeasible below $A_R \approx 0.061$, feasible but not chosen up to $A^* \approx 0.096$, and chosen beyond, where the launch price drops by $A^*/2$ and then falls by half of every further improvement. (c) The per-period losses of Proposition B.6: L, from keeping the legacy price stale, rises with the improvement, while B, the legacy version's value to the line, falls, and they cross at A^*. With a reset cost of $\kappa_L/D = 0.001$ and no listing or exit cost, the provider keeps the stale price for improvements below about 0.061, resets it between about 0.061 and 0.294, and replaces the legacy version beyond.

Lemma 1 with the rival in place of the free alternative: their cutoffs solve the conditions of Proposition 1, the first as $(\theta - h(\theta))(q_{K+1} - q_R) = c_{K+1} - p_R$, and their prices do not depend on (q_o, c_o). The free alternative moves only the prices of versions $1, \ldots, K$, which are set between it and the rival. With uniform types, versions of qualities 1 and 4 and costs 0.1 and 2, $c_o = 0$, and a rival of quality 2 at price 1, the top price stays at 2.5 when q_o rises from 0 to 0.1, while the lower price falls from 0.3 to about 0.287. A closed rival between the free alternative and the top thus insulates the top in level, not only in proportion. This is a best response to a given rival price, not an equilibrium among closed providers, whose prices would respond to one another.

Remark B.3 (Several open models and non-use). With several open models i of qualities q_i and hosted prices c_i, and the option of not using a model, a buyer's outside option is

Table B.1: The worked example

Panel A. The line	Free alternative	Version 1	Version 2	Version 3
Quality	1	2	3	6
Cost per task	0.1	0.1	0.4	2.2
Optimal cutoff θ_k		0.5	0.65	0.8
Optimal price	0.1	0.6	1.25	3.65
Share of buyers	0.5	0.15	0.15	0.2
After the free alternative's quality rises to 1.2				
Lead ℓ_k		0.8	1.8	4.8
Optimal price	0.1	0.5	1.15	3.55
Cutoff at the old prices		0.625	0.65	0.8
Share at the old prices	0.625	0.025	0.15	0.2

Panel B. Cuts after the improvement					
Versions cut	Cut d_1	Cut d_2	Cut d_3	Loss R_k	Best for menu costs
None	0	0	0	0.0125	above 0.0069
1	0.056	0	0	0.0056	0.0035 to 0.0069
1 and 2	0.083	0.063	0	0.0021	0.0021 to 0.0035
1, 2, and 3	0.1	0.1	0.1	0	below 0.0021

Notes: Uniform types on $[0,1]$. For the free alternative, cost and price are its hosted price c_o, and for the versions, cost is the provider's marginal cost. Panel A gives the cutoffs of Proposition 1, the prices of the ladder $p_j = c_o + \sum_{k \le j} \theta_k \Delta q_k$, and the cutoffs and shares when the old prices meet the improved free alternative. Panel B gives the cuts and losses of the bottom blocks in Proposition 3(b) and the menu costs at which each block minimizes $k\kappa + R_k$. Every number has also been recomputed from the buyers' choices.

the upper envelope $u_o(\theta) = \max\{0, \max_i(\theta q_i - c_i)\}$, which is convex and piecewise linear. If every $q_i < q_1$, version 1's surplus crosses the envelope once, at θ_1, which we take to be interior, and the results of this section hold with (q_o, c_o) the option that is uniquely active at θ_1, for changes that keep θ_1 inside its segment: an improvement of that open model lowers every price, while a change in an open model that is not active at θ_1, or any change in the open models when non-use is active there, leaves every price unchanged. The best open model by capability need not be active for any type: a more capable model can be so much more expensive that a cheaper one is better for every buyer. An open model at least as good as version j and no more expensive, with one inequality strict, takes all of its buyers. The condition is sufficient but not necessary, since a version can also lose its buyers to a combination of open models and the provider's own versions. A version below the best open model but cheaper than it can keep the buyers for whom the better models cost too much.

Empirically, we call the best open model the open leader and its capability the open floor. The open leader is the relevant open option for the buyers who value capability most whenever any buyer prefers it to the other open models, while the premium at equal capability compares each closed model with the open models near its own capability, the relevant segment for its buyers. Section 6.3 checks dominance version by version.

Remaining life. Proposition 3 compares a one-time menu cost with the loss in a single period, as if the line were sold for one more period. Suppose instead that profit flows are discounted by $\delta \in (0, 1]$, that the free alternative does not improve again, and that a cut, once made, lasts for the rest of the line's life. In the simplest case the line is sold for two periods and then retired, so that when the free alternative improves, a young line has two periods to run and an old line one.

Proposition B.1 (A line with more life ahead is repriced sooner)**.** *If every version in the line has D discounted periods to run, Proposition 3 holds with κ replaced by κ/D: version k is cut only if $\ell_k \leq D\Delta^2/(4\kappa)$, and the threshold Δ^* falls with D, in proportion to $1/\sqrt{D}$ when the menu cost is small. With two-period lives, $D = 1 + \delta$ for a young line and $D = 1$ for an old one, so an improvement between the two thresholds brings a cut in a young line and none in an old one. If the line's life is random and its retirement hazard rises with age, its expected discounted remaining life falls with age, and the improvement needed to bring a cut rises with the line's age.*

The proposition concerns a whole line with a common remaining life. In a line whose versions differ in age, the retirement of one version changes the cutoffs and the value of cuts on its neighbors, so the proposition does not show that a young version in such a line is cut before an old one at the same distance from the free alternative. We read the age of the versions cut in Section 6 as consistent with the remaining-life mechanism, not as a test of it.

Launches into a sticky menu. New versions arrive at dates and qualities set by training, which the model takes as given. Propositions 1 and 2 price a whole menu whose prices are free to move, but a launch into a line whose legacy prices stay put is a different problem. Suppose the provider launches a version of quality q_n and cost c_n, sets its price, and leaves

the legacy prices and the rest of the line unchanged, and that the new version and every legacy version keep buyers. Proposition B.3 drops the last condition.

Proposition B.2 (Launch prices with sticky legacy prices)**.** *(a) A new top version, $q_n > q_J$, is priced at $p_n = p_J + \theta_n(q_n - q_J)$, where $(\theta_n - h(\theta_n))(q_n - q_J) = c_n - c_J$. Its price depends on the old top's price, quality, and cost, but not on the free alternative or on the prices below the old top. (b) A new version between the free alternative and the lowest legacy version, $q_o < q_n < q_1$, is priced against both while the free alternative keeps buyers. With uniform types,*

$$p_n = \frac{(q_n - q_o)(2p_1 - c_1 + c_n) + (q_1 - q_n)(c_o + c_n)}{2(q_1 - q_o)}, \tag{B.1}$$

which falls as the free alternative improves if $2p_1 > c_1 + c_o$, as it does when p_1 was optimal at an earlier free alternative under the same costs. At a legacy price p_1 that is optimal for the current free alternative it falls by $(q_1 - q_n)/(2(q_1 - q_o))$ per unit of improvement: by close to one half for a version launched just above the free alternative, and by close to nothing for one launched just below version 1. (c) If the provider instead replaces the whole line at the launch, every new price is set as in Proposition 1 and moves with the free alternative as in Proposition 2.

With sticky legacy prices, the free alternative therefore reaches launch prices only through versions launched next to it. A new frontier version carries forward the level of the legacy prices beneath it: with uniform types, if the old top kept its price after the free alternative improved by Δ, the new top is launched $\Delta/2$ above the price that a reset line would give it. A launch can also leave a legacy version without buyers at its stale price, which then needs no posted cut.

Replacement. Proposition B.2 assumes that every legacy version keeps buyers. A new version can instead be priced so that a legacy version loses its buyers at its stale price. Let types be uniform, and consider one legacy version of quality q and cost $c \geq c_o$ whose price p was optimal when the free alternative stood at $q_o - A$, and a new version of quality $Q > q$ and cost C, with $(c - c_o)/(q - q_o + A) < (C - c)/(Q - q) < 1$, the cost ordering of Assumption 2 at the date the legacy price was set, launched at a price P that the provider

chooses freely. Write $b = (p - c_o)/(q - q_o)$ for the legacy version's cutoff against the free alternative at its stale price, $t = \frac{1}{2}\big(1 + (C - c)/(Q - q)\big)$ for the cutoff of Proposition B.2(a), and $P_R = \frac{1}{2}(Q - q_o + C + c_o)$ and $\theta_R = (P_R - c_o)/(Q - q_o)$ for the new version's stand-alone price and its cutoff against the free alternative.

Proposition B.3 (Replacement)**.** *(a) The launch price is one of two candidates. Coexistence prices the new version at $P_S = p + t(Q - q)$, as in Proposition B.2(a). The legacy version keeps buyers if $b < t$, and the provider earns $\Pi_S = (p - c)(1 - b) + (1 - t)[t(Q - q) - (C - c)]$. Replacement prices it at P_R. The legacy version has no buyers if $\theta_R \leq b$, and the provider earns $\Pi_R = (Q - q_o - C + c_o)^2/(4(Q - q_o))$ while the new version's lead over the free alternative is worth more than its cost increment, $q_o < Q - (C - c_o)$. Beyond that point the new version has no profitable sales, and $\Pi_R = [Q - q_o - (C - c_o)]_+^2/(4(Q - q_o))$. At least one candidate is feasible, and the provider takes the more profitable. (b) The two prices are the new version's stand-alone price at two dates: $P_S = \frac{1}{2}(Q - q_o + A + C + c_o)$, at the free alternative's quality when the legacy price was set, and P_R, at its current quality, so $P_S - P_R = A/2$. Neither depends on the legacy version's quality or cost. (c) Replacement is infeasible when the legacy price is current, $A = 0$, and feasible once A reaches a threshold A_R between zero and the legacy version's lead when its price was set, $q - q_o + A$. The provider prices against the legacy version for improvements below a second threshold $A^* > A_R$ and replaces it for larger ones, and at A^* the launch price drops by $A^*/2$. (d) Under coexistence, $\partial P_S/\partial p = 1$ for a shift in the legacy price that leaves qualities and costs unchanged, and $\partial P_S/\partial q_o = 0$. Under replacement, $\partial P_R/\partial p = 0$ and $\partial P_R/\partial q_o = -\frac{1}{2}$, and the legacy version has no buyers. Whether it stays listed is the subject of Proposition B.6.*

The proposition makes replacement a regime rather than an assumption. Take a legacy version of quality 1 and cost 0.2 priced at 0.6 against a free alternative of quality 0 and cost 0, and a new version of quality 1.5 and cost 0.4. After an improvement of 0.01 the new version is launched at 0.95 and both sell. Replacement becomes feasible at an improvement of about 0.061 and is chosen from about 0.096. After an improvement of 0.12 the new version is launched at 0.89, leaving the legacy version without buyers. In a line of legacy versions whose prices were optimal at a common date, every coexistence candidate, priced against whichever

legacy version keeps buyers, gives the new top the same stale price, $P_R + A/2$, so the choice is again between the stale and the current stand-alone price. Part (b) also says what the data can separate. Because neither launch price depends on the legacy version's quality or cost, a predecessor whose price differs because its serving cost differs passes nothing through in either regime. The regimes differ in which quality of the free alternative the launch price reflects, the one when the predecessor was priced or the current one, and in whether the predecessor keeps buyers. The restriction $c \geq c_o$ matters. If the closed versions are much cheaper to serve than the free alternative is to host, the new version's stand-alone cutoff can fall to zero. It then serves every buyer at the price c_o, and its price follows the hosting cost rather than the free alternative's quality.

Launches as repricing opportunities. A launch is also an occasion to change legacy prices. The price list, the documentation, and the announcement are redone anyway, buyers reconsider which version to use, and a cut presented with a new lineup is less likely to be read as a sign of weakness or to reopen negotiated terms. Suppose that at a launch changing a legacy price costs $\kappa_L < \kappa$ per version, that types are uniform, and that the legacy prices were optimal when the free alternative stood at $q_o - A$, so that $A \geq 0$ is the improvement accumulated since the provider last reset them. Write $\Delta^*(\kappa)$ for the threshold of Proposition 3(d) at menu cost κ.

Proposition B.4 (Launches as repricing opportunities)**.** *(a) Among resets that keep every legacy version bought, the provider resets at a launch a bottom block of legacy versions, or none, with the cuts of Proposition 3 for the improvement A and the block that minimizes $k\kappa_L + R_k$. The threshold Δ^* rises with the menu cost, so in the one-period comparison of Proposition 3 an accumulated improvement between $\Delta^*(\kappa_L)$ and $\Delta^*(\kappa)$ brings no cut between launches and a cut at the next launch. (b) A new top version is launched at the price of Proposition B.2(a) less the cut on the old top: $A/2$ below it if the reset covers the whole legacy line, which happens if $J\kappa_L \leq A^2/(4\ell_J)$ and only if $\kappa_L \leq A^2/(4\ell_J)$, and at that price otherwise. Measured against the old top's price before the launch, the new top's price then falls with the improvement accumulated since the old top's price was set. (c) Suppose the free alternative does not improve again, the line runs on beyond the next launch, which is T*

periods ahead, and profits are discounted by δ. The provider cuts a block k before the launch rather than at it only if $k(\kappa - \delta^T \kappa_L) < D_T(R_0 - R_k)$, where $D_T = \sum_{t<T} \delta^t$. The condition is necessary, not sufficient. The improvement it requires grows as the launch approaches, so cuts tend to wait for the launch, and cuts between launches need large improvements.

The extension changes how the free alternative reaches launch prices. With sticky legacy prices a new top inherits the old top's level (Proposition B.2). When the launch resets the line, the new top is priced against the free alternative's current quality, and its price, measured against the old top's, falls with the improvement accumulated since the old top was priced. A version launched without a reset inherits the level set at the provider's last reset, so a launch soon after that reset carries a level set against a more recent, and better, free alternative than a launch long after it. A reset of the whole line and a replacement give a new top the same launch price, its stand-alone price, but a reset cuts the posted legacy prices, while a replacement leaves them in place without buyers and comes only after the improvement that Proposition B.3 requires. With a closed rival at a fixed price between the provider's versions (Remark B.2), the same holds for the rival's improvements, with the versions above the rival in place of the line. The single improvement A presumes that the whole legacy line was optimal at a common date. After a partial reset the distortions differ across versions. Which regime applies is an empirical question. With sticky legacy prices, a launch price responds to the free alternative only if the version enters next to it, and a new top's price reflects the free alternative's quality when its predecessor was priced. With replacement or a reset it reflects the current quality, and only a reset cuts posted legacy prices. Proposition B.6 says when the provider prefers each, and Section 6 tests what the data can separate.

Withdrawal.

Proposition B.5 (Withdrawal)**.** *(a) If an open model is at least as good as version j and no more expensive, $q_o \geq q_j$ and $c_o \leq p_j$ with one inequality strict, every buyer of positive type strictly prefers it to version j. Once $q_o \geq q_j$, restoring demand takes a price below c_o, which loses money on every sale if $c_j \geq c_o$. A version without buyers whose listing has a positive cost is withdrawn, unless contracts or migration costs keep it listed. (b) Withdrawal*

can come before the free alternative overtakes the version. At a fixed price $p_j > c_o$, the lowest version's demand ends when q_o reaches $q_j - (p_j - c_o)/\bar{\theta} < q_j$, and earlier if a higher version bounds its buyers from above. If the price is reset optimally, types are uniform on $[0, 1]$, and $a = c_j - c_o \geq 0$, a single version earns $(\ell - a)^2/(4\ell)$ at lead $\ell = q_j - q_o > a$, so with a listing cost m it is withdrawn when its lead falls to $a + 2m + 2\sqrt{m(a+m)}$, which tends to the cost gap a, not to zero, as $m \to 0$. (c) In a line, the provider keeps a version if the line's best profit with it, net of the listing cost, exceeds its best profit without it. A withdrawn version's buyers move partly to the provider's other versions, so its own margin on its sales overstates its value to the provider, and the provider's own versions can take its segment before the free alternative reaches it.

The proposition has two parts of different standing. Dominance in quality and price is a sufficient condition for a version to lose all its buyers at its posted price at once, and Section 6.3 checks it version by version. The timing of withdrawal is not tied to the overtaking: at sticky prices demand can vanish before it, and with a cost of keeping a version listed the provider withdraws it at a point set by prices, costs, buyers' valuations, and the rest of the line. With a cost of withdrawing a version, Proposition B.6 explains why a version without buyers can stay listed for months, as overtaken versions do in the data. Without switching costs, the model does not explain why such versions keep their users.

Remark B.4 (Switching costs and legacy demand). Let a buyer of type θ who uses a legacy version of quality q at price p pay a cost $\zeta \geq 0$ to move to another option, and let a new buyer pay none. An installed buyer stays on the legacy version rather than move to a version of higher quality Q at price P if $\zeta > \theta(Q - q) + (p - P)$, and the same holds against the free alternative with (q_o, c_o) in place of (Q, P). A version that a successor or the free alternative dominates in quality and price then keeps the installed buyers whose switching cost exceeds their gain from moving, while every new buyer chooses from the current menu. Its demand is a tail that shrinks as the gain from moving grows, rather than falling to zero at once. The provider can then sell new capability cheaply to new and mobile buyers and keep a legacy version with traffic, and the withdrawal of Proposition B.5 weighs the tail's revenue against the cost of serving and listing the version. The usage data of Section 6.5 fit this reading without measuring switching costs. Within Anthropic's Opus line the old version kept 0.8

percent of the two versions' tokens, a small tail, while versions an open model dominated kept 41 percent of the tokens, as larger costs of moving to another provider's model, or attributes the index does not measure, would produce.

Coexistence, replacement, or a reset. Propositions B.3 and B.4 solve two conditional problems, replacement with the legacy price frozen and repricing at a launch, and they can recommend different actions to the same provider. A provider that can reprice a legacy version at the launch, keep it at its stale price, or withdraw it compares the three in one decision. Let types be uniform, and consider one legacy version of quality q and cost $c \geq c_o$ whose price was optimal when the free alternative stood at $q_o - A$, and a new version of quality $Q = q + v$ and cost C, launched at a price the provider chooses freely, with $0 \leq a/\ell < r < 1$, where $\ell = q - q_o$, $a = c - c_o$, and $r = (C - c)/v$. The line runs for D discounted periods. At the launch the provider can keep the legacy version at its stale price, reprice it at a one-time cost κ_L, or withdraw it at a one-time cost X, the cost of migrating its users and ending its contracts. A listed legacy version costs m a period. Write

$$\Pi_F = \frac{(\ell - a)^2}{4\ell} + \frac{v(1-r)^2}{4}, \qquad L = \frac{A^2}{4\ell}, \qquad B = \frac{v(r\ell - a)^2}{4\ell(\ell + v)} \tag{B.2}$$

for the profit of the two versions when both prices are free, the loss from keeping the legacy price stale, and the legacy version's value to the line, with b, t, and θ_R as in Proposition B.3.

Proposition B.6 (Coexistence, replacement, or a reset)**.** *(a) A period in which both versions sell at the stale legacy price earns $\Pi_F - L$, a reset menu earns Π_F, and the new version alone earns $\Pi_F - B$. The legacy version's value to the line, B, is below its own operating profit in the repriced menu, and it falls as the free alternative improves, while L rises. (b) The provider chooses the option with the smallest loss per period among four: keeping the legacy version listed while both sell, $L + m$, feasible if $b < t$; keeping it listed after the new version has left it without buyers, $B + m$, feasible if $\theta_R \leq b$; resetting its price, $\kappa_L/D + m$; and withdrawing it, $B + X/D$. (c) A reset is chosen only if $\kappa_L/D \leq L$, $\kappa_L/D \leq B$ when the legacy version can be displaced, and $\kappa_L/D + m \leq B + X/D$, so only after moderate improvements: small ones leave the loss from a stale price below the cost of repricing, and large ones leave the legacy*

version worth less than its reset. (d) Displacement and withdrawal have different thresholds. Proposition B.3's threshold A_R is the improvement at which the new version's stand-alone price first leaves the listed legacy version without buyers. Setting the reset aside, if the listing cost exceeds the exit cost spread over the line's life, $m > X/D$, withdrawal is chosen from the start if $m - X/D \geq B$ at $A = 0$, and otherwise from the improvement A_W at which $L - B = X/D - m$, which lies below the switch A^* of Proposition B.3, and below A_R if and only if $L-B > X/D-m$ at A_R. If $m = X/D$ the provider is indifferent between withdrawing a displaced version and keeping it listed. If $m < X/D$, the legacy version is never withdrawn, and once the new version displaces it, it stays listed without buyers.

The proposition reads the absence of posted legacy cuts and the timing of retirements together. In the example after Proposition B.3, after an improvement of 0.12 and over one period, $L \approx 0.0041$ and $B \approx 0.0024$: with no listing or exit cost the new version displaces the legacy version, a reset that costs 0.001 beats displacement, a reset that costs 0.003 does not, and with a reset at 0.001 a listing cost of 0.002 makes withdrawal best. A provider that rarely cuts the posted price of a legacy version at its launches is one for which repricing, spread over the line's remaining life, costs more than the loss from a stale price or more than the legacy version is worth to the line. A legacy version that the new version leaves without buyers stays listed while the cost of withdrawing it, spread over the line's life, exceeds the cost of keeping it listed. With constant costs this is a choice between withdrawing at once and keeping the version until the line expires, so the proposition explains continued listing after displacement but not the date of a later withdrawal. A retirement some months after a launch needs an exit cost that falls as contracts end, or a change in costs.

Remark B.5 (The launch regimes with general types). Let F satisfy Assumption 1, and let one legacy version of quality q and cost $c \geq c_o$ be listed at a price p that was optimal at an earlier free alternative, with a new version of quality $Q > q$ and cost C priced freely. With the legacy price in place, profit as a function of the launch price has two branches: while both versions sell, $(p-c)(1-F(b))+(1-F(\theta_N))[\theta_N(Q-q)-(C-c)]$ with $\theta_N = (P-p)/(Q-q)$ and $b = (p-c_o)/(q-q_o)$, and once the legacy version has no buyers, $(1-F(\theta))[\theta(Q-q_o)-(C-c_o)]$ with $\theta = (P-c_o)/(Q-q_o)$. Each branch is single-peaked in its cutoff by the argument of Proposition 1, so the launch price is the better of two candidates, the coexistence price P_S

at the cutoff t that solves $(t - h(t))(Q - q) = C - c$, and the replacement price P_R at the cutoff θ_R that solves $(\theta_R - h(\theta_R))(Q - q_o) = C - c_o$. Replacement is feasible only if $\theta_R \leq b$. At the date the legacy price was set, b equals the optimal lowest cutoff and $\theta_R < t$ by the weighted-average argument in the proof of Proposition B.3, and $b < t$ there, so replacement is infeasible until the free alternative has improved enough for b to reach θ_R. With listing, repricing, and exit costs, the provider's choice among keeping the stale price, keeping a displaced version listed, resetting, and withdrawing compares the four per-period losses of Proposition B.6(b), with L the loss from the stale price and B the retention value in the repriced menu defined for F. The proofs that P_S and P_R are the stand-alone prices at two dates, that $P_S - P_R = A/2$, that L and B take closed forms, and that both are monotone in A use uniform types.

