

Evaluating Machine Learning Forecasts of AI-Related Google Trends: Sparse Data and Predictive Uncertainty

Barak Or

ArtificialGate Ltd., Israel

barak@artificialgate.ai

Abstract

We evaluate fourteen probabilistic forecasting procedures on six artificial intelligence (AI)-related Google Trends indices using 48 four-week-ahead targets per query. For Agentic AI, Ridge records a mean ranked probability score (RPS) of 4.141 versus persistence's 6.631, a 37.6% reduction, and mean absolute error (MAE) of 4.771 versus 7.521. However, its nominal 95% prediction intervals cover only 24 of 48 targets. For sparse Physical AI, random forest has the lowest observed RPS among the fourteen procedures (5.095), but intervals average 78 index points wide. Hurdle Beta covers all 45 zeros and none of the three positives without fitting either supervised component. For both sparse queries, tree validation uses only constant-response branches. No procedure leads across all six series. These exploratory findings connect score rankings to interval behavior, executed components and validation support; a restricted normalization analysis separately bounds earlier inputs compatible with retrospectively retrieved data.

Keywords. machine learning; Google Trends; probabilistic forecasting; forecast evaluation; predictive uncertainty; sparse time series.

1 Introduction

Machine learning procedures are commonly compared through their forecasts, but sparse series can make fitting and selection records equally important. A procedure may forecast entirely through fallback rules, or fit during evaluation after validation has used only a constant-response branch. Repeated zeros can yield high interval coverage without informative forecasts of positive values. Retrospective retrieval raises a further question: whether the observed historical series identifies the inputs available at the original forecast dates.

Artificial intelligence (AI)-related Google Trends indices provide a setting in which these issues can be examined together. Some query histories remain positive throughout the observed period; others contain long runs of reported zeros and few positives. Google Trends reports a relative index, and zero does not establish that no searches occurred [11]. Search data have been used to predict external outcomes [3], and recent work studies preprocessing to restore their forecasting value [5]. Here the published index is itself the forecast target. The task therefore connects search-trend predictability [23] with measurement issues arising from normalization, rounding and retrieval variation [6, 20, 27].

We evaluate complete forecasting procedures rather than isolate a single learning algorithm. Table 1 relates this evaluation to prior work on forecast assessment, probabilistic calibration, model multiplicity and search-data measurement. Occurrence-magnitude modelling for intermittent series supplies a further connection [25], while the present analysis distinguishes fitted components from empirical fallbacks.

The main contribution of this work

We make three linked contributions:

- **Joint assessment of forecast scores and model execution.** We compare fourteen procedures through predictive scores, interval behavior, target composition and records of the components actually fitted.
- **Analysis of choices unresolved by validation.** We identify exact tuning ties and untested fitted branches, and assess equal-weight pools of tied empirical recency settings.
- **Conditional identification of earlier inputs.** We characterize compatible earlier scales and rounded inputs under fixed-path normalization assumptions, without claiming reconstruction of historical Google releases.

Section 2 describes the data, procedures and evaluation; Section 3 presents the findings. Sections 4 and 5 discuss their implications and state the main conclusions and future directions. Appendices A-E give implementation details and derivations.

Table 1. Positioning relative to established evaluation and measurement work. The contribution concerns the specific evidence and identification problem examined here.

Prior work	Scope of the present study
Forecast evaluation. Hewamalage et al. [14] review pitfalls in point-forecast evaluation, including partitioning, error measures and statistical comparisons. Calibration and sharpness. Gneiting et al. [10] distinguish these properties and combine distributional diagnostics with proper scoring rules. Multiplicity and pooling. Watson-Daniels et al. [26] study risk-estimate variation across near-optimal classifiers; Gneiting and Ranjan [9] analyze distributional aggregation. Google Trends measurement. West [27] calibrates queries using anchors and rounding-error bounds. Eichenauer et al. [6] reconcile frequencies and reduce sampling variation; Medeiros and Pires [20] study retrieval variation and averaging.	Link matched predictive distributions to training support, executed components and the fitted branches exercised during validation. Explain score and interval behavior jointly with target counts, including the different contributions of zeros and positives to coverage and loss. Examine exact validation ties and later forecast variation within declared empirical recency grids, then assess equal-weight pools of the tied settings. Characterize earlier prefix scales and rounded coordinates compatible with one separately normalized final series, conditional on fixed latent values and deterministic rounding.