General type distributions. Proposition 3 rests on uniform types. For any distribution that satisfies Assumption 1, the same structure holds locally.

Proposition B.7 (Costly repricing with general types)**.** *Under Assumptions 1–2, start from the optimal menu, let the free alternative improve by a small Δ, let the menu cost be of order Δ^2, and keep the set of active versions fixed. Write $d_0 = s\Delta$, with s as in Proposition 2, $w_k = 2\Delta q_k/[f(\theta_k)(1-h'(\theta_k))]$, and $C_k = \sum_{i \leq k} w_i$. The loss from cuts d is $\sum_k (d_{k-1} - d_k)^2/w_k$ up to terms of order smaller than Δ^2. The versions cut form a bottom block, a block $k < J$ cuts $d_i = s\Delta(1 - C_i/C_{k+1})$ and leaves a loss of $s^2\Delta^2/C_{k+1}$, and, to the same order, version k is cut only if $\kappa C_k \leq s^2\Delta^2$. With uniform types, $w_k = \Delta q_k$, $s = 1/2$, and Proposition 3 is the exact counterpart.*

As in the uniform case, the condition $\kappa C_k \leq s^2\Delta^2$ is necessary, not sufficient: with a small enough menu cost every price is cut, the top included. One uniform result does not carry over: a smaller lead of the lowest version raises the loss from leaving prices unchanged, $s^2\Delta^2/C_1$, only if the marginal buyer's terms $f(\theta_1)$, $h'(\theta_1)$, and s are held fixed, and they move with the lead. Under a truncated exponential distribution on $[0, 10]$, which satisfies Assumption 1, with $\Delta c_1 = 1$, the coefficient of Δ^2 in that loss falls from about 0.0500 to 0.0279 as the lead closes from 1/2 to 1/3, because the marginal buyer moves into the thin upper tail, where less demand is at stake.

Proof. At fixed prices the lowest cutoff becomes $\tilde{\theta}_1 = (p_1 - c_o)/(\Delta q_1 - \Delta)$, which moves by $\theta_1/\Delta q_1$ per unit of Δ, while the optimal cutoff moves by $(\theta_1 - h(\theta_1))/[(1 - h'(\theta_1))\Delta q_1]$. The difference is $s/\Delta q_1$. The cutoffs θ_k, $k \geq 2$, depend only on closed prices, and their optima do not move. Since $T_k' = 0$ and $T_k'' = -f(\theta_k)(1 - h'(\theta_k))\Delta q_k$ at the optimum, a deviation of cutoff k by $e_k/\Delta q_k$ costs e_k^2/w_k to second order. Cuts d move cutoff k by $(d_{k-1} - d_k)/\Delta q_k$ relative to its new optimum, with $d_0 = s\Delta$, which gives the loss. The block structure and the block solution follow as in the proof of Proposition 3, with w_k in place of ω_k, $s\Delta$ in place of $\Delta/2$, and C_{k+1} in place of ℓ_{k+1}, and so does the necessary condition. For the counterexample, $h(\theta) = 1 - e^{\theta - 10}$. With $\Delta c_1 = 1$ the cutoff solves $\theta - h(\theta) = 1/\Delta q_1$, which gives $\theta_1 \approx 2.999$ for $\Delta q_1 = 1/2$ and $\theta_1 \approx 3.998$ for $\Delta q_1 = 1/3$, and the coefficient $f(\theta_1)(1 - h'(\theta_1))s^2/(2\Delta q_1)$ takes the two values in the text.

B.3 Proofs of the further results

The propositions have also been checked numerically, computing demand from the buyers' choices rather than from the formulas below: the exact loss and the block solutions of Proposition 3 on random menus, with the optimal set of cuts found by searching over all subsets, and over all cuts on the exact profit, including cuts that leave a version without buyers, the launch prices of Proposition B.2 under three distributions, the regimes of Proposition B.3 against a brute-force search over the launch price, the resets at launches of Proposition B.4, the joint choice of Proposition B.6 against brute-force profits of each option, and every example in the text.

Remark B.2. Let the rival keep buyers, so that the buyers of versions $K+1, \ldots, J$ are the types above $\theta_{K+1} = (p_{K+1} - p_R)/(q_{K+1} - q_R)$. Writing $p_j = p_R + \theta_{K+1}(q_{K+1} - q_R) + \sum_{k=K+2}^{j} \theta_k \Delta q_k$ for $j > K$ and repeating the argument of Lemma 1, the profit from those versions is $(1 - F(\theta_{K+1}))[\theta_{K+1}(q_{K+1} - q_R) - (c_{K+1} - p_R)] + \sum_{k=K+2}^{J}(1 - F(\theta_k))(\theta_k \Delta q_k - \Delta c_k)$. It involves neither (q_o, c_o) nor the prices of versions $1, \ldots, K$, whose profit in turn depends only on their own prices, the free alternative, and the rival, through the cutoff between version K and the rival. The two parts are maximized separately, and the first as in Proposition 1. In the example, the top cutoff is $\frac{1}{2}(1 + (2-1)/(4-2)) = 0.75$ and the top

price $1 + 0.75 \cdot 2 = 2.5$, while the lower price maximizes $(p_1 - 0.1)\big((1 - p_1) - p_1/(1 - q_o)\big)$, which gives $p_1 = \frac{1}{2}\big(0.1 + \ell_1/(\ell_1 + 1)\big)$ with $\ell_1 = 1 - q_o$.

Remark B.3. The envelope is a maximum of affine functions of θ, the zero of non-use among them, hence convex and piecewise linear. Because every $q_i < q_1$, version 1's surplus $\theta q_1 - p_1$ rises faster in θ than every segment of the envelope, so it crosses the envelope once, at θ_1, and every type above θ_1 prefers version 1, or a higher version, to every outside option. On the segment that contains θ_1 the envelope equals $\theta q_{i^*} - c_{i^*}$, or zero, so the first-order conditions and their derivatives with respect to that option's quality and price are those of Section 5 with $(q_o, c_o) = (q_{i^*}, c_{i^*})$, for changes that keep θ_1 inside the segment. A change in an option that is not active at θ_1 does not change u_o near θ_1. An open model i is better than a cheaper one k for no type if $q_i > q_k$ and $c_i - c_k \geq \bar{\theta}(q_i - q_k)$. For the dominance claim, a buyer of type θ prefers version j to open model i if and only if $\theta(q_i - q_j) \leq c_i - p_j$, which fails for every $\theta > 0$ when $q_i \geq q_j$ and $c_i \leq p_j$ with one inequality strict, and holds for $\theta \leq (c_i - p_j)/(q_i - q_j)$ when $q_i > q_j$ and $c_i > p_j$.

Proposition B.1. After the improvement the free alternative stays put, so each period brings the same loss for given cuts, and delaying a cut only shortens the time over which it pays: a cut, if made, is made at once. With D discounted periods ahead for every version, the provider chooses the block and the cuts to minimize $k\kappa + D\,\mathcal{L}$, which is D times the problem of Proposition 3 with menu cost κ/D, so its conditions hold with κ/D in place of κ. The gain from each block, $D(R_0 - R_k) - k\kappa$, rises with D, so the threshold falls with D. When the menu cost is small, so is the threshold, the gains are close to $\frac{\Delta^2}{4}(1/\bar{\ell}_1 - 1/\bar{\ell}_{k+1})$, and the threshold is proportional to $\sqrt{\kappa/D}$. With two-period lives, a line with two periods to run has $D = 1 + \delta$ and one with a single period $D = 1$. With a random life, survival probabilities G_α and hazards λ_α by age, the expected discounted life left at age α is $D(\alpha) = \sum_{t \geq 0} \delta^t G_{\alpha+t}/G_\alpha$, and each ratio $G_{\alpha+t}/G_\alpha = \prod_{u=\alpha}^{\alpha+t-1}(1 - \lambda_u)$ falls with α when the hazard rises with age.

Proposition B.2. (a) With legacy prices fixed and every option bought, the new top takes the types above $\theta_n = (p_n - p_J)/(q_n - q_J)$ from version J and changes no other cutoff, so the provider's profit changes by $(1 - F(\theta_n))[(p_n - c_n) - (p_J - c_J)] = (1 - F(\theta_n))[\theta_n(q_n - q_J) -$

$(c_n - c_J)]$. This has the form of a term of (1) and is maximized where $(\theta_n - h(\theta_n))(q_n - q_J) = c_n - c_J$, as in Proposition 1. Neither the condition nor p_J involves (q_o, c_o) or the lower prices. (b) The new version is bought by the types between $\theta^- = (p_n - c_o)/(q_n - q_o)$ and $\theta^+ = (p_1 - p_n)/(q_1 - q_n)$: it draws the types between θ^- and θ_1 from the free alternative and those between θ_1 and θ^+ from version 1. With uniform types the change in profit is $(p_n - c_n)(\theta^+ - \theta^-) - (p_1 - c_1)(\theta^+ - \theta_1)$, a concave quadratic in p_n whose first-order condition gives (B.1). With the leads $\ell_1 = q_1 - q_o$ and $\ell_n = q_n - q_o$, $\partial p_n/\partial q_o = -(\ell_1 - \ell_n)(2p_1 - c_1 - c_o)/(2\ell_1^2)$, which is negative when $2p_1 > c_1 + c_o$. A legacy price that was optimal at an earlier free alternative under the same costs satisfies this, since then $p_1 > c_o$ and $p_1 - c_1 = h(\theta_1)\Delta q_1 > 0$ at that optimum, and the margin has not changed since. At the optimal legacy price, $2p_1 - c_1 - c_o = q_1 - q_o = \ell_1$, which gives $-(q_1 - q_n)/(2(q_1 - q_o))$. (c) is Propositions 1 and 2 applied to the new line. For the statement after the proposition, a reset line prices the old top at $p_J - \Delta/2$ and the new top at that price plus $\theta_n(q_n - q_J)$, with the same θ_n.

Proposition B.3. (a) With the legacy price fixed, profit is continuous in P and has two branches. If the cutoff between the two versions, $\theta_N = (P - p)/(Q - q)$, exceeds b, both versions sell, and Lemma 1 gives the profit $(p - c)(1 - b) + (1 - \theta_N)[\theta_N(Q - q) - (C - c)]$, maximized at $\theta_N = t$. If $\theta_N \leq b$, the legacy version has no buyers, the new version sells to the types above $(P - c_o)/(Q - q_o)$, and the profit $(P - C)[1 - (P - c_o)/(Q - q_o)]$ is maximized at P_R. This branch contains P_R if and only if $\theta_R \leq b$, because at $\theta_N = b$ the new version's cutoffs against the legacy version and against the free alternative coincide. Each branch is a concave quadratic, so the optimum is the better feasible candidate. Both cannot be infeasible. While the cost ordering holds at the current free alternative, $(C - c_o)/(Q - q_o)$ is a weighted average of $(c - c_o)/(q - q_o)$ and $(C - c)/(Q - q)$, so $\theta_R < t$, and part (c) shows that the ordering fails only after coexistence has ended, where replacement is feasible. With $c \geq c_o$, $\theta_R > 0$. The upper bound $\theta_R < 1$ holds if and only if $C - c_o < Q - q_o$: while coexistence is feasible, because there $\theta_R < t < 1$, and after it ends until q_o reaches $Q - (C - c_o)$, where the new version's sales end and Π_R is zero. (b) With $p = \frac{1}{2}(q - q_o + A + c + c_o)$, $P_S = p + \frac{1}{2}(Q - q) + \frac{1}{2}(C - c) = \frac{1}{2}(Q - q_o + A + C + c_o)$. (c) Write $\bar{\ell} = q - q_o + A$ and $\bar{\ell}_N = Q - q_o + A$ for the leads when the legacy price was set, and $a = c - c_o$ and $a_N = C - c_o$.

Then $b - \frac{1}{2} = (a + A)/(2(\bar{\ell} - A))$ and $\theta_R - \frac{1}{2} = a_N/(2(\bar{\ell}_N - A))$, so replacement is feasible if and only if $\varphi(A) = (a + A)(\bar{\ell}_N - A) - a_N(\bar{\ell} - A) \geq 0$. The cost ordering at the date the legacy price was set gives $a/\bar{\ell} < a_N/\bar{\ell}_N$, so $\varphi(0) < 0$, while $\varphi(\bar{\ell}) = (a+\bar{\ell})(\bar{\ell}_N - \bar{\ell}) > 0$ because the cutoff at that date was positive. Since φ is a concave quadratic, it has a single root A_R in $(0, \bar{\ell})$ and is nonnegative from there to $\bar{\ell}$. The current ratio $a/(\bar{\ell} - A)$ rises with A and reaches $(C - c)/(Q - q) = r$ at $\bar{\ell} - A = a/r$. Coexistence ends when b reaches $t = \frac{1}{2}(1 + r)$, at $\bar{\ell} - A = (\bar{\ell} + a)/(1 + r)$, which exceeds a/r because $a/\bar{\ell} < r$, so the comparison below stays within the cost ordering until coexistence ends. On the feasible range, Π_S falls with A and is concave, because b rises and is convex in A, and Π_R falls with A and is convex, so $\Pi_S - \Pi_R$ is concave. At A_R, P_R lies on the boundary of the coexistence branch, so $\Pi_S > \Pi_R$ there. A concave function that is positive at A_R crosses zero at most once to its right. It does cross before b reaches t, where coexistence becomes infeasible: there Π_S is the profit at the boundary, which lies in the replacement branch, so Π_R is at least as large. The drop in price at A^* follows from (b). (d) $\partial P_S/\partial p = 1$, and P_S does not involve the current q_o once p is given. The replacement price P_R involves neither p nor (q, c), and $\partial P_R/\partial q_o = -\frac{1}{2}$. For the statement after the proposition, in a line whose legacy prices were optimal at a common date, the coexistence candidate against version k is $p_k + t_k(Q - q_k)$ with $t_k = \frac{1}{2}(1 + (C - c_k)/(Q - q_k))$, which by the argument of part (b) equals $\frac{1}{2}(Q - q_o + A + C + c_o)$ for every k. On each branch on which a legacy version k stays active below the new top, the derivative of profit with respect to P is $2(t_k - \theta)$ with $\theta = (P - p_k)/(Q - q_k)$, which has the sign of $p_k + t_k(Q - q_k) - P$. Every such branch therefore increases up to the same price $P_R + A/2$ and falls after it, and the branch on which no legacy version keeps buyers peaks at P_R, so these are the only two interior candidates.

Proposition B.4. (a) The legacy prices were optimal at $q_o - A$, so the proof of Proposition 3 applies with A in place of Δ and the leads $\ell_k = q_k - q_o$: the exact loss is (3) with $d_0 = A/2$. The new top adds the term $(d_J - d_n)^2/(q_n - q_J)$, where d_n is the amount by which the new top is priced below its price in Proposition B.2(a). The new price carries no menu cost, so $d_n = d_J$ and the term vanishes. A run of changed legacy prices that ends at the old top still recovers nothing when the version below it keeps its price, and the block

structure and the choice of block are those of Proposition 3 with κ_L in place of κ. The same holds wherever the new version enters. If it splits a step of the legacy line into ω_L and ω_R, between options whose cuts are d_L and d_R (with $d_L = d_0$ at the bottom), its own cut d_n is free, and $\min_{d_n}[(d_L - d_n)^2/\omega_L + (d_n - d_R)^2/\omega_R] = (d_L - d_R)^2/(\omega_L + \omega_R)$, reached at $d_n = (\omega_R d_L + \omega_L d_R)/(\omega_L + \omega_R)$. Optimizing the new price therefore restores the legacy term for that step, and the legacy problem is unchanged as long as every option keeps buyers. Each gain $R_0 - R_k - k\kappa$ falls with κ, so the smallest improvement at which some gain is positive, Δ^*, rises with κ. (b) The new top's price is $p_J - d_J + \theta_n(q_n - q_J)$, with $d_J = A/2$ for the whole legacy line and $d_J = 0$ for a smaller block (Proposition 3(b)). The conditions are those of Proposition 3(c) with κ_L. (c) Cutting block k at once costs $k\kappa$ and leaves the loss R_k in every period. Cutting the same block at the launch instead loses R_0 in each of the T periods before it, costs $k\kappa_L$ at the launch, and leaves R_k after it. The losses after the launch are the same, so cutting at once is cheaper only if $k\kappa + D_T R_k < D_T R_0 + \delta^T k\kappa_L$. With $x = \delta^T$ and $D_T = (1-x)/(1-\delta)$, the condition requires $R_0 - R_k > k(\kappa - x\kappa_L)(1-\delta)/(1-x)$, whose right side rises with x, its derivative having the sign of $\kappa - \kappa_L > 0$. Since $R_0 - R_k$ rises with A, the improvement that a cut before the launch needs grows as the launch approaches.

Proposition B.5. (a) A buyer of type θ prefers version j to the free alternative only if $\theta(q_j - q_o) > p_j - c_o$. With $q_o \geq q_j$ and $c_o \leq p_j$ the left side is at most zero and the right side at least zero, and for $\theta > 0$ one of the two inequalities is strict. Once $q_o \geq q_j$, a buyer of type $\theta > 0$ buys version j only if $p_j < c_o - \theta(q_o - q_j) \leq c_o$, below cost if $c_j \geq c_o$. (b) The lowest version is bought only by types above $(p_j - c_o)/(q_j - q_o)$, which reaches $\bar{\theta}$ at $q_o = q_j - (p_j - c_o)/\bar{\theta}$. A higher version caps its buyers at the next cutoff, which ends its demand earlier. With uniform types and a single version, profit at cutoff θ is $(1-\theta)(\theta\ell - a)$, maximized at $\theta = \frac{1}{2}(1 + a/\ell)$ with value $(\ell - a)^2/(4\ell)$ for $\ell > a$. This rises with ℓ and equals m at the larger root of $(\ell - a)^2 = 4m\ell$, $\ell = a + 2m + 2\sqrt{m(a+m)}$. (c) The buyers of a withdrawn version choose among the remaining options, which include the provider's other versions, so the line's profit falls by less than the version's own margin on its sales.

Proposition B.6. (a) By Lemma 1, the reset menu's profit is the sum of the legacy version's stand-alone term against the free alternative, $(\ell - a)^2/(4\ell)$, and the increment of the new version, $v(1-r)^2/4$. With the legacy price stale, $p - c_o = \frac{1}{2}(\ell + A + a)$, and the legacy term at the coexistence price becomes $(p-c)(1-b) = ((\ell-a)^2 - A^2)/(4\ell)$, while the new version's term is unchanged, so coexistence earns $\Pi_F - L$. The new version alone earns $(\ell+v-a-rv)^2/(4(\ell+v))$. With $x = \ell - a$ and $y = v(1-r)$, $x^2/\ell + y^2/v - (x+y)^2/(\ell+v) = (vx - \ell y)^2/(\ell v(\ell+v))$ and $vx - \ell y = v(r\ell - a)$, which gives $\Pi_F - (\Pi_F - B) = B$. Some of the legacy version's buyers move to the new version when it is withdrawn, so B is below the legacy version's own operating profit in the repriced menu. At the stale price its value relative to selling only the new version is $B - L$, which can be negative. With $a \geq 0$, $d\ln B/d\ell = 2r/(r\ell - a) - 1/\ell - 1/(\ell+v) > 0$ because $2r/(r\ell - a) \geq 2/\ell$, so B falls as q_o rises, and L rises because A rises and ℓ falls. (b) Over D periods the four options earn $D(\Pi_F - L - m)$, $D(\Pi_F - B - m)$, $D(\Pi_F - m) - \kappa_L$, and $D(\Pi_F - B) - X$. Dividing by D gives the losses, and the feasibility conditions are those of Proposition B.3. (c) The conditions compare the reset with each other option. At $A = 0$, $L = 0 < \kappa_L/D$. As A rises, B falls toward zero, which it reaches when a/ℓ rises to r, or at the overtaking if $a = 0$. The legacy version can be displaced well before, so for large enough improvements $\kappa_L/D > B$ and keeping the displaced version, or withdrawing it, beats a reset. (d) Without a reset, withdrawal beats keeping a displaced version if and only if $X/D < m$, and it beats coexistence if and only if $L - B \geq X/D - m$. By (a), $L - B$ rises with A, and it is zero at A^*, where $\Pi_S = \Pi_R$. If $m > X/D$, withdrawal is chosen from $A = 0$ if $B \leq m - X/D$ there, and otherwise from the root A_W of $L - B = X/D - m$, which lies below A^* because $L - B$ is negative at A_W and zero at A^*. Since $L - B$ rises with A, $A_W < A_R$ if and only if $L - B > X/D - m$ at A_R. If $m < X/D$, coexistence beats withdrawal below A^*, displacement is feasible from $A_R \leq A^*$ on, and a displaced version kept listed beats withdrawing it.

C Detail Behind the Main Text

This appendix gives the detail and the estimates behind several passages of the text, in the order of the text.

C.1 Data

Usage. The OpenRouter marketplace (OpenRouter, 2026) publishes weekly, and for part of the period daily, token volumes by model. We use them for the usage checks in Appendix F: 117 weeks of weekly volumes (65 captures), whose named models carry 59 percent of the marketplace's tokens, and daily volumes of the nine leading models from 11 September 2025 to 10 May 2026. Many captures also hold a table of every model's tokens, which Section 6.5 uses (Appendix F).

C.2 Adjustment through versions

A chained Fisher-type imputation index in the spirit of Aizcorbe (2006) and Byrne et al. (2018) shows how much of the decline depends on this. Each pair of adjacent weeks is compared on the versions listed in the earlier week, with the current price of a version that exits imputed from the hedonic function, and on the versions listed in the later week, with the earlier price of a version that enters imputed. The geometric mean of the two comparisons is chained. List prices come without quantities, so every version gets equal weight. The index falls 29 percent a year with imputations from the pooled regression and 35 percent with adjacent-period regressions, 57 to 72 percent of the time-dummy decline in logs, against 7 percent for continuing versions alone. An entering version is priced -0.50 log points relative to the hedonic line of the week before. The index formula does not change the finding that the decline arrives mostly through new versions, but it does change the size of the decline, which lies between 29 and 45 percent a year.

The free alternative behaves differently. The cheapest hosted price of an open-weight model of given ECI falls 36 percent a year in the litellm listings, but this index is fragile, for two reasons. First, hosted open prices barely depend on capability. A point of ECI adds only 0.021 log points once active parameters are controlled for, while the price elasticity with respect to active parameters is 0.67 (0.14). On the Artificial Analysis data the index coefficient is 0.003 (0.006) and the elasticity in active parameters 0.68 (0.10), across 116 open models. Hosted prices track the cost of serving a model, which depends on its size, not on what it can do. With size held fixed the litellm index falls 18 percent a year. Second,

the litellm file adds cheap hosts late and later lists hosts that Artificial Analysis does not benchmark. Across 45 open models in both sources, the litellm cheapest price exceeds the Artificial Analysis one by 0.44 log points in the first quarter after listing, and the gap is −0.27 to −0.43 log points after five quarters (Appendix E). On the curated Artificial Analysis host set, the cheapest price of a given open model rises 2 percent a year. We therefore treat the hosted price as the free alternative's cost of use at a point in time, and not as a reliable index of its decline.