2 Data and methods

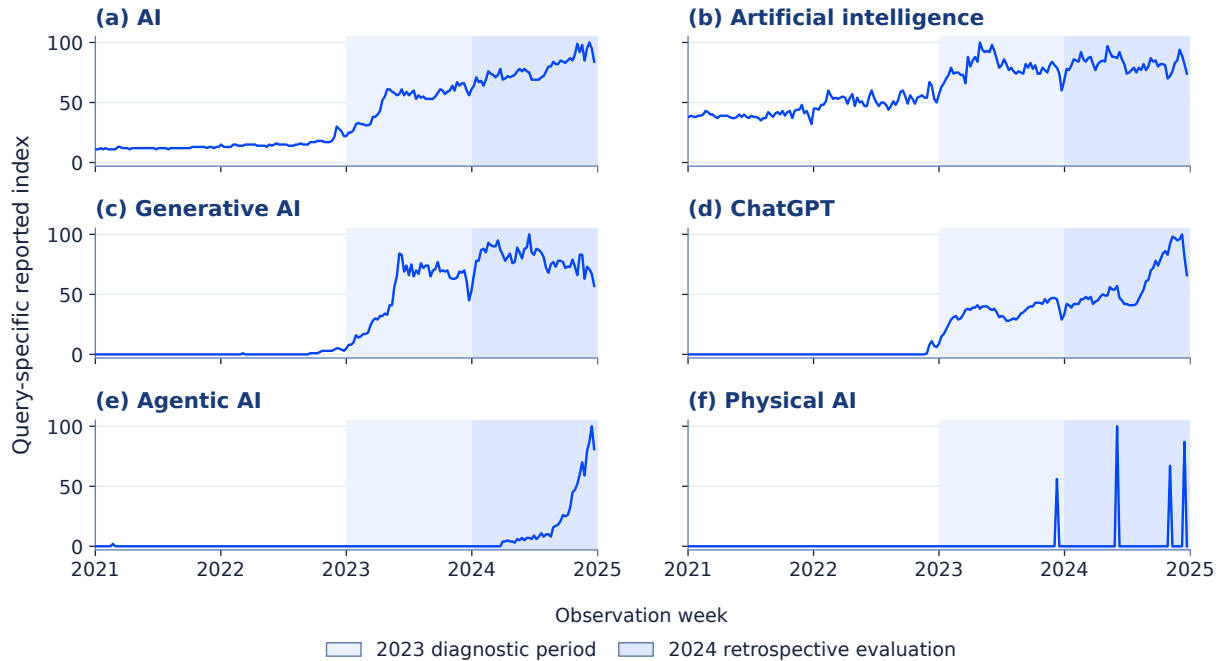


Figure 1. Six separately normalized weekly Google Trends indices. Shading marks the 2023 diagnostic period and the 2024 evaluation period. The last observation is labelled 22 December 2024; 2025 marks the right axis boundary, not additional data. Retrieval date: 8 October 2026. Separate requests have separate index scales.

2.1 Data and forecast target

The six requests are AI, "artificial intelligence", "generative ai", ChatGPT, "agentic ai", and "physical ai". These are search strings, with quotation marks defining phrases, not Google topic identifiers [12]. All use Worldwide, Web Search, and All Categories settings. Each series contains 208 consecutive complete weeks, labelled from 3 January 2021 to 22 December 2024;

zeros are retained. The official visible chart tables were retrieved on 8 October 2026. Raw responses, request URLs and timestamps document the retrieval.

The target is the published 0-100 index. Separate requests have separate scales; zero need not mean no searches. Google also reports added statistical noise, particularly visible for low-interest queries [11]. A positive index does not independently establish adoption or a new underlying search event. AI and the artificial-intelligence phrase are positive in all 208 weeks, whereas Physical AI is positive in only four (Figure 1).

2.2 Evaluation design

The primary comparison uses a nominal 104-week training window, a four-week horizon, and 48 weekly targets from 28 January to 22 December 2024. In the retrieved series, training responses never postdate their forecast origin. Twelve targets in 2023 select hyperparameters, which remain fixed across subsequent refits. Exact validation ties follow the declared grid order. The first validation fit contains 97 supervised pairs; each evaluation fit contains 100 (Appendix A).

The original twelve methods also have one- and eight-week evaluations, an earlier-year diagnostic, and alternative-window checks. The trees enter only the primary four-week comparison. Model extensions and diagnostics followed inspection of earlier outcomes.

2.3 Machine learning and reference procedures

Eight procedures use historical predictors. Their regression, mixture, local-learning and tree-ensemble foundations are reviewed by Hastie et al. [13]; the distributional implementations are specified below. Common predictors are current level, one- and four-week changes, a four-week mean, and a time coordinate; Gaussian mixture regression (GMR) and local analogues use level and four-week change. Estimation and any standardization use only each query’s current training pool. All fourteen procedures return probability mass functions (PMFs) for the same 101 integer outcomes.

Ridge Gaussian (Ridge) fits penalized linear regression and discretizes a Gaussian with spread estimated from training residuals. *GMR* conditions a joint mixture for level, four-week change and response on the current predictors [2]. *Beta without atom* uses logit-Ridge predictions and a selected precision to define a discretized Beta law; its zero-bin probability comes from the continuous lower tail.

Hurdle Beta combines a separate positive-occurrence model with a Beta law for positive values, following the zero-augmented modelling principle of Ospina and Ferrari [22]. Its implementation uses logistic regression for occurrence and logit Ridge for the positive-response location, with empirical shrinkage and fallbacks. *Hurdle logit-normal* retains the same occurrence construction and instead models positive logits with Ridge and Gaussian residual spread. *Local empirical* averages Gaussian kernels around historical responses, weighting observations by proximity in level and four-week change and by recency, with 2% uniform mass.