List prices are sticky. Within a model, list prices rarely change. The clean panel has 11 price cuts in 4,732 week-to-week intervals. The 8 provider cuts make 9 of them, because Google posted a single price for both versions of Gemini 1.5 Pro, and the other 2 were temporary promotions. A within-model regression of the log price on the lead over the open floor and a trend gives a coefficient of 0.004 (0.003). One tariff changed. When Google took Gemini 2.5 Flash out of preview in June 2025, it replaced the preview's two output prices, 0.60 dollars per million tokens without thinking and a higher one with it, by a single price of 2.50 dollars, and raised the input price from 0.15 to 0.30 dollars. The file moved the preview to the new prices, so for use without thinking the price rose and for use with thinking it fell. No other continuing version's price rose. Two versions launched late in the sample, Gemini 3.7 Flash and Gemini 4 Argon, were listed at introductory prices that Google said would double, from January 2027 and December 2026, and the panel carries the prices in effect, as it does during the temporary promotions. The decline in quality-adjusted prices arrives at launch. Across the 87 closed models listed within forty-five days of their release, the log launch price rises by 0.227 per ECI point (0.026) and changes by −3.19 log points a year at given capability (0.45; $R^2 = 0.45$). This time slope is much steeper than the hedonic index's fall of 45 percent a year, and the two measure different things. Capability at launch rises almost in step with the launch date, by about 15 ECI points a year (correlation 0.91), so the launch regression reaches a fixed capability only by extrapolating across dates. Along the path the launches actually take, the log launch price rises by 0.31 a year. The index measures the fall at fixed characteristics among all versions on sale each week, legacy versions included, whose list prices barely move. Conditional on the distance to the closed frontier,

a model's lead over the open floor at launch carries 0.137 log points per point (0.048), and 0.094 (0.062) with a time trend. The schedule is the same per task, for the 59 launches that match Artificial Analysis token counts (Appendix F).

C.3 The premium over the free alternative

Attributes beyond capability. Equal capability is not equal service: models and hosts differ in speed, latency, context window, and numerical precision, and the premium could pay for those. It does not shrink when they are held fixed (Table F.4 in Appendix F). Against the cheapest standard listing, without tags for quantization (reduced numerical precision, which hosts use to serve a model more cheaply) or for a discounted or premium tier, the premium is 1.92 log points, against 1.95 on all listings, and in the Artificial Analysis data adding each model's log output speed and log time to the first streamed chunk (the wait before the first words of an answer appear) raises the premium from 1.80 to 2.33 log points per token on the models with speed data (0.23), because the closed models are faster and faster models are cheaper at equal capability. Equal measured capability may also overstate the open models' true capability: they score worse on private benchmarks than on public ones, which suggests that they are tuned to the public benchmarks from which the indexes are built (Edwards and Emberson, 2026).

C.4 Where and when posted prices are cut

Over the 4,006 closed model-weeks at risk, a provider cut a listed price about once per 10 model-years. In the eight weeks after the 25 frontier open releases the rate is 0.111 cuts a week, against a base rate of 0.066.

The eight cuts one by one. Table F.9 in Appendix F lists the 8 cuts with what surrounded each, so that the claims of Section 6.2 can be read against the events they rest on. Three things stand out. First, every cut came within sixty days of a frontier open release, and 6 within sixty days of a major one, but 85 percent of the panel's weeks lie within eight weeks after a frontier release, and each of the 7 decisions came 11 to 20 days after the latest

one, a regularity noticed after seeing the dates and not tested. Second, closed rivals launch near the open floor all the time. Over the sample 73 closed versions of any developer were launched within five ECI points of the open floor, 0.6 a week, so every cut has a rival launch within five points of the cut version in the four weeks on either side, and the coincidence by itself says nothing. The test is whether cuts cluster after such launches, the closed events that compete with the versions cut, and they do not. The cut rate in the eight weeks after a near-floor closed launch is -0.008 a week relative to the eight weeks before, against 0.060 after frontier open releases, a difference of 0.068 (one-sided $p = 0.032$, with the dates of the cut decisions redrawn), and the same holds when each provider's cuts are counted only around other developers' near-floor launches (a post-minus-pre cut rate of -0.003 a week around those launches against 0.015 around frontier open releases, one-sided $p = 0.031$, with the dates of the cut decisions redrawn). Third, the block prediction of Proposition 3. In 4 of the 8 cuts a lower version of the same provider that was still active, above the price-adjusted floor and not displaced by an own version at least as capable and no dearer, kept its price. Read as one line per provider, those cuts are not bottom blocks. Read line by line, with lines the tier classes of the regressions (small, standard, and pro), 2 remain, and in both the version that kept its price belongs to a different product family, a non-reasoning small model beside the reasoning o1-mini and a coding model beside GPT-5.6 Terra. They fit the prediction only if lines are also split by product family, a definition made after reading the cuts.

C.5 When versions are retired

Announcements come in batches, 21 provider-weeks with up to 8 versions each. Given that a provider retires some versions in a week, the ones it retires are below the floor more often than the same number drawn at random from its versions listed that week would be (50 against 39.5 expected; exact $p < 0.001$). But the data cannot separate being overtaken from aging. 93 percent of the model-years above the floor fall in a version's first six months, when versions on either side of the floor are retired at the same low rate (0.11 and 0.11 per model-year). The record of the undominated versions is also confined to young ages ($p = 0.31$ within age bands). A local comparison at the floor is too thin to help (Appendix F). Retirement also

comes late: the 31 versions retired after an open release pushed them below the floor were retired a median of 34 weeks after the crossing, and never sooner than 7 weeks. Retirement thus follows the overtaking of a version, which Proposition B.5 allows but does not require, and later than the model would have it without a cost of withdrawing a version, a cost that lets a version without buyers stay listed (Proposition B.6). A lifecycle in which a version's retirement is announced at about a year of age would produce the same pattern, and the data cannot tell the two apart. The rising age profile is itself the premise of the remaining-life mechanism of Proposition B.1: a retirement hazard that rises with age is what makes young versions the ones worth repricing. Adding the retirements of Mistral's closed models, Amazon Bedrock's legacy notices, and the panel's versions back to 2023 does not resolve this (Table F.19 in Appendix F).

D Data Construction

Snapshots and version matching. The litellm file's commit history gives one snapshot per week. A listing is closed if its provider is OpenAI, Anthropic, Google's Gemini API, or xAI, its mode is chat or responses, and its input and output prices are positive. Dated keys (for example `gpt-4o-2024-08-06`) are matched to the model version in the ECI released within fourteen days of the key's date. Undated aliases (`gpt-4o`, `gemini-2.5-flash`) follow the providers' documented alias moves, so a price change that reflects an alias pointing to a newer version is not recorded as a cut. Preview names are kept as their own versions, and xAI's reasoning variants are matched to the reasoning version in the ECI. Listings after a model's shutdown date are flagged and dropped (311 model-weeks).

Validation against provider pages. We fetched archived captures of the OpenAI, Anthropic, Google, and xAI pricing pages and documentation from the Wayback Machine at quarterly intervals and parsed the prices with page-specific parsers. A check compares a model's output price on the page with the file's price in the nearest week. The disagreements concern five entries on the pages. Three are legacy models whose entries the file never updated (Claude Instant and Claude Instant 1.2, Gemini 1.0 Pro), and two are provider

Table D.1: Verified price events (USD per million output tokens)

Event	Model	Class	Provider date	Old	New
V01	Gemini 1.5 Flash (May 2024)	provider cut	2024-08-12	1.05	0.3
V02	GPT-4o (May 2024)	alias moved to a newer version	2024-10-02	15.0	10.0
V03	Gemini 1.5 Pro (Sept 2024)	provider cut, missed by the file	2024-10-01	10.5	5.0
V04	Claude 3.5 Haiku	provider cut	2024-12-03	5.0	4.0
V05	o1-mini	provider cut	2025-01-31	12.0	4.4
V06	o3	provider cut	2025-06-10	40.0	8.0
V07	GPT-3.5 Turbo (Nov 2023)	file correction	–	2.0	1.5
V08	GPT-5.5 Pro	wrong launch price in the file	–	360.0	180.0
V09	GPT-5.6 Terra	provider cut	2026-07-30	15.0	12.0
V10	GPT-5.6 Luna	provider cut	2026-07-30	6.0	1.2
V11	GPT-5.6 Sol	temporary promotional cut	2026-08-21	30.0	20.0
V12	Gemini 3.6 Flash	temporary promotional cut	2026-08-17	7.5	3.75
V13	Grok 3	stale entry after retirement	–	15.0	2.5
V14	Grok 4	stale entry after retirement	–	15.0	2.5
V15	Grok-3 mini	stale entry after retirement	–	0.5	2.5
V16	Grok 4.20	provider cut	2026-05-06	6.0	2.5
V17	Grok 4 Fast	stale entry after retirement	–	0.5	2.5
V18	Gemini 2.5 Pro (Mar 2025)	wrong launch price in the file	–	1.0	10.0
V19	Gemini 2.5 Flash (May 2025)	tariff change at general availability	2025-06-17	0.6	2.5
V20	Claude Sonnet 5	wrong launch price in the file	–	15.0	10.0

cuts that the file recorded late or not at all (Claude 3.5 Haiku, Gemini 1.5 Pro), which the verified events correct.

Verified price events. Every change of a model's output price in the weekly panel was traced to the commit that made it and compared with the provider's pages before and after. Table D.1 lists the classified events. The corrected prices enter the clean panel, and provider cuts are dated by the provider's announcement. A cut is a change in a posted price. Google posted one price for Gemini 1.5 Pro that covered both of its versions and cut it in October 2024, which counts as one cut, while OpenAI's same-day cuts of GPT-5.6 Terra and Luna changed two posted prices and count as two cuts in one decision. The version-level tests use the September version of Gemini 1.5 Pro. The May version, which sat below the open floor, is left out of them.

Open-weight releases. Table D.2 lists the major releases. Licence, parameters, and country come from Epoch's model database, with active parameters of mixture-of-experts models from the developers' documentation and the Artificial Analysis records.

Artificial Analysis. The models page is a Next.js page whose data sit in embedded JSON chunks. Three layouts occur between 2024 and 2026, and we parse each model record with

Table D.2: Major open-weight releases, January 2024 – October 2026

Date	Model	Leader gain (ECI)	Country
2024-04-17	Mixtral 8x22B	3.60	France
2024-05-07	DeepSeek-V2 (MoE-236B, May 2024)	1.79	China
2024-07-23	Llama 3.1-405B	3.49	United States
2024-12-12	Phi-4	1.51	United States
2024-12-26	DeepSeek-V3	1.82	China
2025-01-20	DeepSeek-R1	6.52	China
2025-05-28	DeepSeek-R1 (May 2025)	1.98	China
2025-07-25	Qwen3-235B-A22B-Thinking (Jul 2025)	2.63	China
2026-01-27	Kimi K2.5	1.98	China
2026-04-07	GLM-5.1	1.75	China
2026-07-16	Kimi K3	5.92	China

a JSON decoder. Captures cut short by the archive are kept for the complete records they contain, except the 8 captures from July 2026 on, in which the page itself carries only its first 24 to 28 records, and which are not used. The index version changes when almost every model's score changes between consecutive captures. The cost of an index run is reported directly from mid-2026 and is computed from the reported token counts and prices before that.

OpenRouter. From February 2025 each archived rankings page carries the main chart of the previous 52 weeks. We take each week from the capture whose window ends soonest after it, so that the named models are the leaders of that period, and never use a capture's last, partial week. From the capture of 7 October 2025 to that of 12 May 2026 each page also embeds two daily charts of the preceding thirty days, tokens by model in English-language prompts and in prompts in a programming language, each day naming its nine leading models and an Others bucket. We identify the charts by their labels where the page has them and otherwise by size, and take each day from the capture whose window ends soonest after it. Overlapping captures agree on the daily totals and, for nearly all days, on the named models. Captures between mid-May and mid-August 2026 hold no chart data, and later pages show only a weekly ranking. The named models' share of the weekly tokens averages 57, 62, and 51 percent in 2024, 2025, and 2026, and the nine models named each day carry between 44 and 78 percent of English-prompt tokens.

Retirement records. The retirement file records the announcement date, the shutdown date, and the named successor for each retired model identifier. A panel model is matched to its own identifier first and to a normalized name otherwise, and the earliest announcement counts. A dated snapshot's record is matched only to that snapshot, a record of a ChatGPT alias (an identifier ending in chat-latest) is not matched by name, and a deprecation the provider later withdrew is not counted. Where a provider's page gives no announcement date, as for the Gemini 1.5 versions, the shutdown date stands in. The risk set includes 4 preview versions of Gemini 2.5 Pro and Gemini 2.5 Flash whose ends are not counted as retirements, because their generally available versions continued. Since 2 of them ended above the floor, counting them would lower the rate ratio. Almost every record names a successor. The exceptions are records where the provider's page has no successor field, so we do not distinguish retirements with and without a successor.

E The Price of the Free Alternative in Two Sources

Table E.1 compares the cheapest hosted price of the same open-weight model in the litellm file and in the Artificial Analysis data, by quarters since the model's first listing in the file. The file lists a model first at expensive hosts and adds cheap ones late, and after a year it lists hosts below the curated Artificial Analysis set. Measured in the file, the cheapest hosted price of a newly listed open model changes by -0.98 log points over 26 weeks while the number of listed hosts rises from 1.5 to 5.8 (24 models; Figure E.2). A closed model's list price changes by -0.09 log points over the same horizon (51 models). Most of the hosted decline is a difference between sources.

F Robustness

Table F.1 lists every claim in the paper with its design, its main estimate, and the strength of the evidence. The paragraphs that follow report, in this order: the premium by segment, source, screen, and attribute, with the dispersion across hosts and the premiums at launch and over a model's life (Section 6.1); the launch regressions and the predecessor tests (Sec-

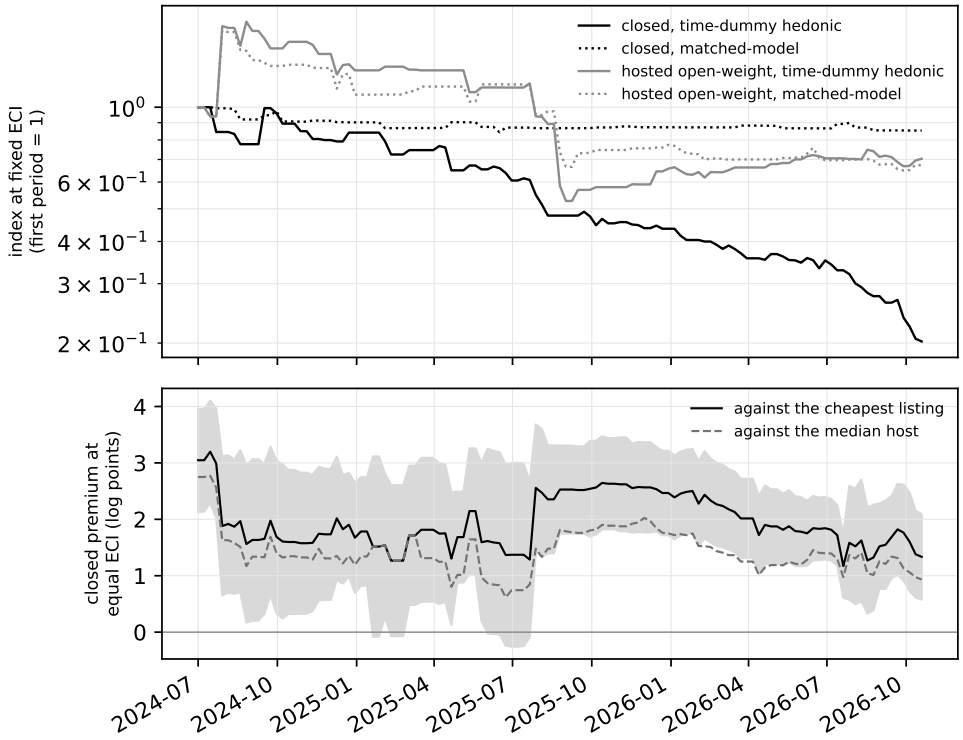

Figure E.1: Quality-adjusted prices and the closed premium, with the hosted open-weight indexes. Top: time-dummy hedonic and matched-model price indexes for closed models and for hosted open-weight models (cheapest host), weekly, log scale. Bottom: the premium of Figure 1, drawn here with the cheapest listing solid, with 95 percent confidence bands, and the median host dashed. The drop in the hosted index in August 2025 is the release of gpt-oss, a capable open-weight model that is cheap to serve.

Table E.1: Cheapest hosted price of the same open model: litellm minus Artificial Analysis (log points)

Quarters since listing	Model-captures	Mean gap	Median gap	litellm higher (%)
0	202	0.44	0.10	55
1	172	0.19	0.00	44
2	175	−0.04	0.00	30
3	107	−0.10	0.00	22
4	98	0.02	0.00	24
5	56	−0.27	−0.28	7
6	53	−0.43	−0.36	0

Notes: 45 open-weight models matched by normalized name; the litellm price is the cheapest listing in the nearest earlier week, at most ten days before the Artificial Analysis capture.

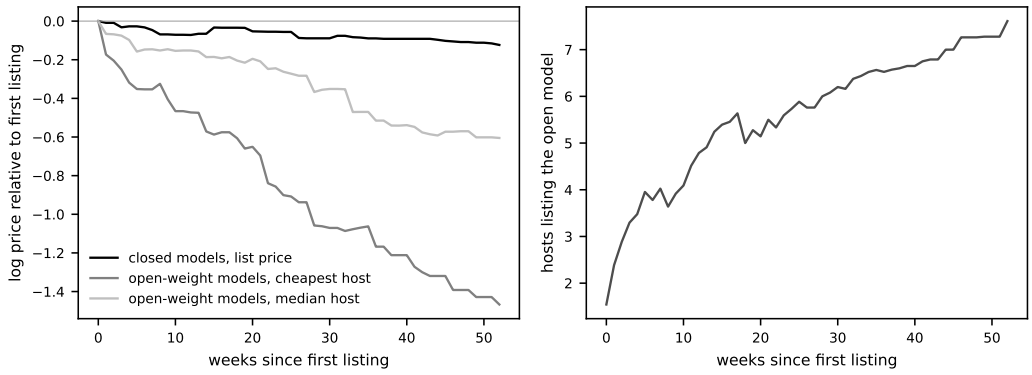

Figure E.2: Prices after first listing in the litellm file: closed models' list prices and open-weight models' cheapest and median hosted prices, log change from the first listing (left), and the number of hosts listing the open model (right). Most of the hosted decline reflects the file's coverage of hosts, as the text of this appendix shows.

tion 6.4); the primary and secondary tests of the location, size, age, and timing of cuts, with the price-adjusted floor and the launch timing (Section 6.2); the index formula; the dose response; the retirement hazards, exact tests, and extensions (Section 6.3); usage by version (Section 6.5) and the usage tests with their power; discounts; host dispersion; and the two benchmark markets.

The premium by segment and near the open leader. The monthly panel gives trends of −0.16 (0.26) against the median host and −0.10 (0.27) against the cheapest listing. Allowing separate ECI slopes and evaluating the premium at the open leader gives a lower level and the same trend (mean 1.84, trend −0.03). The closed slope differs from the open one by −0.022 (0.047). By tier, the premium of the providers' flagship models has a trend of −0.08 a year (0.26) and that of their small tiers 0.73 (0.32), a clear rise. Within five ECI points of the open floor the premium has no clear trend, 0.29 (0.32) against the cheapest listing and 0.12 (0.31) against the median host. The raw prices in the same band, in panel C of Table F.5, appear to diverge faster. In a descriptive fit the cheapest closed price rose by 0.40 log points a year and the cheapest hosted price fell by 0.55. These trends describe a band that moves with the open leader, which gained 14 ECI points a year. Cheap closed tiers that qualified in 2024, such as Gemini 1.5 Flash and GPT-4o mini, fell out of the band and newer, pricier tiers such as GPT-5.6 Luna took their place, while the cheapest open

Table F.1: The paper's claims: design and strength

Claim	Design	Main estimate	Strength
Quality-adjusted prices fall through new versions (P1)	Time-dummy and matched-model hedonic indexes; chained Fisher-type index	−45% a year; −4% within model; 54% through launches and 32% through withdrawals	Measured. The size depends on the index (−29% Fisher-type)
The free alternative has a price, and closed models carry a premium at equal capability (P2)	Pooled hedonic regression with a weekly closed indicator; per task on the Artificial Analysis index	4.2× the median host per token; in benchmark data 6.0× per token and 4.0× per task	Measured. A price premium, not a markup
The premium's trend	Trend of the weekly premium	−0.14 (0.28) log points a year against the median host	No clear trend on litellm prices; narrowing on Artificial Analysis prices
Launch prices rise with the lead over the open floor (descriptive)	Launch schedule; recent gains of the open leader and the closed frontier	0.137 per point of lead (0.094 with a trend); 0.003 per point of ninety-day open gain, not distinguishable from zero	Measured. The response to open gains is not separable from the trend and the rivals
Launch prices respond to the free alternative next to it, not at the top (P5)	Recent gains interacted with the lead over the floor at launch	0.023 near the floor, −0.082 well above it, per point of ninety-day open gain	Not supported. Launches well above the floor respond to the closed frontier's gains, over ninety days the provider's own
New tops respond to recent closed-frontier gains given their predecessors' prices, and not detectably to open gains (P5)	New tops against their predecessors; provider cuts at their own launches	−0.125 per point of ninety-day closed gain and −0.037 per point of open gain, given the predecessor's price; 0.005 per point of gain since the predecessor was priced; 1 of 46 predecessors cut at the launch	Suggestive. Designed after the split, and separates replacement from coexistence only in part
Cuts follow frontier open releases (P3)	Randomization inference with the dates of the cut decisions redrawn; placebo releases	0.042 decisions a week; $p = 0.096$	Suggestive, on 7 decisions, and resting on OpenAI's three
Cuts follow open releases, not closed ones (P3)	Randomization inference on the difference between the two post-minus-pre cut rates	0.067 cuts a week; one-sided $p = 0.045$	Suggestive, on 7 decisions
The versions cut sat near their provider's own frontier (P3)	Within-week percentile of the own-frontier distance; permutation of the median	79 percent; $p = 0.002$; median 3.5 against 12.3 points	Tested, on 8 cuts
Cuts fall near the floor, and bigger steps bring more (P3)	Event time by position; slope on the size of the release	$p = 0.034$ raw, 0.10 to 0.16 adjusted; slope $p = 0.044$	Suggestive
The versions cut are young (P3)	Age percentile of the cut version among the versions listed	0.14; $p = 0.0001$	Tested, on 8 cuts
Versions are withdrawn, not repriced (P4)	Exact rate ratios; hazards; price dominance	8.1 (2.1 to 69); 1.7 at equal age	Measured. Not separable from age
The top of the line shows no detectable response (P5)	Top versions with model effects; frontier launches	0.0023 (0.0018) per point of the open floor	Measured within a model; not separable from the rivals at launch
Usage concentrates on recent versions; dominated versions keep many	Complete usage tables on OpenRouter; tests around releases	72% of tokens under six months; 41% dominated; 0.01 at releases	Measured, one marketplace; release tests uninformative

Notes: Measured: a descriptive estimate whose sign and size do not depend on a causal design. Tested: a hypothesis test against a stated null. Suggestive: in the predicted direction but imprecise or not robust to the specification. Not supported: estimates that do not show the predicted pattern. Uninformative: too imprecise to speak to the claim. The predictions P1–P5 are those of Section 5. Sections 4 and 6 and this appendix give the estimates and their robustness. The estimates of the usage tests around releases are in this appendix only.

model near the leader went from Llama 3.1 70B to DeepSeek's mixture-of-experts models, which are cheap to host for their capability. The hedonic premium, which compares models of equal capability, is the measure of the gap.

The premium on Artificial Analysis host prices. The litellm file adds cheap hosts late (Section 4), so we repeat the regression with each open model's price taken from the cheapest host that Artificial Analysis benchmarks, matched to the ECI version by version (73 of 121 open models). Its archived pages list open models only until June 2026, which removes the last 20 weeks. The premium is higher on these prices and narrows faster. On Artificial Analysis prices the premium averages 2.51 log points, a multiple of 12.3, and falls from 2.98 in 2024 to 2.02 in 2026 (trend −0.74, 0.37). On the cheapest host in either source it averages 2.34 with a trend of −0.49 (0.29). On the 823 open model-weeks priced by both sources the trend is −0.01 (0.40) with litellm prices and −0.57 (0.36) with Artificial Analysis prices, so the difference comes from the prices rather than the sample: early in the sample litellm lists open models at expensive hosts. Within five points of the open floor its trend is −0.41 (0.39) on Artificial Analysis prices and −0.00 (0.32) on the cheapest price in either source.

The free alternative's price. Table F.2 repeats the hedonic premium regression of Section 6.1 with three prices for the open-weight models: the cheapest litellm host, the cheapest host benchmarked by Artificial Analysis, and the cheaper of the two. Artificial Analysis models are matched to the ECI by the stem of the name (parentheses, version codes, and words such as instruct or reasoning removed, parenthesised sizes kept), and among candidates the release date closest to the ECI date within sixty days decides. The price is the latest capture at most 35 days before the week. Figure F.1 shows the three weekly series. The litellm prices in the main series pass a screen for file errors that drops host listings more than twenty times above or below the median of the other hosts of the same model and week: 613 listings, among them 449 W&B prices about ten thousand times too high and 108 Hyperbolic prices far below the other hosts, for Llama 3.1 405B and the May 2025 version of DeepSeek-R1, models whose cheapest host benchmarked by Artificial Analysis charged several times more.

Table F.2: The closed premium at equal ECI by source of the open-weight price (log points)

Open-weight price	Weeks	Open models	Mean	2024	2025	2026	Trend	Trend within 5
litellm hosts (main)	121	64	1.95	1.94	2.02	1.86	−0.02 (0.29)	0.29 (0.32)
Artificial Analysis hosts	101	59	2.51	2.98	2.48	2.02	−0.74 (0.37)	−0.41 (0.39)
cheapest host in either source	121	74	2.34	2.50	2.58	1.93	−0.49 (0.29)	−0.00 (0.32)
common sample: litellm hosts	101	43	2.08	2.30	1.97	2.08	−0.01 (0.40)	–
common sample: Artificial Analysis hosts	101	43	2.43	2.82	2.40	2.01	−0.57 (0.36)	–
litellm hosts, unscreened	121	64	2.01	2.21	2.02	1.88	−0.14 (0.29)	0.07 (0.32)
cheapest host in either source, unscreened	121	74	2.37	2.59	2.60	1.94	−0.52 (0.29)	−0.07 (0.33)

Notes: Same regression as Table 3, panel A: closed and open models no more than ten ECI points below the open floor, week effects, a common ECI slope, and a closed indicator for each week. Trends are OLS slopes of the weekly premium on time in years, with standard errors from the covariance of the weekly premiums clustered by model. The common sample keeps the open model-weeks priced by both sources. The litellm prices are screened (main series); the unscreened rows keep the host listings more than twenty times above or below the median of the other hosts of the same model and week.

Without the screen (the last two rows of Table F.2), the litellm premium averages 2.01 with a trend of −0.14 (0.29), monthly −0.20 (0.27), and within five points of the open floor it has no trend (0.07, 0.32); on the cheaper of the two sources the trend is −0.52 (0.29). The hosted open-weight index falls 33 percent a year without the screen, against 36 percent with it. Table F.3 varies the screen: thresholds of ten and fifty times the other hosts' median, and a rule that drops only the listings that Artificial Analysis contradicts, meaning that every price it records for the same model at the same host in its latest capture at most 35 days earlier lies more than three times above or below the litellm price. Of the 11,812 litellm listings that Artificial Analysis prices at the same host, 84 percent lie within half again of its price and 1,103 are contradicted, 499 of them because the litellm price is lower. The rule cannot judge listings at hosts that Artificial Analysis does not benchmark, among them Groq's 74 weekly listings of Llama 3.1 405B at the price of Groq's smaller Llama models; the last row of the table drops those as well. Against the median host the trend hardly moves, and no trend in this paragraph is distinguishable from zero. Against the cheapest listing it falls by about 0.00 to 0.02 log points a year once the evident errors are gone and by about 0.14 when the cheap listings stay in. In detail, the trend against the cheapest listing is −0.14 with no screen and −0.13 with a threshold of fifty times; dropping only the listings that Artificial Analysis contradicts gives −0.12, and dropping as well Groq's listing of Llama 3.1 405B, which Artificial Analysis never benchmarks, gives 0.00.

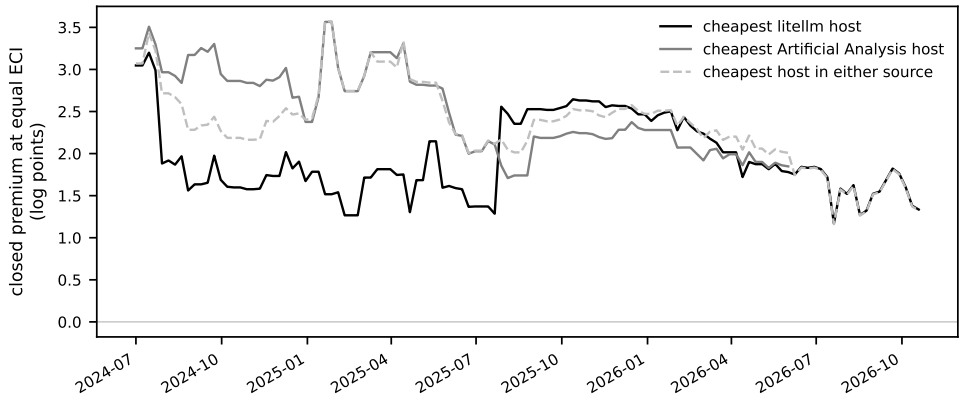

Figure F.1: The closed premium at equal ECI with the open-weight price from the cheapest litellm host, the cheapest Artificial Analysis host, and the cheaper of the two.

Table F.3: The closed premium under different treatments of suspect host listings (log points)

		Against the cheapest listing					Against the median host		
Listings dropped when	Number	Mean	2024	2026	Trend	Within 5	Mean	Trend	Within 5
none: all listings kept	0	2.01	2.21	1.88	−0.14 (0.29)	0.07	1.41	−0.15 (0.28)	0.12
ten times the other hosts' median	1,195	1.92	1.91	1.84	−0.02 (0.29)	0.30	1.42	−0.13 (0.28)	0.13
twenty times (main)	613	1.95	1.94	1.86	−0.02 (0.29)	0.29	1.43	−0.14 (0.28)	0.12
fifty times	470	2.03	2.21	1.89	−0.13 (0.29)	0.09	1.43	−0.14 (0.28)	0.12
contradicted by Artificial Analysis	1,103	1.92	2.13	1.85	−0.12 (0.28)	0.05	1.35	−0.15 (0.28)	0.12
contradicted, and Groq's Llama 3.1 405B	1,177	1.85	1.85	1.83	0.00 (0.28)	0.27	1.35	−0.15 (0.28)	0.13

Notes: Same regression as Table 3, panel A. Rows: listings more than ten, twenty, or fifty times above or below the median of the other independent hosts of the same model in the same week dropped; listings that Artificial Analysis contradicts at the same host dropped; and the same plus Groq's listing of Llama 3.1 405B. Trends are OLS slopes of the weekly premium on time in years, with standard errors from the covariance of the weekly premiums clustered by model. Within five points: the trend among models no more than five ECI points below the open floor.

Attributes. Table F.4 collects the premiums with attribute controls. Quantization and tier tags come from the listing keys and context windows from the file. Artificial Analysis reports output speed and time to the first streamed chunk per model on medium-length prompts, available for 663 model-captures in 24 captures, so these controls describe the model rather than the cheapest host. For reasoning models the time to the first chunk can include thinking time. Adding the log context window and a quantized-listing indicator as controls raises the premium against the cheapest listing to 2.44: the closed models have longer context windows, and at equal capability longer windows go with lower prices (−0.51, 0.13). Only 5 percent of the cheapest listings are quantized. Speed and latency controls also raise the premium per task, from 1.39 to 1.76: the closed models are faster, a median of 110 against 62 tokens a second, and at equal index faster models are cheaper (−0.67, 0.15).

Table F.4: The closed premium with attribute controls (log points)

Open-weight price and controls	Premium	Trend	Reading
litellm, cheapest listing (main)	1.95	−0.02 (0.29)	week effects, ECI
litellm, cheapest standard listing	1.92	−0.03 (0.29)	no quantization or tier tag
litellm, median standard host	1.41	−0.12 (0.28)	independent hosts
litellm, cheapest listing with controls	2.44	−0.26 (0.31)	log context −0.51 (0.13); quantized −1.18 (0.27)
Artificial Analysis, per token	1.80 (0.21)	–	same rows as below
with speed and latency	2.33 (0.23)	–	log speed −0.67 (0.15); log first chunk −0.14 (0.08)
Artificial Analysis, per task	1.39 (0.20)	–	same rows as below
with speed and latency	1.76 (0.22)	–	log speed −0.27 (0.16); log first chunk −0.14 (0.07)

Notes: litellm rows: the regression of Table 3, panel A, with the open price at the stated listing; the controls are common to closed and open rows. Artificial Analysis rows: the within-capture regression of Table 3, panel B, on the model-captures with speed and latency, with and without the log output speed and the log time to the first chunk, with standard errors clustered by model in parentheses.

These speeds are the model's as Artificial Analysis reports them, not the cheapest host's.

Token intensity. The gap between the two premiums widened over time: per token and per task the premium was 1.88 and 1.62 in 2025 and 1.68 and 1.11 in 2026. Across all models, a reasoning model uses 6.0 times the output tokens of a non-reasoning model of equal index, and an open-weight model 60 percent more than a closed model (standard error 0.07 log points). The two open-weight gaps differ mainly because the second holds reasoning fixed and the first does not: with the same control, the gap near the frontier is 0.55 log points (a multiple of 1.7), and without it the gap across all models is 0.73 (2.1). The per-task premium uses the gap without the control, because a buyer pays for every token a model spends on a task, reasoning or not.

Dispersion across hosts. The free alternative is not one price. Across the 2,658 open model-weeks with at least three hosts (61 models, a median of 6 hosts, with aggregators that relay a host's price excluded and the same screen applied), the most expensive host charges a median of 4.6 times the cheapest and the median host 1.9 times the cheapest. The highest price is more than twice the lowest in 78 percent of model-weeks (Figure F.8). Closed models, by contrast, sell at one price across channels. Of 11,229 listings of 88 closed models by 14 resellers, among them Azure, Bedrock, Vertex AI, and Databricks, 75 percent match the provider's own price and 94 percent are within a quarter of it, against 45 percent of open hosts' prices within a quarter of their model-week median. The closed provider sets one price, as the model assumes, while the free alternative's price is set by competing hosts,

Table F.5: The closed premium over the free alternative at equal capability: all estimates

Object	Estimate	Reading
A. Per token, hedonic premium (litellm, ECI); models no more than ten points below the open floor		
Cheapest listing: mean; 2024; 2025; 2026	1.95; 1.94; 2.02; 1.86	multiple 7.0
Cheapest listing: trend per year	−0.02 (0.29)	monthly panel −0.10 (0.27)
Median host: mean; 2024; 2025; 2026	1.43; 1.58; 1.45; 1.30	multiple 4.2
Median host: trend per year	−0.14 (0.28)	monthly −0.16 (0.26); within five points 0.12 (0.31)
At the open leader, separate slopes: mean; trend	1.84; −0.03 (0.30)	extra closed slope −0.022 (0.047)
Cheapest listing within five points: mean; trend	1.92; 0.29 (0.32)	
On Artificial Analysis host prices: mean; 2024; 2026; trend	2.51; 2.98; 2.02; −0.74 (0.37)	multiple 12.3; within five points −0.41 (0.39)
On the cheapest host in either source: mean; trend	2.34; −0.49 (0.29)	within five points −0.00 (0.32)
Flagships: mean; trend	2.84; −0.08 (0.26)	small tiers 0.32; 0.73 (0.32)
B. Per token and per task (Artificial Analysis), within capture; 24 captures, 705 model-captures		
Per token; per task	1.79 (0.20); 1.40 (0.19)	multiples 6.0 and 4.0
Output tokens per index run, closed relative to open	−0.66 (0.13)	open models use 1.9 times as many
2025; 2026 (per token / per task)	1.88/1.62; 1.68/1.11	
Cheapest closed at the open leader's index, relative to the open leader	0.66 per task; 1.14 per token	per task smaller in 88% of captures
C. Launch and legacy premiums (litellm, ECI)		
At launch: mean; median	0.73; 0.51	34 models; multiple 2.1
Over the model's life: mean; median	3.19; 3.40	3,873 model-weeks, 83 models
Cheapest closed vs cheapest hosted within five points: mean; 2026; end	0.28; 1.08; 1.02	descriptive trends: closed 0.40, hosted −0.55 a year

Notes: Log points, with standard errors clustered by model in parentheses. Trends are OLS slopes of the weekly premium on time in years, with standard errors from the covariance of the weekly premiums clustered by model; a line fitted through the weekly estimates as if they were data would give the median-host trend a standard error of 0.05. Panel A pools closed models and each open model's cheapest listing or median independent host (screened litellm prices) with ECI at least the open leader's minus ten; the regression has week effects, a common ECI slope (0.113), and a closed indicator for each week. Panel B uses the captures with token counts, published from April 2025, and within each, models of any developer no more than ten index points below the index's open leader (the four providers' closed models alone give a per-task multiple of 4.2); the cost per task is each model's input and output token counts for an index run at the provider's own input and output prices (closed) or the cheapest benchmarked host's (open).

which is why we report the premium against the median host as well as against the cheapest listing. The host-dispersion paragraph below relates the spread to host entry, host type, and listing attributes.

The price of matching the free alternative. At each capture, the cheapest closed model whose index is at least the open leader's costs 0.66 log points more per task than the open leader at its cheapest host, against 1.14 log points per token (Figure 3, right panel). The per-task gap is the smaller one in 88 percent of the 24 captures. In the litellm data the cheapest closed model within one ECI point of the open floor fell from $10.50 to $0.50 per million output tokens, but its fitted trend is 0.03 log points a year, so the endpoint fall reflects where the sample starts, and the series is a sawtooth around a flat line, rising when a major open release overtakes the cheapest matching closed model and falling when a new closed version is launched at the floor.

Launch premiums and legacy premiums. A closed model launched when an open model of at least its capability was already hosted carries a premium of 0.73 log points over that open model at launch (median 0.51, 34 models), a multiple of 2.1. Over the life of a model the premium over the cheapest open model of at least its capability averages 3.19 log points (median 3.40, 3,873 model-weeks on 83 models): list prices of continuing versions do not follow the free alternative down, the legacy premium the model predicts (P1, P2).

Launch prices and the two frontiers. If the adjustment through versions answers the free alternative, a version launched soon after the open leader improves should be priced lower at given capability. The levels of the open leader and the closed frontier move together, but their gains over a window do not (correlation -0.11 over ninety days and 0.23 over a year across the launches), so we add both gains over the window before the release to the launch schedule, together with provider, tier, and reasoning indicators (Table F.6). Over ninety days, a point of open-leader gain changes the launch price by 0.003 log points (0.058; permutation $p = 0.96$) and a point of closed-frontier gain by -0.077 (0.067). Over a year the open gain has no effect (0.040, 0.036). Gains of Chinese open models, whose releases are least likely to be timed against US list prices, give -0.015 (0.057; permutation $p = 0.80$),

and the open leader's gain in the ninety days after a launch, a placebo, has no detectable effect (0.051, 0.042). The open gain's point estimate is small, a rise of about 1 percent of the launch price for a one-standard-deviation gain over ninety days, 2.4 ECI points, and imprecise, with a 95 percent interval from a fall of 23 percent to a rise of 32 percent, so the pooled launches cannot separate the free alternative's effect on launch prices from the trend and from the closed rivals.

Launch prices by position at launch. Proposition B.2 sharpens the test. If legacy prices are sticky, the free alternative reaches the launch prices of versions launched next to it and not those of versions launched above the provider's legacy versions. If launches replace the line, every launch price responds. We therefore interact both gains with the launch's position, its lead over the open floor at the release date, below five ECI points (53 launches, those below the floor included) or five or more (34), and keep the other coefficients common (Table F.6, panel B). The launches near the floor do not respond more. Over ninety days a point of open-leader gain changes their launch price by 0.023 log points (0.066; permutation $p = 0.73$), against -0.082 (0.067; permutation $p = 0.22$) for the launches well above the floor. The difference, 0.105 (0.090; $p = 0.24$), has the wrong sign, and over half a year or a year neither group responds. Two other definitions of position give the same answer. The 29 launches with none of the provider's versions between them and the floor, which Proposition B.2(b) prices against the free alternative, respond no more than the other 58 (0.018, 0.080, against -0.051, 0.075). The 46 new top versions, which Proposition B.2(a) prices off the old top, respond more in point estimate, -0.071 (0.045; permutation $p = 0.14$) over ninety days against 0.035 (0.078) for the 41 versions launched below the provider's top, but none of the 18 group coefficients, over three definitions and three windows, has p below 0.05, and the new-top estimate, the closest to it, vanishes over longer windows. The split does show where the closed frontier matters. Launches well above the floor are priced lower after gains of the closed frontier at every window, by -0.136, -0.141, and -0.155 log points per point over ninety days, half a year, and a year (standard errors 0.073, 0.059, and 0.039), while launches near the floor do not respond to them (-0.038 over ninety days, 0.110). At the bottom, the data do not show the free alternative reaching launch prices, as

Proposition B.2(b) predicts for sticky legacy prices. The interval for the launches near the floor is wide, so they cannot rule it out either. The response of the new tops over ninety days, −0.190 (0.058) to the closed frontier and −0.071 to the open leader, is not what sticky legacy prices predict, which would have new tops respond to neither. It fits replacement or a reset at launch, more clearly for the closed frontier than for the open leader.

Frontier launches and the two levels. Table F.18, panel B, relates the launch prices of the 34 frontier launches to the levels of the open floor and of the rivals' best closed model rather than to their recent gains. Both coefficients are negative, −0.18 (0.08) and −0.16 (0.08) log points per ECI point, but the two levels move almost together across launches (correlation 0.97), so the regression cannot tell them apart. The open-floor coefficient also rests on the launches that followed the two largest open jumps, DeepSeek-R1 and Kimi K3. Without the launches in the 120 days after them it is −0.05 (0.18) on 24 launches. The launch evidence above therefore uses gains over a window, which do not move together.

Launches as repricing opportunities. Propositions B.3 and B.4 distinguish three regimes for a new top. In coexistence it is priced off its predecessor's sticky price and reflects the free alternative's quality when that price was set. Under replacement and under a reset of the line it reflects the current quality, and only a reset cuts the predecessor's posted price. We check them in four ways (Table F.7). The first relates each of the 46 new tops to its predecessor, the provider's best version listed in the week before the launch, using the gains of the open leader and of the rivals' best closed model since the predecessor's price was set, a proxy for the line's last common reset. With a log price ratio, which fixes the coefficient on the predecessor's log price at one, a point of open gain changes the new top's price by −0.090 log points (0.058; permutation $p = 0.16$), short of significance. The data do not support that restriction. Estimated freely, the coefficient on the predecessor's log price is 0.068 (0.182), and the open gain's coefficient falls to 0.005 (0.060; $p = 0.94$). In levels, the spread between the new top's price and its predecessor's changes by −4.29 dollars per million output tokens per point of open gain (1.68; $p = 0.02$), but it also falls with the predecessor's price (−0.86, 0.17), which the model does not predict, and with that price held fixed the

gain gives −1.07 (1.33; $p = 0.46$). The response to the closed frontier's recent gains survives the control. Given the predecessor's price, a point of closed-frontier gain over the ninety days before the launch changes the new top's price by −0.125 log points (0.053; $p = 0.036$), which coexistence does not predict, while a point of open-leader gain changes it by −0.037 (0.039; $p = 0.37$), not distinguishable from zero.

The second check asks whether the new top is anchored on its predecessor's price. Given its own ECI, the step, and a trend, the predecessor's price adds nothing (0.065 in logs, 0.118; 0.09 in levels, 0.18). By Proposition B.3(b) this does not separate the regimes, since a predecessor whose price differs because its serving cost differs passes nothing through in either. The earlier and the current quality of the free alternative, which do separate them, cannot be told apart from the trend in levels, with which the open floor correlates at 0.98 across these launches.

The third check is how any reset is made. The 46 new tops came on 39 launch dates and had 39 distinct predecessors. One of them, Claude 3.5 Sonnet, was released 11 days before the panel begins, so the week after its launch is not observed, and its predecessor, Claude 3 Opus, kept its list price throughout the panel. The predecessor's posted price was cut within a week of the launch in only 1 of the 46 cases, and in 28 cases, 22 distinct versions, the predecessor was later retired, a median of 250 days after the launch, so whatever reset occurs is made by replacement rather than by posted cuts, although the late retirements do not show that the predecessors lost their buyers at the launch. In terms of Proposition B.6, posted cuts at launches are rare when repricing a predecessor costs more than the loss from its stale price or than its value to the line, and retirements that come months after a launch fit predecessors that stay listed while withdrawing them costs more than listing them, until contracts end or costs change. Posted cuts at launches do occur. Of the 8 provider cuts, 2 came on the day the same provider launched a new version, o1-mini's with o3-mini and o3's with o3-pro, against 0.18 expected if cuts were timed at random (randomization $p = 0.036$, for a window we chose after seeing the dates). Within a week of a launch there are 3 cuts against 2.5 expected ($p = 0.49$), because the providers launch so often that 32 percent of days lie within a week of one of their launches. The 5 cuts more than a week from any launch followed open releases that closed 38 percent of the gap to the best closed model on

average, against 33 percent for the 3 cuts at launches, a small difference in the direction of Proposition B.4(c), and one of them, Claude 3.5 Haiku's, followed a release that did not raise the open floor.

The fourth check splits the closed frontier's gain into the gain of the rivals' best closed model and the gain of the provider's own. Over ninety days, what response there is for launches well above the floor runs through the provider's own frontier gains (-0.069, 0.052), not the rivals' (0.002, 0.047), and over half a year the own gain fades (-0.041, 0.042) while the rivals' comes closest to mattering (-0.070, 0.043). A launch soon after the provider's own previous frontier launch inherits the level reset at that launch, as Proposition B.4 implies when lines are reset at launches. The closed rivals of Remark B.2 show, if at all, only over longer windows.

The launch evidence therefore does not support launches priced off the version's position in the line, and new tops respond to the closed frontier's recent gains even given their predecessors' prices, a response the model does not speak to, while their response to the open leader's recent gains, which replacement or a reset predicts, is not detectable once the predecessor's price is held fixed. But the data separate these regimes from coexistence only in part, the posted prices show only that whatever reset occurs is not made by cuts, and the comparisons with predecessors were designed after the split and rest on 46 new tops.

The launch schedule per task. The launch schedule of Section 4 is the same per task. For the 59 of these launches that match Artificial Analysis token counts, we add to the log launch price the model's log output tokens per index run relative to the median model in the same capture. The log launch price per task rises by 0.294 per ECI point (0.032) and changes by -3.40 log points a year (0.56), against 0.239 and -3.10 per token for the same launches, and the lead over the open floor carries 0.275 (0.057) against 0.211 per token. At given capability newer models use somewhat fewer tokens per task, though the estimate is imprecise (-0.30 log points a year, 0.58), so per-task launch prices fall a little faster than per-token ones.

Construction of Table F.6. Table F.6 reports the launch regressions of the paragraphs above. The sample is the launch schedule's: a closed model's first listing within forty-five days of its release (87 launches). Each regression of the log list price per output token at launch holds own ECI and a linear trend fixed, and the rows with controls add provider, small-tier, pro-tier, and reasoning indicators. The gains are the changes in the ECI of the open leader, of the best open model from Chinese developers or from elsewhere, and of the best closed model, the last measured the day before the release, over the window before the release. The one-year column, where the windows of most launches overlap, has two coefficients the text does not use. The gain of the best non-Chinese open model raises launch prices, the wrong sign for the free alternative, and the closed frontier's gain lowers them, as it does at every window for launches well above the floor (panel B). Neither appears at the shorter windows, and the open leader's gain has no effect at any window, so we do not read the one-year rows as evidence. The placebo row uses the open leader's gain over the window after the release, on the launches whose window ends inside the ECI data. Standard errors are heteroskedasticity-robust, and the permutation p-values reshuffle the open gain across launches within the launch year, 5,000 times. Panel B interacts the open and closed gains with the launch's position and keeps the other coefficients common. The primary position is the lead over the open floor at the release date, below five ECI points or five or more. The lead at the first listing classifies every launch the same way. The alternatives are adjacency in the provider's line, meaning that no version the provider listed in the last panel week before the release lies between the floor and the launch, and whether the launch is above every version the provider listed that week, a new top of its line. The permutation p-value for a group reshuffles the open gain among that group's launches within the launch year. The separate regressions estimate every coefficient within each lead group, and the placebo row adds the open gain after the launch, interacted with the lead group.

Construction of Table F.7. Table F.7 reports the checks of the paragraphs above. Panel A counts the verified provider cuts that came within a given number of days of a launch by the same provider, the release date, as Epoch dates it, of one of its versions in the price panel, and compares the count with randomization that redraws the date of each cut decision

Table F.6: Launch prices and recent gains of the open leader and the closed frontier

Gain over the window before the release (ECI points)	90 days	180 days	365 days
A. All launches			
Open-leader gain, no controls	−0.066 (0.072); $p_{\text{perm}} = 0.37$	−0.062 (0.047); $p_{\text{perm}} = 0.21$	−0.001 (0.047); $p_{\text{perm}} = 0.98$
Open-leader gain, with controls	0.003 (0.058); $p_{\text{perm}} = 0.96$	−0.003 (0.042); $p_{\text{perm}} = 0.95$	0.040 (0.036); $p_{\text{perm}} = 0.24$
Closed-frontier gain, with controls	−0.077 (0.067)	−0.041 (0.052)	−0.076 (0.035)
Gain of the best Chinese open model, with controls	−0.015 (0.057); $p_{\text{perm}} = 0.80$	0.005 (0.046); $p_{\text{perm}} = 0.91$	−0.019 (0.018); $p_{\text{perm}} = 0.35$
Gain of the best non-Chinese open model, with controls	−0.001 (0.040)	−0.010 (0.031)	0.136 (0.053)
Placebo: open gain after the launch	0.051 (0.042)	0.041 (0.057)	−0.129 (0.217)
Launches (placebo row)	87 (71)	87 (61)	87 (43)
Correlation of the open and closed gains	−0.11	0.25	0.23
B. By position at launch: gains interacted with the position, controls shared			
Open-leader gain, launched near the floor (lead below 5)	0.023 (0.066); $p_{\text{perm}} = 0.73$	−0.015 (0.063); $p_{\text{perm}} = 0.82$	0.041 (0.049); $p_{\text{perm}} = 0.35$
Open-leader gain, launched well above the floor (5 or more)	−0.082 (0.067); $p_{\text{perm}} = 0.22$	0.008 (0.037); $p_{\text{perm}} = 0.79$	0.034 (0.037); $p_{\text{perm}} = 0.38$
Difference, near less well above	0.105 (0.090); $p = 0.24$	−0.024 (0.068); $p = 0.73$	0.007 (0.053); $p = 0.89$
Closed-frontier gain, launched near the floor	−0.038 (0.110)	0.010 (0.068)	−0.020 (0.043)
Closed-frontier gain, launched well above the floor	−0.136 (0.073)	−0.141 (0.059)	−0.155 (0.039)
Open-leader gain, adjacent to the floor in the provider's line	0.018 (0.080); $p_{\text{perm}} = 0.84$	−0.041 (0.062); $p_{\text{perm}} = 0.54$	0.019 (0.065); $p_{\text{perm}} = 0.73$
Open-leader gain, provider's versions between it and the floor	−0.051 (0.075); $p_{\text{perm}} = 0.50$	0.021 (0.053); $p_{\text{perm}} = 0.70$	0.040 (0.035); $p_{\text{perm}} = 0.23$
Open-leader gain, launched below the provider's top	0.035 (0.078); $p_{\text{perm}} = 0.67$	−0.018 (0.067); $p_{\text{perm}} = 0.80$	0.078 (0.063); $p_{\text{perm}} = 0.16$
Open-leader gain, a new top of the provider's line	−0.071 (0.045); $p_{\text{perm}} = 0.14$	0.017 (0.039); $p_{\text{perm}} = 0.67$	0.011 (0.034); $p_{\text{perm}} = 0.72$
Closed-frontier gain, a new top of the provider's line	−0.190 (0.058)	−0.104 (0.052)	−0.063 (0.039)
Separate regressions, open-leader gain: near; well above	0.020 (0.069); −0.086 (0.048)	−0.028 (0.064); −0.020 (0.036)	0.050 (0.052); 0.004 (0.023)
Placebo, open gain after the launch: near; well above	0.055 (0.076); 0.017 (0.052)	−0.034 (0.089); 0.088 (0.052)	−0.109 (0.231); −0.115 (0.217)
Launches: near, well above; adjacent, not; below the top, new top		53, 34; 29, 58; 41, 46	

Notes: Coefficients on the gain in ECI points over the window before the release, in log points of the launch price per output token, with heteroskedasticity-robust standard errors in parentheses and permutation p-values (p_{perm}). Every row holds own ECI and a linear trend fixed; controls are provider, small-tier, pro-tier, and reasoning indicators. Panel B: near the floor is a lead over the open floor below five ECI points at the release date, below the floor included. Adjacent means that none of the provider's listed versions lies between the floor and the launch, and a new top is above every version the provider listed in the week before the release. Rows in panel B include the controls, and the permutation p-values reshuffle the open gain within the group and the launch year.

uniformly over the panel's days, keeping its provider, 10,000 times. A version's first week in the file does not count as a launch, since the file's dates are not the provider's. As in the timing test, the two cuts OpenAI made on one day move together, so the 8 cuts are redrawn as 7 decisions, but over days rather than weeks, since the windows are counted in days. Panel B relates a new top's price at its first listing to its predecessor's price in the week before the release, the ECI step, the years since the predecessor's price was set (its release or a later provider cut), and the gains of the open leader and of the rivals' best closed model, the best closed model of the other three providers, since then, with indicators for the pro tier of both versions, the small tier and reasoning of the new top, and the provider. The first column uses the log price ratio, which fixes the coefficient on the predecessor's log price at one. The next two estimate that coefficient, with and without its square. The level columns use the spread between the two prices, without and with the predecessor's price as a control, and the last column replaces the accumulated gains by those over the ninety days before the launch. Without pro tiers, the accumulated open gain with the predecessor's

Table F.7: Launches as repricing opportunities

A. Provider cuts near a launch by the same provider				
Days between the cut and the launch, at most	0	3	7	14
Cuts near a launch, of 8	2	2	3	3
Expected if cuts were timed at random	0.18	1.23	2.54	4.39
Randomization p, one-sided	0.036	0.350	0.488	0.888

B. New tops against their predecessors	Log ratio	$\log p_J$ free	$+(\log p_J)^2$	Spread	Spread, p_J held	90 days, $\log p_J$ free
Open-leader gain	−0.090	0.005	0.001	−4.292	−1.069	−0.037
	(0.058)	(0.060)	(0.063)	(1.684)	(1.335)	(0.039)
permutation p	[0.16]	[0.94]	[0.99]	[0.02]	[0.46]	[0.37]
Rivals' gain (last column: closed frontier)	−0.101	−0.016	−0.014	−0.643	−0.247	−0.125
	(0.053)	(0.041)	(0.041)	(2.030)	(1.377)	(0.053)
Predecessor's price		0.068	−0.018		−0.862	0.075
		(0.182)	(0.244)		(0.172)	(0.145)
ECI step from the predecessor	0.240	0.094	0.089	3.711	2.262	0.084
	(0.059)	(0.051)	(0.055)	(1.701)	(1.368)	(0.042)
New tops; R^2	46; 0.77	46; 0.86	46; 0.86	46; 0.80	46; 0.89	46; 0.88

C. Anchoring: the new top's price on its predecessor's, given its own ECI	Logs	Levels	Levels, without pro tiers
Predecessor's price	0.065 (0.118)	0.095 (0.182)	−0.018 (0.262)
Own ECI	0.024 (0.059)	0.050 (1.850)	−0.369 (1.803)
ECI step from the predecessor	0.090 (0.052)	2.134 (1.529)	−0.231 (1.375)
Trend (years)	−0.282 (0.953)	−0.010 (26.287)	0.133 (25.245)
New tops; R^2	46; 0.86	46; 0.82	36; 0.46

D. The closed gain split into the rivals' and the provider's own (near; well above)			
Gain over the window before the release	90 days	180 days	365 days
Open leader	0.016 (0.073); −0.045 (0.060)	0.013 (0.062); 0.004 (0.037)	0.015 (0.044); 0.030 (0.033)
Rivals' best closed model	−0.018 (0.065); 0.002 (0.047)	−0.057 (0.053); −0.070 (0.043)	0.059 (0.069); −0.074 (0.039)
Provider's own best model	−0.065 (0.050); −0.069 (0.052)	−0.067 (0.047); −0.041 (0.042)	−0.058 (0.031); −0.098 (0.026)

Notes: Panels B to D: coefficients with heteroskedasticity-robust standard errors in parentheses, in log points per ECI point in the log columns and in dollars per million output tokens per ECI point in the level columns. In panel B, permutation p-values are in brackets. Panels B and C: a new top is above every version its provider listed in the week before the release, and its predecessor is the best of those versions. Panel D: launches near the floor and launches well above it, with the controls of Table F.6.

log price free gives −0.001 (0.058; $p = 0.99$). Panel C regresses the new top's price on its predecessor's, its own ECI, the step, and a trend, with the same controls. Panel D repeats the split of Table F.6 with the closed frontier's gain divided into the gain of the rivals' best closed model and the gain of the provider's own, both measured the day before the release. Permutation p-values reshuffle a gain within the launch year, and within the group in panel D, 5,000 times.

Table F.8: Primary tests of the four claims about cuts (Section 6.2)

Claim	Primary test	Estimate; p-value
Cuts follow frontier open releases	Weekly cut rate in the eight weeks after each of the 25 frontier open releases minus the eight weeks before; dates of the cut decisions redrawn	0.042 decisions a week; two-sided $p = 0.096$ (cuts 0.060, $p = 0.054$)
Cuts fall near the open floor	Post-minus-pre contrast in the weekly cut probability of models zero to five ECI points above the floor, 11 major releases; dates of the cut decisions redrawn	0.0092; one-sided $p = 0.034$; adjusted 0.10 (three positions), 0.16 (twelve contrasts)
Bigger steps bring more cuts	Slope of the cuts credited to each frontier release on a proxy for the loss in Proposition 3; proxy permuted across releases	0.066; one-sided $p = 0.044$
The versions cut are young	Mean percentile of a cut model's age among the closed models listed in its week (one half if age does not matter)	0.14; one-sided $p = 0.0001$

Notes: Adjusted p-values are Romano–Wolf stepdown over the three positions of the main definition and over the twelve contrasts of three positions under four definitions of position (Table F.12). Every other test of these four claims is secondary.

Primary tests of the claims about cuts. Section 6.2 makes four claims about cuts, that they follow open releases at the frontier, that they fall near the open floor, that bigger steps bring more cuts, and that the versions cut are young. For each claim we name one primary test, the one that maps most directly into Propositions 3 and B.1, and treat every other test as secondary (Table F.8). The secondary tests show how much the primary result depends on the choice of sample, window, and definition, and we do not count them as further evidence for these claims. They are reported below. The tests of Table 5, which ask whether other forces could produce the cuts, are not secondary tests of these claims, and each has its own null. For the location claim, where we examined several positions and definitions of position, we also report p-values adjusted for that search.

Where the cuts fall, how they scale, and which versions they hit. Where do the cuts fall? In the model-week panel we interact the event-time bins with the model's position: below the open floor, zero to five ECI points above it, and five or more above. The post-minus-pre contrast in the weekly cut probability is 0.0092 (0.0054) for models just above the floor, against 0.0009 (0.0029) below it and 0.0003 (0.0023) far above it. The

contrast for models just above the floor is the primary test of the location claim. Keeping the release calendar and redrawing the date of each cut decision 2,000 times over the weeks in which its versions were at risk gives a one-sided p of 0.034. Redrawing the release weeks instead gives 0.030, and the 132 circular shifts of the release calendar give 0.008. Because we examined three positions under four definitions of position, we adjust it for that search with the stepdown procedure of Romano and Wolf (2005) on the same redraws, which gives 0.10 over the three positions and 0.16 over all twelve contrasts (Table F.12). The location of the cuts is therefore suggestive rather than established. A stacked comparison that credits each fall in a model's price to the nearest release before it, the discrete form of a regression discontinuity in time, finds 3 cuts just above the new floor, 3 among the many models below it, and 1 far above it. Read against exposure in Figure 4, versions zero to five points above the floor account for 14 percent of version-weeks but 3 of the 8 cuts in the week of the cut, and versions below it for 79 percent of version-weeks and 4 cuts.

The eight cuts one by one. Table F.9 lists the 8 cuts case by case, with their positions against both floors, their ages, the provider's lower versions, the open releases before them, and the rival and own launches around them; Section 6.2 reads it, and the placebo on closed launches near the open floor is in Table 5. Table F.10 gives what each provider said about its cut.

The versions cut are young. The remaining-life mechanism of Proposition B.1 suggests that cuts fall on young versions if the life a version has left falls with its age, and it does: in the hazards of Section 6.3, a year of age multiplies the retirement hazard by 3.7. The 8 provider cuts hit models a median of 1.6 months after their launch, against a median of 8.0 months for the listed models. On average a cut model was younger than 86 percent of the closed models listed in its week (simulation $p = 0.0001$). This is the primary test of the age claim. The cut versions are young also when compared only with versions at the same distance from the floor (mean percentile 0.22, $p = 0.0008$; Table F.14). Experimentation with prices after launch would also bring early cuts, but it would not make them follow open releases. The proposition concerns whole lines with a common remaining life, so the youth

Table F.9: The eight provider cuts, one by one

Date	Version	Cut (%)	Lead over the floor: ECI; price-adjusted	Age (months)	Below own top	Lower active versions that kept their price	Nearest frontier open release before	Rivals' launches within five points, four weeks either side	Own launches, four weeks either side
2024-08-12	Gemini 1.5 Flash (May) (Google)	−71	−6.1; −2.6	2.7	4.3	none listed	Llama 3.1-405B, 20 days before, 3.5 points closed	GPT-4o mini (OpenAI), 25 days before	none
2024-10-01	Gemini 1.5 Pro (Sept) (Google)	−52	+2.8; +4.2	0.2	0.0	Gemini 1.5 Flash (Sep 2024) (other tier)	Qwen2.5-72B, 12 days before, 0.2 points closed	4: nearest o1-mini (OpenAI), 19 days before	2: nearest Gemini 1.5 Pro (Sept), 7 days before
2024-12-03	Claude 3.5 Haiku (Anthropic)	−20	−1.9; +1.1	1.4	6.4	none of 5	Mistral Large 2 (Nov), 15 days before, 0.0 points closed	2: nearest Grok-2 (Dec 2024) (xAI), 9 days after	none
2025-01-31	o1-mini (OpenAI)	−63	−3.1; +3.4	4.6	6.0	GPT-4o mini	DeepSeek-R1, 11 days before, 6.5 points closed	4: nearest Gemini 2.0 Pro (Google), 5 days after	2: nearest o3-mini, same day
2025-06-10	o3 (OpenAI)	−80	+5.6; +5.6	1.8	0.0	o4-mini (other tier)	DeepSeek-R1 (May), 13 days before, 2.0 points closed	2: nearest Gemini 2.5 Pro (Jun) (Google), 5 days before	o3-pro, same day
2026-05-06	Grok 4.20 (xAI)	−58	+1.2; +1.2	2.6	0.0	none of 5	Kimi K2.6, 16 days before, 1.1 points closed	5: nearest Gemini 3.5 Flash (Google), 13 days after	Grok 4.3 Beta, 19 days before
2026-07-30	GPT-5.6 Terra (OpenAI)	−20	+2.2; +5.3	0.7	2.8	GPT-5.3 Codex; GPT-5.6 Luna cut the same day	Kimi K3, 14 days before, 5.9 points closed	4: nearest Qwen 3.8 Max (Alibaba), 3 days after	3: nearest GPT-5.6 Sol, 21 days before
2026-07-30	GPT-5.6 Luna (OpenAI)	−80	−1.4; +1.7	0.7	6.4	none of 25	Kimi K3, 14 days before, 5.9 points closed	7: nearest Qwen 3.8 Max (Alibaba), 3 days after	3: nearest GPT-5.6 Sol, 21 days before

Notes: The verified provider cuts, dated by the providers' announcements and pricing pages (o1-mini's by the file's commit), with the cut in percent of the output price. Lead over the floor: the version's ECI minus the open leader's in the week of the cut, and minus the best open model hosted at no more than the version's own list price (the price-adjusted floor). Below own top: ECI points below the provider's best listed version that week. Lower active versions: the provider's lower versions listed that week that were above the price-adjusted floor and not displaced by an own version at least as capable and no dearer; those that kept their price are named, with "other tier" where the tier class (small, standard, or pro, from the model's name) differs from the cut version's. Nearest frontier open release: the latest release no more than one ECI point below the prior open leader's in the sixty days before the cut, and the points it closed on the best closed model, the rise of the open floor, which is zero for a release below the prior leader. Rivals' launches: closed versions of other developers in Epoch's index launched within five ECI points of the cut version in the four weeks before or after the cut, with the one nearest in time. Own launches: the release dates, as Epoch dates them, of the provider's versions in the price panel, in the same window, the rule of Table F.7.

Table F.10: What the providers said about their price cuts

Decision	Cut	Provider's source	Reason given, paraphrased	Class
2024-08-12	Gemini 1.5 Flash (Google)	Google Developers Blog, 2024-08-08	to make its most used high-volume model more affordable	affordability
2024-10-01	Gemini 1.5 Pro (Google)	Google Developers Blog, 2024-09-24	announced with updated models and no reason beyond cheaper building with Gemini	none stated
2024-12-03	Claude 3.5 Haiku (Anthropic)	Anthropic news, 2024-12-03	to make the model accessible for more uses; announced beside faster inference on AWS Trainium2 but not tied to it	affordability
2025-01-31	o1-mini (OpenAI)	–	no announcement found; the cut came with the launch of o3-mini and the new price appears only on the undated model page	no announcement found
2025-06-10	o3 (OpenAI)	OpenAI Developer Community (announcement), 2025-06-10	an optimized inference stack serving the same model	own serving cost
2026-05-06	Grok 4.20 (xAI)	–	no announcement found; the new price appears on xAI's API page when Grok 4.3 is listed at the same price	no announcement found
2026-07-30	GPT-5.6 Terra and Luna (OpenAI)	OpenAI, 2026-07-30	gains in serving efficiency passed on to customers; serving cost down 20 percent	own serving cost

Notes: One row per cut decision, a provider and a date. Each source is the provider's own announcement, read in November 2026. Own serving cost means that the provider attributed the cut to lower costs of serving the model. The statements rest on private information that outsiders cannot check and may be selective, so they can bias as well as inform (Milgrom, 1981). A provider also has reason to present a cut as an efficiency gain, since a cut on a listed version can be read as a sign that the version is losing ground (Section 5). The class records what the provider said, not why it cut.

of the versions cut is consistent with its mechanism rather than a test of it.

Do bigger steps bring more cuts? Proposition 3 says that the loss from leaving prices unchanged grows with the square of the free alternative's improvement and falls with the gap that remains, so steps that bring the open leader closer to the closed models should bring more cuts. The three releases that closed most of the gap to the best closed model, Llama 3.1 405B, DeepSeek-R1, and Kimi K3 (74, 69, and 52 percent), were each followed within 20 days by cuts of 20 to 80 percent (Table F.16). The primary test regresses the cuts credited to each of the 21 frontier releases, each cut credited to the nearest release before

it, on a proxy for that loss, points closed squared over the points by which the open leader still trails the best closed model. The proposition's loss uses the lead of the lowest active version of each line and the improvement accumulated since its price was last set, neither of which the proxy measures. The slope is 0.066 with a one-sided permutation p of 0.044. The secondary tests agree in sign, but the evidence rests on three releases.

Secondary tests of the location claim. With week fixed effects, the difference between the near and far positions is 0.0094 (0.0063), and the adjustment of Holm (1979) for the search over positions gives 0.10 over the three positions and 0.37 over all twelve contrasts. Of the 7 cut decisions, 2 hit a model within three ECI points of a major open release in the sixty days before, against 13 percent of model-weeks so exposed (one-sided binomial $p = 0.24$). These estimates are imprecise, but their pattern fits Proposition 3. Models far above the floor are not repriced more after the releases, and cuts fall just above the floor more often than its share of model-weeks would have it, although 4 of the 8 cuts fell on versions below the floor. For major releases the pattern survives other definitions of position and leaving out any one release, but it rests on a few cuts (Table F.11). For major releases, the contrast for models just above the floor is the largest under every partition we try (zero to three, zero to five, or zero to ten points above the floor, or a one-point tolerance below it). Its one-sided randomization p-values are between 0.012 and 0.050, and between 0.10 and 0.21 after the adjustment for twelve contrasts (Table F.12). Leaving out one major release at a time, the contrast for models just above the floor lies between 0.0068 and 0.0106 (t between 1.3 and 1.8) and is the largest of the three in all 11 cases. It rests on two dates, however: without the cut that followed GLM-5.1 it falls to 0.0049, and without the two cuts that followed Kimi K3 to 0.0065. With all frontier releases instead of the major ones, the location is not identified: the contrast is 0.0033 (0.0037) just above the floor and 0.0136 (0.0132) far above it. Across distance windows of three to ten ECI points and time windows of thirty to ninety days, the exposure test finds 2 to 6 of the 7 cut decisions on close substitutes of a recent major release, with one-sided p-values between 0.0015 and 0.24.

The price-adjusted floor. The positions of Table F.11 measure a version's lead over the open leader by capability alone. The price-adjusted floor of a version in a week is the highest ECI among open-weight models hosted that week at an output price no higher than the version's own list price, on the weekly hosted panel with batch tiers and file errors removed. A version with no cheaper hosted open model counts as far above it. Under this definition 68 percent of model-weeks are below the floor, 21 percent zero to five points above it, and 11 percent far above, against 79, 14, and 7 percent by capability alone, and 90 percent of model-weeks have the same position under both. The post-minus-pre contrasts after major releases are 0.0007 (0.0007) below the price-adjusted floor, 0.0076 (0.0098) zero to five points above it, and 0.0043 (0.0039) far above, with one-sided randomization $p = 0.092$ for the near contrast on 2,000 redraws of the dates of the cut decisions (0.021 with the release weeks redrawn instead). The 8 cuts fall 1, 5, and 2 by position. This check was added after the multiplicity adjustment of Table F.12 and is reported beside it, not inside its family.

Price changes instead of cut counts. These are secondary tests of the timing and location claims. Counting cuts ignores their size. With the weekly change in the log price as the outcome, the same regression gives an eight-week effect of −0.055 log points (0.034) for models just above the floor after major releases (one-sided randomization $p = 0.10$, with the release weeks redrawn), against −0.026 (0.015) below it and 0.004 (0.029) far above it. After all frontier releases the effect just above the floor is 0.033 (0.038; $p = 0.88$). A stacked comparison, the discrete form of a regression discontinuity in time, asks the same question without event-time bins. For each major release it takes the closed models listed in the release week and eight weeks later and compares the change in their log price over those eight weeks with the same change after placebo dates drawn from the panel. Because list prices are sticky, the price in the release week is the price just before the release, so no pre-period is needed. Because the windows of nearby releases overlap, a cut can fall in several of them: o1-mini's falls in the windows of Phi-4, DeepSeek-V3, and DeepSeek-R1, where it would count once in each position. We therefore credit each fall in a model's price to the nearest release before it and apply the same rule to the placebo dates. Over 311 model-release pairs the mean change is then −0.018 log points (randomization $p = 0.040$),

and 2.2 percent of the models are cut ($p = 0.028$), 7 cuts in all. Classified by position after the release, the lead that Proposition 3 refers to, 8 percent of the 36 models just above the new floor are cut ($p = 0.010$), against 1 percent of the 262 models below it ($p = 0.180$) and 1 of the 13 far above it ($p = 0.095$). The cut far above the floor is o3's, by four fifths two weeks after the May 2025 version of DeepSeek-R1. Counting o1-mini's cut in every window instead, the share cut far above the floor would be 15 percent ($p = 0.05$). Classified by position before the release, the models just above the old floor are cut no more often than after placebo dates (4 percent, $p = 0.26$), which fits cuts that follow where the release moves the floor. The evidence again rests on a few cuts. Kimi K3 shows both sides of the prediction. It brought two GPT-5.6 versions, Sol and Terra, within five points of the floor, and OpenAI cut them by 33 and 20 percent (Sol's a temporary promotion, which the cut counts leave out), but it also cut GPT-5.6 Luna, which Kimi K3 overtook in capability, by 80 percent. Luna was not dominated in price, however: the cheapest host of Kimi K3 charged $15.00 per million output tokens in its first week in the hosted data and $12.50 at the end of the sample, against Luna's $6.00 before the cut, and Proposition B.5 rules out a profitable cut only on a version that an open model dominates in both capability and price and that costs at least the open model's hosted price to serve.

Where cuts fall. Table F.11 reports the post-minus-pre contrast in the weekly cut probability by position under four partitions, for major and for all frontier releases, with randomization p-values for the group just above the floor (1,000 redraws of the dates of the cut decisions, with the release calendar kept). Its major-release rows repeat the contrasts of Table F.12 on a sample that keeps the weeks after a retirement announcement, and the text cites Table F.12.

Multiple testing. Table F.12 reports the twelve location contrasts, three positions under four definitions of position, on the sample of Table 4, panel B, with one-sided randomization p-values from 2,000 redraws of the dates of the cut decisions, with the release calendar kept, and two adjustments: the Romano–Wolf stepdown, with each contrast studentized by its randomization standard deviation, and Holm's. Only the contrasts for models just above

Table F.11: Where cuts fall: contrast in the weekly cut probability by position

Releases	Partition (ECI points from the open floor)	Below	Just above	Far above	Randomization p
Major	Main: below 0; near 0–5; far 5+	0.0008 (0.0025)	0.0093 (0.0054)	0.0001 (0.0023)	0.035
Major	Near 0–3; far 3+	0.0008 (0.0025)	0.0123 (0.0072)	0.0002 (0.0013)	0.016
Major	Near 0–10; far 10+	0.0008 (0.0025)	0.0068 (0.0038)	−0.0016 (0.0016)	0.035
Major	Below −1; near −1 to 5; far 5+	0.0007 (0.0026)	0.0074 (0.0042)	0.0001 (0.0023)	0.035
Frontier	Main: below 0; near 0–5; far 5+	0.0009 (0.0009)	0.0033 (0.0037)	0.0136 (0.0132)	0.156
Frontier	Near 0–3; far 3+	0.0009 (0.0009)	0.0043 (0.0049)	0.0075 (0.0071)	0.134
Frontier	Near 0–10; far 10+	0.0009 (0.0009)	0.0055 (0.0037)	−0.0002 (0.0004)	0.042
Frontier	Below −1; near −1 to 5; far 5+	0.0009 (0.0010)	0.0026 (0.0030)	0.0136 (0.0132)	0.188

Notes: Same regression as Table 4, panel B: model-weeks, event-time bins interacted with position, position, provider and tier indicators, age, and a trend, with standard errors clustered by week. The randomization p-value is one-sided for the group just above the floor. Unlike Table 4, the sample keeps the model-weeks after a retirement announcement, so the main row differs slightly from it. Table F.12 uses the sample of Table 4.

Table F.12: Location contrasts with multiplicity-adjusted p-values

				Romano–Wolf		Holm
Definition of position	Position	Contrast (s.e.)	One-sided p	Three positions	Twelve contrasts	Twelve contrasts
main (below 0; near 0–5; far 5+)	below	0.0009 (0.0029)	0.412	0.582	0.774	1.000
main (below 0; near 0–5; far 5+)	near	0.0092 (0.0054)	0.034	0.103	0.165	0.368
main (below 0; near 0–5; far 5+)	far	0.0003 (0.0023)	0.373	0.582	0.774	1.000
near 0–3; far 3+	below	0.0009 (0.0029)	0.414	–	0.775	1.000
near 0–3; far 3+	near	0.0121 (0.0071)	0.012	–	0.096	0.144
near 0–3; far 3+	far	0.0003 (0.0014)	0.348	–	0.776	1.000
near 0–10; far 10+	below	0.0009 (0.0029)	0.411	–	0.774	1.000
near 0–10; far 10+	near	0.0067 (0.0038)	0.046	–	0.209	0.465
near 0–10; far 10+	far	−0.0020 (0.0019)	0.846	–	0.846	1.000
tolerance (below −1; near −1–5)	below	0.0008 (0.0031)	0.424	–	0.776	1.000
tolerance (below −1; near −1–5)	near	0.0073 (0.0042)	0.050	–	0.209	0.465
tolerance (below −1; near −1–5)	far	0.0003 (0.0023)	0.354	–	0.766	1.000

Notes: Same regression and sample as Table 4, panel B, under each definition of position, with standard errors clustered by week. The p-values are one-sided (a rise in cuts) and come from the same redraws of the dates of the cut decisions for all twelve contrasts. With the major-release weeks redrawn instead, the twelve-contrast adjusted p of the primary contrast is 0.13, and over the 132 circular shifts of the release calendar it is 0.06. The Romano–Wolf p-values adjust within the three positions of the main definition and within all twelve contrasts.

the floor are large, and none survives either adjustment at the five percent level.

Launch timing around open releases. The timing of closed launches does not respond to open releases, which the model assumes: closed launches per week after the 11 major releases are no more frequent than before (randomization $p = 0.95$), and the hedonic residual of the launch price for the 45 of the 87 launches that came within sixty days after a major release differs from the others' by 0.30 log points (permutation $p = 0.18$).

Secondary tests of the timing claim. Redrawing the release weeks instead of the cut decisions, uniformly and independently, gives a two-sided p of 0.009 for frontier releases, but that null moves into the panel the releases that predate it and breaks up their clusters;

circular shifts of the release calendar give 0.025. The pooled statistic counts a week once for every release whose window contains it. Counting each week once on each side, the union of the post windows against the union of the pre windows, gives 0.025 cuts a week (two-sided $p = 0.18$) and 0.015 decisions ($p = 0.33$), but 84 of the 103 weeks in some post window also lie in a pre window, so that version compares two largely overlapping sets of weeks. For the 11 major releases the difference is 0.051 (one-sided $p = 0.159$), and for the 8 major releases from Chinese developers, which are least likely to be timed against US list prices, 0.054 (one-sided $p = 0.175$). Two placebos, which test the mechanism rather than the claim, fail to show the pattern: open releases more than five points below the open floor (33 releases; -0.008, $p = 0.863$) and new closed frontier models (27 releases; -0.007, $p = 0.833$). A distributed-lag regression with binned leads and lags (Schmidheiny and Siegloch, 2023) and Newey–West standard errors gives a post-minus-pre effect of 0.061 (0.037) for frontier releases and 0.070 (0.054) for major ones. The leads are imprecise and of mixed sign. Closed launches do not respond (randomization $p = 0.95$), and closed models launched within sixty days after a major open release are not priced lower than others at given characteristics (difference 0.30 log points, permutation $p = 0.18$). Two further secondary checks ask whether open releases coincide with shocks that would bring cuts anyway. The hosted prices of the 21 open-weight models released before July 2024, far below the open floor throughout, track the cost of serving a model and the competition among its hosts. Their index does not move around frontier releases (post-minus-pre change -0.0024 a week, $p = 0.22$), and removing the part of the weekly cut series that its current and lagged changes predict (0.8 percent of the variance) leaves the timing statistic at 0.063 ($p = 0.041$; Table F.13). The same providers' prices of image, video, and audio products, which do not compete with open-weight language models, fell 5 times in the sample without clustering after frontier open releases (-0.008, $p = 0.43$). That placebo has little power, and its dates are the file's. Leaving out any one of the 7 cut decisions, the statistic lies between 0.025 and 0.073 cuts a week. The two-sided p stays at or below 0.058 when the decision left out is Google's, Anthropic's, or xAI's, and rises to between 0.14 and 0.31 when it is one of OpenAI's three, the largest for the cuts that followed Kimi K3. With four- and twelve-week windows it is 0.083 ($p = 0.072$) and 0.043 ($p = 0.068$). The rate in the eight weeks before a release, 0.051, is below the base rate, so part of the contrast

is a lull before releases, but the post-window rate alone exceeds the base rate (one-sided $p = 0.015$, with the dates of the cut decisions redrawn), and in the distributed-lag regression the nearest lead bin is negative, the same lull, and the other two are small and of mixed sign (-0.048, 0.019, and -0.026 cuts per week per release in the three lead bins; standard errors 0.028, 0.025, and 0.025), so cuts do not precede releases.

The open-minus-closed contrast. The difference between the post-minus-pre cut rate after frontier open releases and after new closed frontier models is 0.067 cuts a week, with a one-sided p of 0.045 (two-sided 0.091) from 20,000 redraws of the dates of the cut decisions, both event calendars kept, against the 27 new closed frontier models since January 2024, and 0.096 one-sided counting each decision once. Against the 11 major releases alone the difference is 0.058 ($p = 0.165$), and the frontier releases exceed the open releases far below the open floor by 0.068 ($p = 0.145$).

The provider's own frontier. For each of the 8 provider cuts, the cut version's distance to its provider's best listed version in the week of the cut (the week containing the provider's date) is compared with the same distance over all closed model-weeks. The cut versions sat a median of 3.5 points below their provider's top, against 12.3 for listed model-weeks (permutation $p = 0.016$ over 10,000 draws of 8 model-weeks), 5 of the 8 were within five points of it against 24 percent of model-weeks (exact binomial $p = 0.023$), and 3 were the provider's top version itself (Gemini 1.5 Pro (Sept 2024), o3, and Grok 4.20). Within the week of the cut, a cut version was closer to its provider's top than 79 percent of the listed models (one half if position in the line does not matter; simulation $p = 0.002$, drawing a random listed model in each cut's week). The cut versions' lead over the open floor was a median of -0.1 points against -6.9 for listed model-weeks, and their distance to the best closed model of any provider a median of 6.7 against 14.8.

Release timing. Table F.13 reports the timing statistic with the serving-cost controls and the placebo on non-language products. Its distributed-lag rows drop the first five weeks of the panel so that the lags are defined, which is why they differ slightly from those of Table 4. The cost index is a matched-model index of the hosted prices of the 21 open-weight models

Table F.13: Release timing with a serving-cost control and a placebo

Outcome and control	Post minus pre	p	Reading
Provider cuts, frontier releases	0.060	0.054	as in Table 4, with new redraws
residualized on the median-host cost index	0.063	0.041	current and four lagged weekly changes, trend
residualized on the cheapest-host cost index	0.064	0.039	
distributed lag; with cost controls	0.065 (0.037); 0.064 (0.039)	–	Newey–West s.e.
Provider cuts, major releases	0.051	0.159	one-sided
residualized on the median-host cost index	0.054	0.129	one-sided
Weekly change of the cost index, frontier; major	−0.0024; 0.0003	0.22; 0.92	do releases coincide with cost shocks?
Placebo: non-language price cuts, frontier; major	−0.008; −0.001	0.43; 0.66	5 provider-week events

Notes: Post minus pre: mean weekly outcome in the eight weeks after each release minus the eight weeks before, two-sided p-values unless stated. For provider cuts the p-values keep the release calendar and redraw the dates of the cut decisions (20,000 times for the raw series, 5,000 for the residualized ones, which are residualized again in every draw); for the cost index and the non-language products they redraw the release weeks, 5,000 times. Residualized series remove a trend and the current and four lagged weekly changes of the cost index. The distributed-lag rows drop the first five weeks of the panel so that the lags are defined.

released before July 2024, at the median independent host and at the cheapest listing. It rose 11 and 2 percent a year, so hosts did not pass declines in serving cost on to old models, and it is an imperfect proxy for the cost of serving frontier models. The non-language events are falls in a listed price of the four providers' image, video, and audio products that last at least four weeks, on products listed at least four weeks before, grouped by provider and week.

Age of the versions cut. The remaining-life mechanism of Proposition B.1 compares lines at the same distance from the free alternative, and versions sink below the floor as they age, so a secondary test compares each cut version only with the closed models listed in the same week and the same position relative to the floor. The mean percentile is then 0.22 (exact randomization $p = 0.0008$), and each of the 7 cut versions that shared its position with other versions was younger than the median version there. The pattern holds for the 4 cuts below the floor (0.12, $p = 0.002$), which Proposition 3 does not predict but to which the remaining-life logic applies all the same, and for the 3 just above it, where each comparison set has only a few versions (0.26, $p = 0.03$). In the stacked comparison the models cut were a median of 6 weeks old against 34 for all models listed. Table F.14 lists the 8 provider cuts with the cut model's position relative to the open floor in the week of the cut, its age, the number of closed models listed in that week and position, and its age percentile among them. The randomization test draws, for each cut, one model at random from the same week and position, so a cut with no other model in its position adds nothing to the test. A

Table F.14: The versions cut, compared with versions in the same position

Model	Cut	Position	Age (months)	In position	Percentile
Gemini 1.5 Flash (May 2024)	2024-08-12	below	2.7	15	0.10
Gemini 1.5 Pro (Sept 2024)	2024-10-01	0 to 5 above	0.2	4	0.25
Claude 3.5 Haiku	2024-12-03	below	1.4	17	0.09
o1-mini	2025-01-31	below	4.6	24	0.25
o3	2025-06-10	5 or more above	1.8	1	0.50
Grok 4.20	2026-05-06	0 to 5 above	2.6	6	0.33
GPT-5.6 Terra	2026-07-30	0 to 5 above	0.7	5	0.20
GPT-5.6 Luna	2026-07-30	below	0.7	51	0.05

Notes: The 8 verified provider cuts. Position is the cut model's ECI relative to the open floor in the week of the cut, in ECI points. In position counts the closed models listed in that week and position, the cut model included. Percentile is the cut model's age percentile among them, the share of those models younger than the cut model, with ties counted half, and it is one half if age does not matter. Mean percentile 0.22, exact randomization $p = 0.0008$.

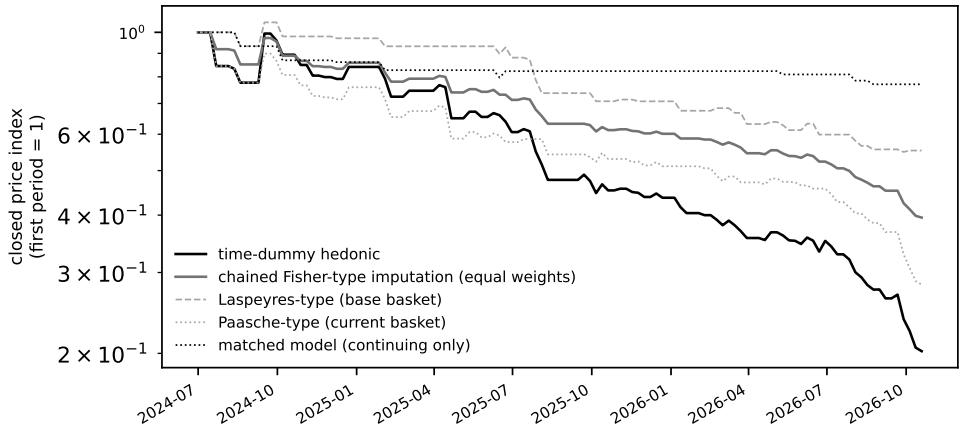

Figure F.2: Closed-model price indexes, weekly: time-dummy hedonic; chained Fisher-type imputation index with equal weights and its base-basket (Laspeyres-type) and current-basket (Paasche-type) components; and the chained index of continuing versions alone. Imputations from the pooled hedonic regression.

version's remaining life can also be read from the providers' retirement notices: none of the cut versions had an announced retirement when it was cut, against 0.51 expected if cuts fell at random within week and position, a check without power.

The index formula. Figure F.2 compares the time-dummy index with the chained Fisher-type imputation index and its two baskets.

Entry and exit in the index. The time-dummy index can be split exactly. With week effects, each week's effect is the mean of that week's log prices net of the fitted characteristics and provider effects, so its change from one week to the next is the sum of the price change of continuing versions, the change in their fitted characteristics, the versions listed for the first time, those listed for the last time, and those that leave the file and return. Each entering version adds its distance from the continuing versions, weighted by its share of the week's versions, and each leaving version subtracts its own. A first listing within forty-five days of release is a launch, the rule of the launch schedule, and a last listing is a withdrawal when the version's shutdown follows within six weeks. The parts reproduce all 120 week-to-week changes of the index, and fitting the index's trend to each cumulative part splits its slope of -0.600 log points a year (Table F.15). Launches entered below continuing versions with the same characteristics, and the versions withdrawn were relatively expensive, so withdrawals lowered the index by about a third of its decline. A withdrawal need not bring any buyer a new price. For 21 of the 29 versions withdrawn, the same provider listed, in the version's last week and in the week after, a version of at least its ECI at a strictly lower output price, the same number on blended prices, and for 19 one that led it by at least five ECI points. The others are small-tier versions, the cheapest of their lines, and Gemini 3 Pro. The comparison takes the ECI as the measure of capability and ignores the attributes that separate tiers. Other last listings, versions that leave the price file without a recorded shutdown, add little. The accounting uses the index's own weights and characteristics, so it inherits the index's dependence on the measure of capability.

Epoch's revised index. Epoch re-estimated the index with a Bayesian model on 4 November 2026 (Álvarez and Barry, 2026), after the frozen vintage of 22 October 2026 that the paper uses. On the 100 of the 102 closed versions that the vintage archived on 12 November 2026 scores under the same name, the hedonic index falls 44.2 percent a year with the October scores and 44.9 percent with the November ones. All 8 cuts keep their position relative to the open floor, whether below it, zero to five points above it, or further above. The open leader changes in 20 of the 121 weeks, and 141 of the 4,745 matched closed version-weeks change position band. The revised release calendar has 26 frontier open releases rather than

Table F.15: Entry and exit in the closed hedonic index

Part	Versions	Log points a year	Share of the decline (%)
Launches, first listed within forty-five days of release	85	−0.325	54
Withdrawals, last listed before a shutdown	29	−0.189	32
Other last listings	6	−0.028	5
Price changes of continuing versions		−0.072	12
Fitted characteristics of continuing versions		0.007	
Late first listings	1	0.007	
Versions that leave and return		−0.0005	
Time-dummy index		−0.600	

Notes: Slopes of the cumulative parts of the weekly closed time-dummy index of Table 2, from the same regression; they add to the index's slope. Shares are of the index's slope.

25, and 12 major ones rather than 11. With it, on the same redrawn decision dates, the post-minus-pre difference is 0.048 cuts a week (two-sided $p = 0.069$) and 0.030 decisions ($p = 0.19$). The price results do not depend on the vintage, and the timing evidence, which rests on the dates of a few releases, weakens.

Dose response. We describe each of the 21 frontier releases whose sixty-day window lies in the sample by the ECI points it adds to the open leader, the share of the gap to the best closed model it closes, the points by which the open leader still trails the best closed model afterwards, and a proxy for the loss in Proposition 3, points closed squared over points behind. Of the steps that closed most of the gap, the fourth largest, Qwen3-235B Thinking (42 percent), was not followed by a cut. GLM-5.1 was a major step for open models but closed only 15 percent of the gap, because the closed frontier had moved far ahead: the open leader still trailed by 9.8 points after it, against 5.6 after Kimi K3. The secondary tests (Table F.17) agree in sign: with the same crediting, more points closed, a larger share closed, fewer points behind, and a higher index all go with more cuts. One-sided permutation p-values are between 0.016 and 0.090 for the slopes and between 0.056 and 0.141 for the rank correlations, and the points by which the open leader still trails give the strongest rank correlation (-0.35, $p = 0.056$). Counting the cuts within thirty days of each release gives the same pattern, with slope p-values between 0.011 and 0.081 and rank p-values between 0.074 and 0.102. Counting the cuts within sixty days shows no relation, because the windows of nearby releases overlap: the cuts that followed Kimi K3 also fall in

Table F.16: Frontier open-weight releases, the size of the step, and the provider cuts that followed

Date	Release	Points closed	Share closed (%)	Points behind	Cuts, 30 days	Cuts, 60 days	Cuts credited
2024-07-23	Llama 3.1-405B	3.49	74	1.23	1	1	0.5
2024-07-23	Llama 3.1-70B	0.65	14	4.07	1	1	0.5
2024-09-19	Qwen2.5-72B	0.20	3	6.73	1	1	1.0
2024-11-18	Mistral Large 2 (Nov 2024)	0.00	0	6.73	1	1	1.0
2024-12-12	Phi-4	1.51	22	5.22	0	1	0.0
2024-12-26	DeepSeek-V3	1.82	16	9.42	0	1	0.0
2025-01-20	DeepSeek-R1	6.52	69	2.90	1	1	1.0
2025-04-28	Qwen3-235B-A22B	0.33	4	7.58	0	1	0.0
2025-04-29	Qwen3-32B	0.00	0	7.58	0	1	0.0
2025-04-29	Qwen3-14B	0.00	0	7.58	0	1	0.0
2025-05-28	DeepSeek-R1 (May 2025)	1.98	26	5.60	1	1	1.0
2025-07-25	Qwen3-235B-A22B-Thinking (Jul 2025)	2.63	42	3.57	0	0	0.0
2025-09-29	DeepSeek-V3.2-Exp	1.15	18	5.09	0	0	0.0
2025-11-06	Kimi K2 Thinking	0.91	16	4.77	0	0	0.0
2025-12-01	DeepSeek-V3.2	0.27	4	6.98	0	0	0.0
2026-01-27	Kimi K2.5	1.98	19	8.26	0	0	0.0
2026-04-07	GLM-5.1	1.75	15	9.75	1	1	0.0
2026-04-20	Kimi K2.6	1.12	11	8.63	1	1	1.0
2026-06-12	Kimi K2.7 Code	0.00	0	12.48	0	2	0.0
2026-06-16	GLM-5.2	1.00	8	11.48	0	2	0.0
2026-07-16	Kimi K3	5.92	52	5.56	2	2	2.0

Notes: Points closed: the rise of the open leader's ECI. Share closed: points closed over the gap between the best closed model and the open leader before the release. Points behind: that gap after the release. Cuts: verified provider cuts dated by the provider, counted in the window after the release or credited to the nearest preceding release.

Table F.17: Dose response: provider cuts and the size of the open-weight step

	Cuts credited to the nearest release		Cuts within 30 days		Cuts within 60 days
Measure of the step	Rank correlation (p)	Slope p	Rank correlation (p)	Slope p	Rank correlation (p)
Points closed	0.26 (0.124)	0.016	0.33 (0.074)	0.011	−0.09 (0.661)
Share of the gap closed	0.24 (0.141)	0.054	0.29 (0.102)	0.030	−0.18 (0.784)
Points behind the closed frontier	−0.35 (0.056)	0.090	−0.29 (0.100)	0.081	0.37 (0.953)
Proposition 3 index	0.26 (0.127)	0.044	0.32 (0.079)	0.030	−0.15 (0.737)

Notes: Spearman rank correlations and OLS slopes of the cut count on the measure across releases, with p-values from 10,000 permutations of the measure across releases, one-sided in the direction of the hypothesis (more cuts after bigger steps and after releases that leave the open leader fewer points behind). The Proposition 3 index is points closed squared over points behind.

the windows of two smaller releases a few weeks before it, and the cut of o3 in the windows of three Qwen releases. Deeper cuts also follow bigger steps in rank terms (0.48, $p = 0.11$, eight cuts). The evidence leans toward the dose response the model predicts, but it rests on three releases. Table F.16 lists the frontier releases with a complete sixty-day window, the size of each step, and the cuts that followed, Table F.17 reports the tests, and Figure F.3 shows the cuts credited to each release against two of the measures. A cut is credited to the nearest release in the sixty days before it, split equally between releases on the same date. Crediting it to the largest preceding step instead would build the hypothesis into the outcome.

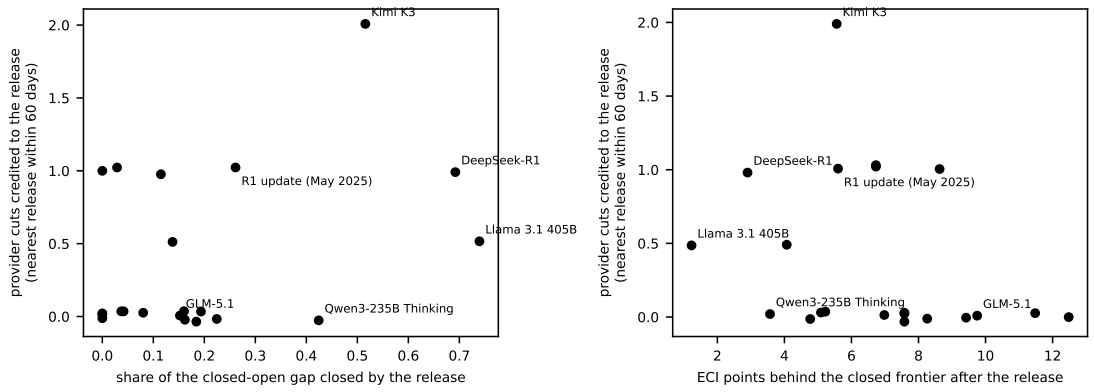

Figure F.3: Provider cuts credited to each frontier open-weight release against the share of the closed–open gap it closed (left) and the ECI points by which the open leader still trailed the best closed model afterwards (right).

Retirement hazards. Proposition B.5 predicts that a version that an open model dominates in capability and price loses its buyers at its posted price and, if serving it costs at least the open model's hosted price, is withdrawn rather than repriced, so retirements should concentrate below the open floor, although withdrawal can also come before the overtaking. We estimate discrete-time complementary log-log hazards of a retirement announcement on 4,065 model-weeks of 100 closed models, with 52 announcements, controlling for age, the distance to the closed frontier, small and pro tiers, provider, and a trend. With age entered linearly, being below the open floor raises the hazard by a factor of 5.1 (coefficient 1.63; standard error 0.98 clustered by announcement week, 0.84 by model), and a year of age raises it by a factor of 3.7. The distance to the closed frontier has a coefficient of −0.013 (0.034), and the distance to the provider's own best version, used instead, −0.025 (0.032): a version's position relative to newer versions does not predict its retirement. The continuous lead over the open floor has no effect in the same specification (0.047, 0.094). Announcement rates are 0.85 per model-year below the floor, 0.08 just above it, and 0.16 far above it (Table F.18). A logit with the same controls gives an odds ratio of 5.1.

More retirement events. Adding retirement events does not separate the overtaking from the version's age, because the added events also fall below the floor. Mistral's closed models and Amazon Bedrock's legacy notices for Anthropic's models add 2 and 7 announce-

Table F.18: Retirement hazards and the top of the line

Object	Estimate	Reading
A. Retirement announcements: 52 events, 100 models, 4,065 model-weeks; complementary log-log		
Below the open floor	1.63 (0.98; 0.84)	hazard ratio 5.1; s.e. by week; by model
Age (years)	1.31 (0.43)	hazard ratio 3.7
Distance to the closed frontier	−0.013 (0.034)	newer versions do not predict retirement
Lead over the open floor (continuous, separate model)	0.047 (0.094)	no gradient
Rates per model-year: below; 0–5 above; 5 or more above	0.85; 0.08; 0.16	logit odds ratio below the floor 5.1
Below the open floor, age in bands instead of linear	0.73 (0.89)	likelihood-ratio $p = 0.42$ (linear age: 0.02)
Exact rate ratio below the floor: all; within age bands	8.1; 1.7	intervals 2.1–69; 0.3–18
B. The top of the line (log list price)		
Top version, model fixed effects: open floor; ECI of the rivals' best closed model	0.0023 (0.0018); 0.0018 (0.0022)	no detectable response
Frontier launches: open floor; ECI of the rivals' best closed model	−0.18 (0.08); −0.16 (0.08)	34 launches; the two levels move together (correlation 0.97)
open floor, without the launches after the two largest open jumps	−0.05 (0.18)	24 launches

Notes: Age bands and exact rate ratios are those of Table F.19. Hazard models include small and pro tier indicators, provider effects, and a trend. The age bands are under three months, three to six, six to nine, and nine to twelve months, one year to eighteen months, and older. Frontier launches are the launches of Table F.6 within two ECI points of the closed frontier the day before the release, when the levels are measured. The two largest open jumps are DeepSeek-R1 and Kimi K3, and the last row drops the frontier launches released within 120 days after either. Launch regressions include own ECI, a reasoning indicator, log context window, provider effects, and a trend, with heteroskedasticity-robust standard errors. The within-model regression uses each provider's top listed model by week, with week-clustered standard errors.

ments, and following the panel's versions back to their release in 2023 adds 1. Pooled with Mistral's and Bedrock's events, the rate ratio within age bands is 1.6 (0.2 to 18). With the earlier window, which adds versions above the floor at older ages but few retirements on either side, it is 3.5 (0.6 to 29).

Exact and local tests of the retirement threshold. Table F.19 collects the tests of Section 6.3, all on the model-weeks of Table F.18 unless stated. A local comparison at the floor is too thin to help: within five ECI points of it there are 11 announcements below

and 1 above (rate ratio 7.6, interval 1.1 to 328), and local linear hazards with separate slopes on each side show no significant positive jump (the one significant estimate, in the narrowest window with controls, is negative). The exact rate ratios condition on the number of announcements: given n announcements, the number below the floor is binomial with probability $\psi E_1/(\psi E_1 + E_0)$, where E_1 and E_0 are the model-years below and at or above the floor, so the Clopper–Pearson interval for that probability gives an exact interval for the rate ratio ψ. The stratified versions combine the strata's binomials by convolution, invert the test for the interval, and report the Mantel–Haenszel estimate. The batch tests take each provider-week with an announcement and count how many of the versions retired would be below the floor if the provider had drawn them at random from its versions at risk that week, a sum of hypergeometric counts. The conditional logit uses the same strata with the exact likelihood for tied events. A version is dominated in a week if an open-weight model of at least its ECI is hosted at an output price no higher than its own, on the weekly hosted panel with batch tiers and file errors removed. The local comparisons use only the model-weeks within the stated number of ECI points of the floor. The launch-lead rows of panel C keep the versions launched at or above the floor and replace being below it with the lead over the floor at launch. A version launched further ahead is not retired at a detectably different rate with either age control. The local linear hazards add the lead over the floor on each side, with uniform or triangular kernels and with or without the controls of Table F.18. The placebo thresholds compare windows of 2.5 points on each side of a false threshold, using only versions below the floor. Figure F.4 shows the announcement rate by position and by age: below the floor the rate is higher than above it at every distance, and the model-years above the floor are concentrated in the first six months, where the rates on both sides are low. The extensions take the listings from the weekly litellm snapshots. Mistral's closed models with an ECI score give 2 retirement notices in the sample, for Mistral Large and Mistral Medium 3. Ministral 3B is listed only after its notice and Mistral Large 4 only from October 2026. Anthropic's models on Amazon Bedrock are matched to the panel on the model identifier, with the start of Bedrock's legacy period as the announcement. The earlier window treats the panel's versions as at risk from their release, before the snapshots begin. It adds 1.5 model-years above the floor at ages over six months, for the GPT-4 versions of March and

Table F.19: Retirement: exact tests, the local design at the floor, and more events

A. Exact rate ratios, first group against second

	Events	Rate ratio (95% interval)	p
Below against at or above the floor, all model-weeks	50 / 2	8.1 (2.1–68.9)	<0.001
versions under a year old	24 / 2	5.6 (1.4–48.7)	0.006
versions under six months old	2 / 2	1.0 (0.1–13.7)	1.00
within five ECI points of the floor	11 / 1	7.6 (1.1–327.7)	0.035
within ten ECI points of the floor	19 / 2	5.3 (1.3–47.0)	0.011
within age bands (Mantel–Haenszel)	38 / 2	1.7 (0.3–18.3)	0.91
within provider and age band	20 / 2	1.5 (0.3–17.2)	0.97
Dominated against at or above the floor	50 / 2	9.1 (2.4–76.9)	<0.001
Dominated against undominated below the floor	50 / 0	above 1.6	0.009
within age bands	38 / 0	above 0.5	0.31

B. Within provider batches (21 provider-weeks with announcements): versions retired below the floor

	Observed	Expected	p
Within the batch	50	39.5	<0.001
Within the batch and age band	50	50.4	0.81
Conditional logit on below the floor: alone; with linear age; with age bands	2.40 (0.74); 1.40 (0.79); 0.57 (0.96)		

C. Hazards (Table F.18 specification) and the local design

	Coefficient	Log likelihood	LR p
Below the floor, age linear (Table F.18)	1.63 (0.98)	−247.9	0.015
age quadratic	1.20 (0.93)	−246.1	0.11
age spline, knots at six months and one year	1.08 (0.90)	−243.2	0.22
age bands	0.73 (0.89)	−244.2	0.42
Dominated instead of below: age linear; age bands	2.09 (0.94); 1.42 (0.91)		
Launch lead, versions launched at or above the floor: age linear; age bands	−0.08 (0.10); 0.056 (0.041)		
Distance to the provider's own best version, instead of the closed frontier	−0.025 (0.032)		
Local linear jump at the floor (bandwidths 5, 10, 15, all; uniform and triangular kernels; with and without controls)	−3.11 to 1.16; largest $\lvert z \rvert$ 2.56		
Placebo thresholds below the floor (7; windows of 2.5 points)	rate ratios 0.4 to 2.0; smallest p 0.34		

D. More retirement events

	Events (above)	Within age bands	p
Mistral's closed models with an ECI score	2 (0)		
Amazon Bedrock legacy notices for Anthropic's models, a median 6 months after Anthropic's own	7 (0)		
Panel versions from their release, March 2023 to June 2024	1 (0)	3.5 (0.6–29.1)	0.25
Pooled: Table F.18 sample, Mistral, and Bedrock	61 (2)	1.6 (0.2–18.4)	0.97

Notes: Panel A: exact conditional tests of the ratio of announcement rates per model-year in the first group to the second, two-sided. Within strata, the Mantel–Haenszel estimate with the exact conditional interval, counting the announcements in strata with exposure in both groups. Panel B: provider-weeks with at least one announcement, with expected counts and one-sided p-values from sums of hypergeometric counts, and conditional logit coefficients with standard errors. Panel C: complementary log-log hazards with the controls of Table F.18, standard errors clustered by week. The likelihood-ratio p tests the below-floor indicator given the age specification. Panel D: the rate ratio of the row for the panel versions from their release is for the sample of Table F.18 with the earlier model-weeks added. The pooled row's is for the sample of Table F.18 with Mistral's and Bedrock's model-weeks added.

June 2023 and Claude 2, without a retirement, but in those months the providers retired almost no versions on either side of the floor. The other records cannot be used: Azure lists announcement dates only for its 2023 batch of legacy models, Vertex AI lists only retirement dates, Cohere's closed models and Amazon Nova have no ECI score, and the closed models of Alibaba and Meta have no retirement records.

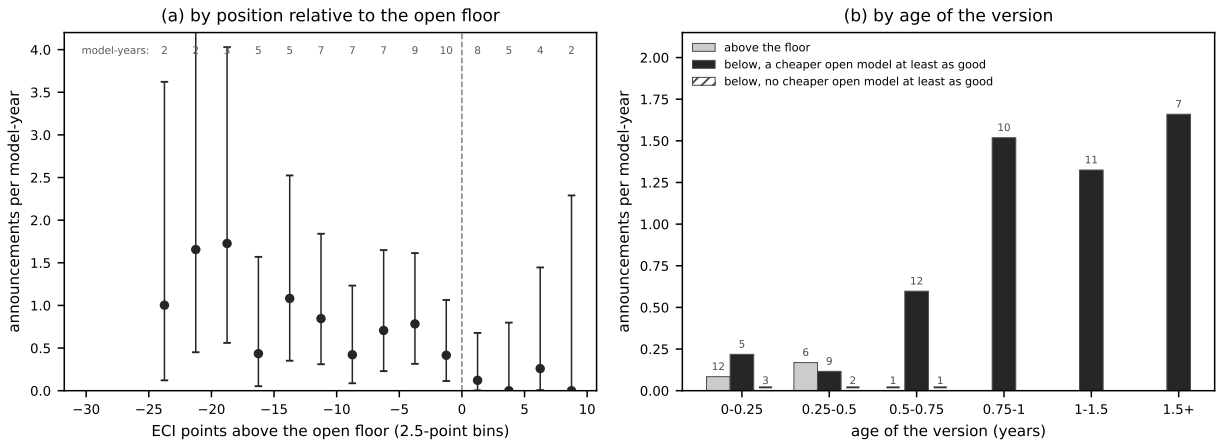

Figure F.4: Retirement announcements per model-year on the model-weeks of Table F.18. (a) By the version's ECI lead over the open leader, in 2.5-point bins with at least one model-year of exposure, with exact 95 percent intervals. The numbers at the top are model-years. (b) By the version's age, for versions at or above the floor, versions below it that an open model of at least equal ECI undercut in price, and versions below it that no such model undercut. The numbers are model-years, and cells with less than half a model-year are left out.

Usage by version. The tables count tokens by version, so new customers, heavier use, and customers who switch versions all enter the same totals. Sections 4 and 6.5 use tables of every model's tokens that many archived rankings pages of OpenRouter hold beside the chart of named models. Captures from July 2024 to January 2025 give each model's weekly tokens over the year before the capture, and they match the chart's named models to within 1 percent. Captures from May 2025 to May 2026 give each model's tokens over the seven days before the capture, and their totals are 0.94 to 1.06 times the chart's tokens on the same days. We use 29 weekly and 49 seven-day tables, each priced at the panel's list prices of the Monday on or before its midpoint, and no complete table covers February to mid-May 2025. Every closed slug of the four providers is mapped to a panel version, a dated slug to the version of that date and an undated one to the version it served that week. Variants of a version count as that version, and free and experimental variants are left out. The panel's versions carry 81 percent of the four providers' language-model tokens, and 17 percent is on chat models without an ECI score, chiefly xAI's Grok Code Fast and Grok Fast models. A version is dominated when an open-weight model of at least its ECI is hosted at no higher output price, at its median host as in Section 6.1 or at its cheapest screened listing as in the

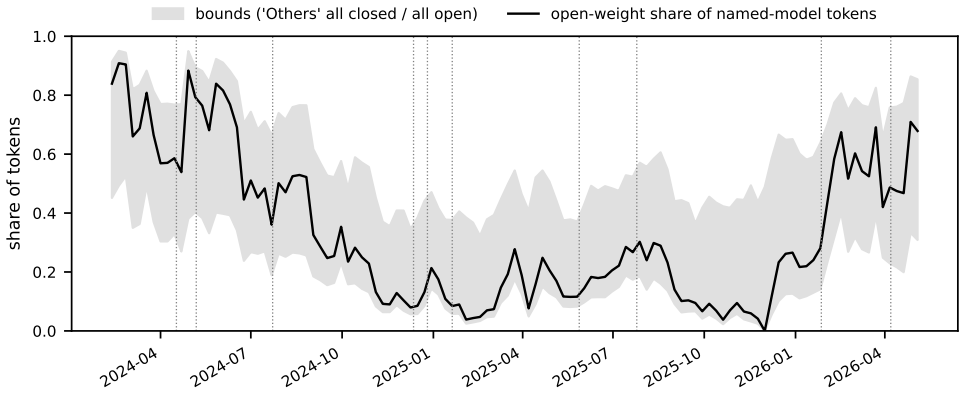

Figure F.5: Open-weight share of tokens on OpenRouter, weekly. Line: share of the tokens of named models. Band: bounds that count all unnamed tokens as closed or as open. Dotted verticals: major open-weight releases.

retirement analysis. Table F.20 reports the shares. The young versions' share of tokens is 83 percent when each week counts equally, and 75 percent with the unscored models added, each dated by its first appearance in the tables. The chart's named models alone, a median of 9 a week, give 89 percent on the same weeks, because older versions with little usage fall into the chart's pooled remainder. Spending is valued at list prices, which overstate what buyers pay where they cache their prompts. Billing every prompt token at a tenth of the input price, the cache-read rate, leaves the young versions with 83 percent of spending. Many dominance relations rest on a few ECI points, and with a margin of two points the dominated share of tokens is 34 percent. Weighted by tokens, buyers paid a geometric mean of 2.1 times the price of the cheapest closed version of at least the same ECI listed that week, against 8.1 across the listed versions, so few tokens pay the cost of staying of Section 4.

The weekly test of usage. On OpenRouter, a marketplace that routes developers' requests to closed and open models, the weekly token volume of the leading models grew from 12 billion to 25,743 billion tokens over the sample. The open-weight share of the tokens processed by named models was 0.47 in 2024, 0.15 in 2025, and 0.49 in 2026 (Figure F.5). The named models carry only 59 percent of all tokens, and the rest go to smaller models that the chart does not name, so the bounds are wide. The share shows no systematic response to open releases: the change in its log odds after major releases is within the range of random

Table F.20: Usage by version on OpenRouter

	Share of tokens or spending (percent)					Price paid over	
			Dominated				
	Under six months	Below the floor	Median host	Cheapest listing	Margin of five	Open model	Cheapest closed version
Tokens, all weeks	72	52	41	48	17	2.4	2.1
Listed version-weeks, equal weights	40	78	66	72	44	–	8.1
Tokens, 2024 (from July)	96	38	7	8	4	6.2	1.6
Tokens, 2025	79	67	45	58	15	2.8	2.6
Tokens, 2026 (to May)	67	44	40	44	18	2.4	1.8
Spending at list prices	88	28	23	24	7	5.2	2.8
Spending, prompt tokens at a tenth	83	33	28	29	9	6.0	2.8
Tokens, without Google's Flash models	78	43	33	38	14	5.0	2.5
Tokens, OpenAI	53	66	54	62	34	3.6	1.8
Tokens, Anthropic	91	32	27	27	8	5.0	3.2
Tokens, Google	63	62	51	60	17	2.1	1.6
Tokens, xAI	79	86	14	66	2	1.2	1.2
Tokens, the chart's named models, same weeks	89	36	24	31	8	2.4	1.9

Notes: Shares of the paid tokens of the closed versions of OpenAI, Anthropic, Google, and xAI in the price panel, or of their spending at list prices, in the 29 weekly and 49 seven-day complete usage tables of OpenRouter. The row of listed version-weeks gives each version listed in the panel in those weeks equal weight. Under six months: since the version's release. Below the floor: the version's ECI is below the open floor. Dominated: an open-weight model of at least the version's ECI is hosted at no higher output price, at its median host or at its cheapest screened listing; with a margin of five, the open model leads by at least five ECI points. Price paid over the open model: the version's output price over the lowest median-host price among open models of at least its ECI, the median over dominated tokens. Over the cheapest closed version: over the cheapest closed version of at least the same ECI listed that week, the geometric mean. Prompt tokens at a tenth: every prompt token billed at a tenth of the input price, the cache-read rate. The chart's named models: the models the weekly chart names, a median of 9 a week.

draws ($p = 0.86$; $p = 0.45$ for frontier releases). This is a check with little power. With 80 percent power the test detects only a rise in the log odds of 0.79 after major releases, about 19 percentage points of the open share at its sample mean, and of 0.54 (13 points) after frontier releases (Table F.22). Individual releases can win large shares quickly, such as Kimi K2.5 with 12 percent of all tokens in its second week and DeepSeek-R1 with 6 percent within two months, but the weekly chart cannot show where those tokens came from. The absence of a measured response is consistent with the differentiation and multihoming that Fradkin (2025) and Demirer et al. (2026) document, and with a premium that buyers of closed models are willing to pay, but it does not establish them.

The daily test of usage. A weekly chart cannot separate a release from everything else that happens in its week. The daily charts allow a regression discontinuity in time around the 6 frontier open-weight releases listed on OpenRouter between 11 September 2025 and 10 May 2026, 2 of them major. GLM-5.1 is among them. Kimi K3 falls in a gap of the archive, and OpenRouter's own activity data reach back only a month. For each release we take the daily log tokens of every incumbent paid model, one listed before the window and named on most of its days, over the fourteen days before and after the listing, fit separate trends on each side with day-of-week effects, and compare the pooled jump with the same statistic at placebo dates. These incumbents carry 34 percent of English-prompt tokens in the fourteen days before a release on average, between 24 and 46 percent, and the closed ones among them 28 percent. Free variants and the marketplace's stealth models are left out, because free promotions move tens of percent of tokens within days: GLM-5.1 was listed on the last day of a free promotion of a closed model, Qwen3.6 Plus, which took 20 to 45 percent of English-prompt tokens on each of its last five days. In English prompts the incumbent closed models' usage does not jump at the releases (0.01 log points, 0.07; placebo $p = 0.80$), and neither does that of the incumbent open models (0.07, $p = 0.42$). Seven-day windows give the same (0.05, $p = 0.57$). Some releases are adopted fast, Kimi K2.5 and K2.6 with 9.7 and 10.0 percent of English-prompt tokens on average over their first ten days and GLM-5.1 with only 1.1, but none visibly draws usage from the named incumbents (Table F.21 and Figure F.6). In programming prompts the incumbent closed

Table F.21: Daily usage on OpenRouter around frontier open-weight releases

Release	Listed	ECI gain	Share, days 1–10 (%)	English prompts: jump in log tokens			Programming
				Closed incumbents	Open incumbents	All tokens	Closed incumbents
DeepSeek-V3.2-Exp	2025-09-29	1.15	0.0	−0.21 (0.07)	0.01 (0.05)	0.02 (0.04)	−0.61
Kimi K2 Thinking	2025-11-06	0.91	0.0	−0.44 (0.48)	–	−0.47 (0.44)	−0.50
DeepSeek-V3.2	2025-12-01	0.27	0.4	0.11 (0.05)	−0.10 (0.12)	−0.13 (0.04)	0.06
Kimi K2.5	2026-01-27	1.98	9.7	0.08 (0.04)	0.03 (0.10)	0.12 (0.03)	0.12
GLM-5.1	2026-04-07	1.75	1.1	0.12 (0.05)	0.12 (0.05)	−0.47 (0.05)	0.02
Kimi K2.6	2026-04-20	1.12	10.0	−0.10 (0.08)	0.10 (0.06)	0.01 (0.07)	−0.24

Notes: Share: the released model's share of English-prompt tokens averaged over its first ten days (zero when it is not among the nine named models). Jumps: local linear regression discontinuity at the listing date with fourteen days on each side, the release day dropped, separate intercepts and slopes for each incumbent model and side, and day-of-week effects, with standard errors clustered by day in parentheses. Programming: the same for prompts in a programming language, point estimates.

models' usage falls (-0.10, $p = 0.005$), but the chart's total falls more (-0.21, $p < 0.001$), so the fall is not a move toward the released models. The daily test also detects only large shifts: with 80 percent power, a fall of 12 percent in the incumbent closed models' usage when the noise is measured by the placebo dates, and of 17 percent when it is measured by the clustered standard error. Had the released models drawn every token of their first ten days from the incumbent closed models, these would have lost about 29 percent of their usage, mostly at Kimi K2.5 and K2.6, which the test would detect. Had they drawn from all models in proportion to their shares, the loss would have been about 4 percent, which it cannot detect. Both demand tests are therefore low-power checks. The daily data are inconsistent with open releases taking their usage wholesale from the leading closed models, but not with their taking a proportional part of it.

Daily usage around releases. Table F.21 reports, for each frontier release in the daily data, the released model's share of English-prompt tokens over its first ten days and the jumps in the log tokens of incumbent closed and open models and of all tokens. Figure F.6 shows the two major releases. Incumbents are paid models listed on OpenRouter before the window opens and named on at least four fifths of the window's days on each side of the release. Standard errors are clustered by day. Placebo dates are drawn among the 79 days more than fourteen days from any release, six at a time, in a thousand draws.

Power of the demand tests. Table F.22 reports minimum detectable effects for the weekly and daily demand tests. The random draws behind each published p-value describe

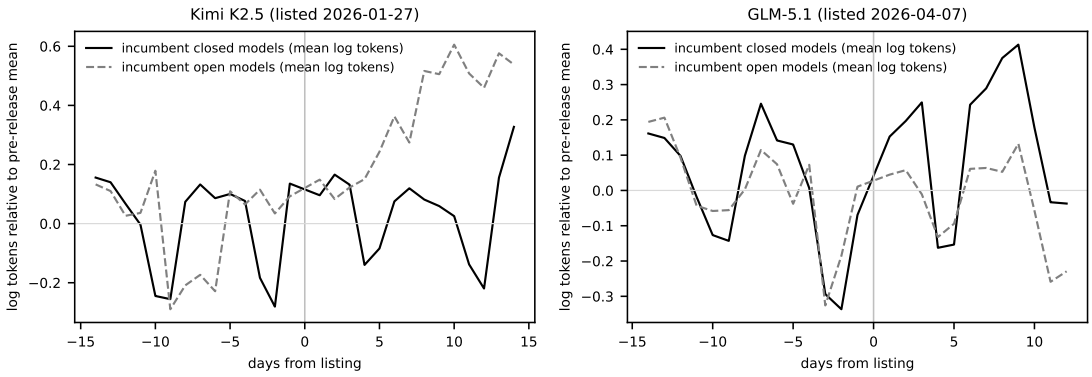

Figure F.6: Daily log tokens of incumbent closed and open models in English-language prompts on OpenRouter around the two major open-weight releases in the daily data, relative to each model's mean before the release.

Table F.22: Power of the demand tests

Test (releases)	Estimate	p	MDE, placebo draws	MDE, standard error
Weekly open share, log odds, major releases (10)	0.02	0.86	0.79 (19 points)	–
Weekly open share, log odds, frontier releases (22)	−0.17	0.45	0.54 (13 points)	–
Daily log tokens, incumbent closed, English, 14 days	0.01	0.80	0.12 (12%)	0.19 (17%)
Daily log tokens, incumbent closed, English, 7 days	0.05	0.57	0.24 (21%)	0.12 (11%)
Daily log tokens, incumbent closed, programming, 14 days	−0.10	0.005	0.10 (9%)	0.28 (24%)
Daily log tokens, incumbent open, English, 14 days	0.07	0.42	0.23 (20%)	0.10 (10%)
Benchmark: all of the released models' tokens from the incumbent closed models	−0.35			
Benchmark: tokens from all models in proportion to their shares	−0.04			

Notes: MDE: minimum detectable effect, the smallest effect that the two-sided 5 percent test detects with 80 percent power. For the weekly tests this is a rise in the log odds of the open share, with percentage points at the sample mean share of 33 percent in parentheses, and for the daily tests a fall in log tokens, in percent in parentheses. Placebo draws: the redraws of release weeks (weekly) and of placebo dates (daily) behind the published p-values, shifted by the effect. Standard error: 2.80 times the standard error clustered by day. Benchmarks: the pooled fall in the incumbent closed models' log tokens had the released models' tokens over their first ten days come entirely from the incumbent closed models, or from all models in proportion to their shares.

the statistic when releases have no effect. A true effect shifts the statistic by a constant, so the power of the two-sided 5 percent test against an effect is the share of the shifted draws that it rejects, and the minimum detectable effect is the smallest effect with a power of 80 percent. For the daily tests the table also gives the effect that a normal test on the clustered standard error detects with the same power, 2.80 standard errors. The two measures of the noise differ in both directions, so we report both. The benchmarks use each release's share of English-prompt tokens over its first ten days and the incumbent closed models' share of the chart's tokens over the fourteen days before it, and pool the implied falls as the regression pools the incumbents.

Discounts and surcharges in the data. The file also lists batch, cached-input, and long-context prices, which we follow as separate series. Batch prices appear for 26 percent of the 4,835 closed model-weeks and for 94 percent of the models listed in the last week. Of these, 99 percent are half the standard price, and no model's batch discount changed. Cached-input prices appear for 66 percent of model-weeks (87 percent in the last week). This discount is set at launch and has deepened across generations: OpenAI launched its models with cached-input discounts of 50 percent in 2024, 75 percent in 2025, and 90 percent in 2026, Google moved from 75 to 90 percent, and Anthropic has offered 90 percent throughout. Within a model the discount deepened 6 times, 4 of them on Google's Gemini 2.5 models, and became shallower once. The file missed one further deepening. With the launch of Claude Haiku 5.5 on 7 October 2026, Anthropic halved the cache-read price of Claude Sonnet 5.5, which by its account took about a fifth off that model's cost on most agentic work, a cut through a discount rather than the list price. The provider pages agree with the file on 142 of 148 cached-input prices, 26 of 26 batch prices (9 of them against half the standard price, where the file had no batch price yet), and 10 of 10 long-context prices. Over a model's life the batch discount is a fixed proportion of the list price, but the cached-input discount deepens now and then, a second margin of repricing that the list prices we study do not show.

Discount series and their timing. Figure F.7 shows the share of closed models that list a batch or a cached-input price and the median ratio to the standard price. In 9 of the 26 batch checks the provider page lists a batch price that the file lacked at the time, so the early shares understate availability. A time-dummy hedonic index on the cached-input price changes by -0.79 log points a year (0.28), against -0.71 (0.24) for the standard input price on the same model-weeks: the deeper discounts of newer models come with higher capability and reasoning, and the regression credits them to those attributes rather than to time (the ECI coefficient is 0.106 on cached and 0.128 on standard input prices). On batch prices the index changes by -0.24 a year, against -0.24 for standard output prices on the same model-weeks. Of the 6 within-model deepenings of the cached-input discount, 2 fall within eight weeks after a major open release, against 51 percent of panel weeks so placed

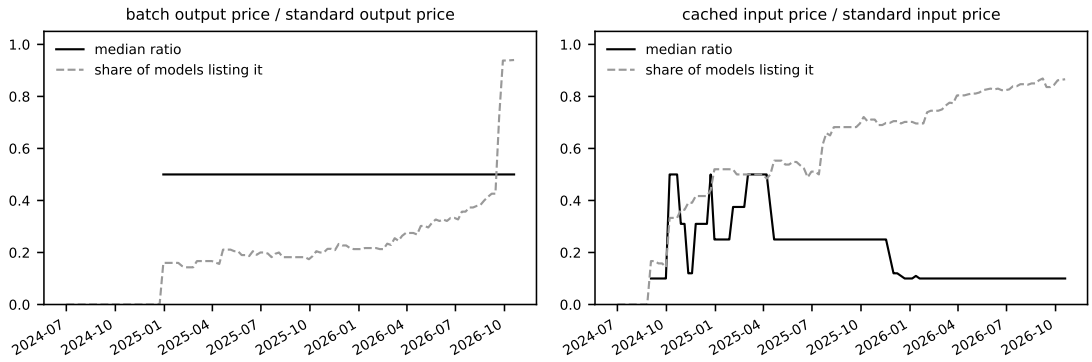

Figure F.7: Batch and cached-input prices of closed models, weekly: share of listed models with each price (dashed) and median ratio to the standard price (solid).

(one-sided binomial $p = 0.90$), so the deepenings show no sign of responding to the free alternative, although they are too few to rule out a response. Long-context prices, listed for 30 models, are a median 1.5 times the standard output price and 2.0 times the standard input price above a context threshold, and none changed within a model. Two file errors are dropped: a cached-input price equal to the standard price for six weeks (Claude 3.5 Haiku) and a long-context price fifteen times the standard one for one week (Gemini 2.5 Pro).

Host dispersion. Figure F.8 shows the distribution of the log ratio of the highest to the lowest host price across open model-weeks and of the log ratio of the reseller's price to the provider's own across closed-model listings. For open models, each host enters with its lowest price in the week. On standard listings alone, without quantized or tiered offers, the median log ratio of the highest to the lowest price is 1.50. Reseller listings are matched to the provider's own model by a stem that drops the provider path, region and vendor prefixes, and version suffixes. An undated alias is matched only when the provider itself lists the same alias at the panel price that week. At a given number of hosts, model age, and size, the standard deviation of log host prices changes by −0.18 a year of calendar time (0.05). Within a model it does not change with age (0.02, 0.02), while the cheapest price changes by −0.14 log points a year of age (0.05). Of the reseller listings, 1.5 percent lie more than three times above or below the provider's own price. The median log ratio of the highest to the lowest price fell from 1.81 in 2025 to 1.39 in 2026. Within a model, the entry of hosts lowers

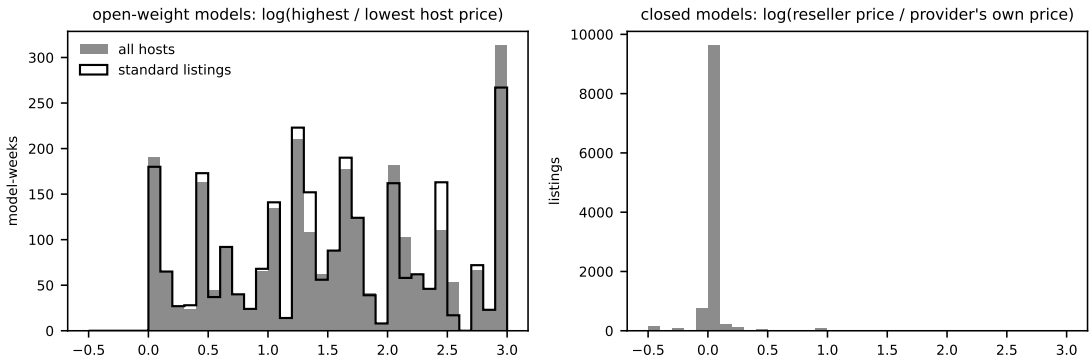

Figure F.8: Price dispersion across hosts of open-weight models (left: log ratio of the highest to the lowest host price per model-week with three or more hosts, all listings and standard listings) and across resellers of closed models (right: log ratio of the reseller's price to the provider's own per listing). Values beyond the axis range are shown at its edges.

the cheapest price, by 37 percent when the number of hosts doubles (−0.66 per log host, 0.14), but does not narrow the spread: the standard deviation of log host prices does not fall as hosts enter (0.08, 0.08). The median hyperscaler listing sits about 5 percent above the median listing of its model and week, and the developer's own API sits at the median and is the cheapest host in 19 percent of model-weeks. Measured attributes explain only part of the spread: quantization, tier tags, context windows, and host type account for 23 percent of the variance of listing prices within a model and week, with an FP8 listing 41 percent cheaper than a standard one and a hyperscaler's listing 2.2 times an inference host's for the same model and week, and the median spread between the highest and the lowest listing narrows only from 1.47 to 1.42 log points once they are held fixed. The multiple for hyperscalers is an average gap to inference hosts with these attributes held fixed, so it differs from the median comparison above. The closed resellers' systematic deviations from the provider's own price are surcharges for regional deployments, about 10 percent in regional data zones, the European ones included (10 percent), and 20 percent in US government regions. Against the median host the premium's trend is about the same with and without the screen (−0.14 and −0.15).

Two benchmark markets. The LLM market combines a rival whose quality jumps and an alternative that is close to free. Two older markets each isolate one of them (Figure F.9). The LLM incumbents neither raise prices when the free alternative improves, as brands do after generic entry, nor cut them broadly. Their rare cuts tend to fall on the versions nearest to it, and they retire the versions it overtakes, although overtaken versions are also old ones. The two paragraphs that follow give the detail.

Microprocessors: a rival quality jump that is not free. When AMD launched its Ryzen processors in March 2017, it closed most of the performance gap with Intel at lower prices. We collect the launch prices and specifications of 545 Intel and 111 AMD desktop processors from 2011 to 2024 and estimate time-dummy hedonic regressions of the log launch price on log cores, log clock speed, simultaneous multithreading, and a high-end desktop indicator, with half-year effects for each vendor. Intel's quality-adjusted launch prices were flat for six years before Ryzen (log slope 0.02 a year) and changed by -0.25 log points a year between 2017 and 2020, a fall of 22 percent a year (Figure F.9). The adjustment ran through new generations. Intel's top unlocked mainstream part cost about \$85 per core in 2015 with four cores; the generations that followed Ryzen kept the price points and added cores, to \$60 per core in 2017 and \$49 in 2020. At equal specifications Intel's launch prices exceeded AMD's by 0.43 log points in 2017 and 0.52 in 2018, a premium that had fallen to 0.14 by 2020. Like the closed LLM providers, the incumbent answered a quality jump of its rival with new versions on a lower schedule. We do not track Intel's recommended prices of existing parts over time, so this benchmark shows the launch margin but not the absence of repricing.

Pharmaceuticals: an entrant that is nearly free. Generic entry brings a substitute of equal quality at a fraction of the price. From the Medicaid State Drug Utilization Data we build quarterly national series of units and reimbursement for 13 brand drugs that faced first generic entry between 2011 and 2019, with entry dated from the data as the first quarter in which generics reach 5 percent of the molecule's units. In the four quarters before entry brand reimbursement per unit rose 0.15 log points. In the four quarters after, it rose 0.05,

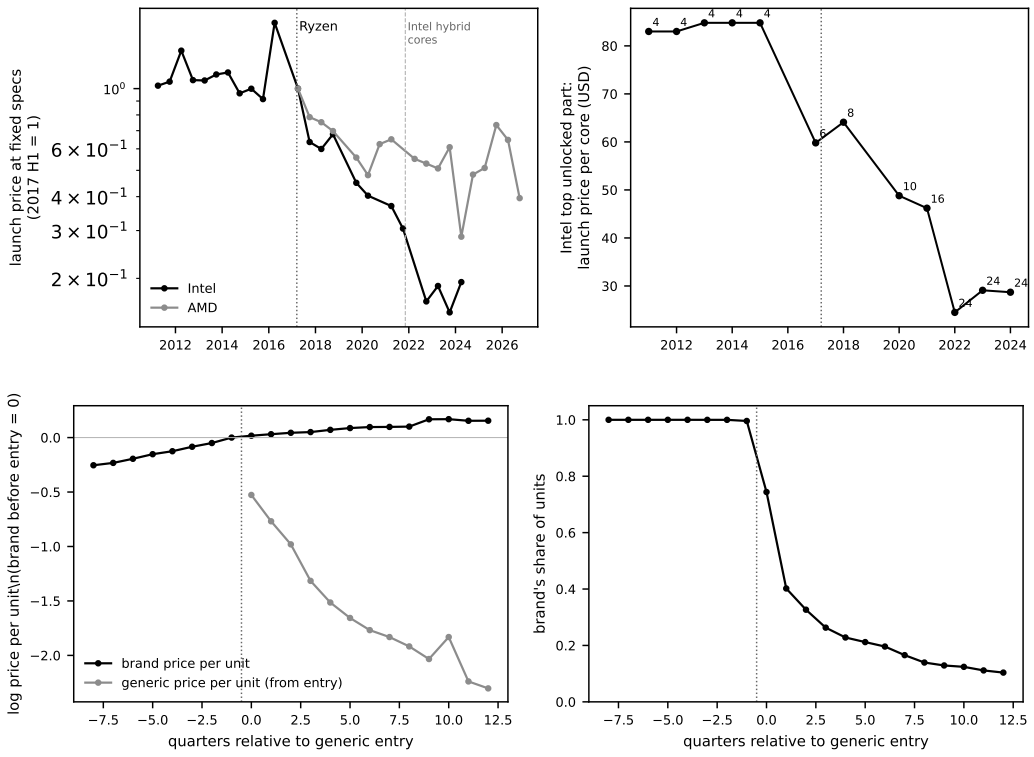

Figure F.9: Benchmark markets. Top left: time-dummy hedonic launch-price indexes for Intel and AMD desktop processors (2017 H1 = 1); the dashed vertical marks Intel's hybrid-core generations, where counting efficiency cores as cores overstates quality. Top right: launch price per core of Intel's top unlocked mainstream processor, with its core count. Bottom left: brand and generic price per unit relative to the brand's price in the quarter before generic entry, mean over brands. Bottom right: the brand's share of the molecule's units. Sources: Wikipedia processor lists; Medicaid State Drug Utilization Data.

and it did not fall for 85 percent of the brands. Four quarters after entry the brand kept 23 percent of units, and generics sold at −1.51 log points relative to the brand's pre-entry price (Figure F.9). This is the segmentation response that Frank and Salkever (1992) model and that Frank and Salkever (1997) and Regan (2008) document: the brand keeps its price for the buyers who stay and lets the rest go. The LLM incumbents do not behave this way. Their prices do not rise when the free alternative improves. In the model, the difference is the distribution of buyers at the margin (Remark B.1): insured, prescriber-driven demand for a branded drug has a segment that does not respond to price, while developers choosing a model for a task form a continuum.