Random forest leaf distributions average the response distributions within learned leaves and add a smoothing prior, following the conditional-distribution approach of quantile regression forests [21]. *Gradient boosting* estimates the conditional mean with successive regression trees under squared-error loss [7]; a discretized Gaussian with spread from training residuals supplies the probabilities.

Six reference procedures complete the comparison. *Persistence* uses the latest observation as its centre [16], with Gaussian spread estimated from historical horizon-matched differences. *Kalman local trend* propagates a linear Gaussian level-and-slope state to obtain the predictive mean and variance [17]. Three empirical references are defined for this comparison. *Smoothed marginal* uses historical response frequencies with one prior count, split equally between zero and the 100 positive bins collectively. *Discounted marginal* applies exponentially decreasing calendar-age weights to those frequencies while retaining the same prior. *Two-rate empirical hurdle* estimates positive occurrence and the positive-value histogram with separately selected recency half-lives. *Zero-only* assigns all probability to zero, providing a diagnostic for repeated zeros.

Appendix B specifies the probability constructions, fitting thresholds and fallback rules. Two uniform pools of exactly tied empirical recency settings are assessed separately (Section 3.4), outside the fourteen-procedure ranking.

2.4 Forecast assessment

Let $\mathcal{Y} = \{0, \dots, 100\}$ be the outcome support, $p(j)$ the forecast probability of $j \in \mathcal{Y}$, and $y \in \mathcal{Y}$ the observed index. The cumulative distribution function (CDF) at integer threshold k is $F_p(k) = \sum_{j=0}^k p(j)$. Write $\mathbf{1}\{A\} = 1$ when statement A holds and 0 otherwise. The primary measure is the unnormalized ranked probability score (RPS),

$$\text{RPS}(p, y) = \sum_{k=0}^{99} [F_p(k) - \mathbf{1}\{y \leq k\}]^2, \quad (1)$$

which sums squared cumulative-probability errors over thresholds $k = 0, \dots, 99$. Lower values are better. RPS is proper: its expectation under the data-generating distribution is minimized by forecasting that distribution [8].

For N forecast-target pairs (p_i, y_i) , indexed by $i = 1, \dots, N$, reported RPS is $N^{-1} \sum_{i=1}^N \text{RPS}(p_i, y_i)$; $N = 48$ in the primary comparison. Scores are compared within queries because their scales differ. Define the predictive quantile $Q_i(u) = \min\{j \in \mathcal{Y} : F_{p_i}(j) \geq u\}$ for $0 < u < 1$, the median $m_i = Q_i(0.5)$, and the mean $\mu_i = \sum_{j=0}^{100} j p_i(j)$. Mean absolute error (MAE) evaluates predictive medians, and root mean squared error (RMSE) evaluates predictive means:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |m_i - y_i|, \quad \text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\mu_i - y_i)^2}. \quad (2)$$

With $\pi_i = 1 - p_i(0)$, occurrence Brier score is $N^{-1} \sum_{i=1}^N [\pi_i - \mathbf{1}\{y_i > 0\}]^2$ [1]. Mean log loss (negative log predictive density, NLPD) is $-N^{-1} \sum_{i=1}^N \log p_i(y_i)$, using natural logarithms and $+\infty$ when an observed outcome receives zero probability [8].

For nominal interval mass $c \in \{0.80, 0.95\}$, let $L_i = Q_i((1-c)/2)$ and $U_i = Q_i((1+c)/2)$. The inclusive integer interval $[L_i, U_i]$ is assessed by

$$\begin{aligned} \text{Coverage}_c &= \frac{1}{N} \sum_{i=1}^N \mathbf{1}\{L_i \leq y_i \leq U_i\}, \\ \text{Width}_c &= \frac{1}{N} \sum_{i=1}^N (U_i - L_i), \quad \text{Mass}_c = \frac{1}{N} \sum_{i=1}^N \sum_{j=L_i}^{U_i} p_i(j). \end{aligned} \quad (3)$$

Here Mass_c is the mean probability assigned to the intervals, which can exceed c because outcomes are discrete. The coverage gap is $\text{Coverage}_c - \text{Mass}_c$. Group-specific coverage evaluates the same intervals on the zero or positive observed targets.

Zero and positive target target counts accompany these outcome-conditional diagnostics; they do not enter model selection [19].

Ranks and relative score differences summarize the matched evaluation targets. The retained block bootstrap covers 36 Hurdle-Beta contrasts with Beta without atom or Smoothed marginal across six queries and three horizons (Appendix C); it does not assess the Ridge or tree comparisons. Those comparisons establish differences in the observed scores, without establishing lower expected² beyond this record.

3 Results

3.1 Predictive scores and interval performance

The lowest observed mean RPS occurs for different procedures across the six queries (Table 2): Ridge for AI and Agentic AI, persistence for Generative AI, local empirical for the artificial-intelligence phrase, Hurdle Beta for ChatGPT, and the forest for Physical AI. For AI, Ridge's mean RPS is 5.9% below persistence's.

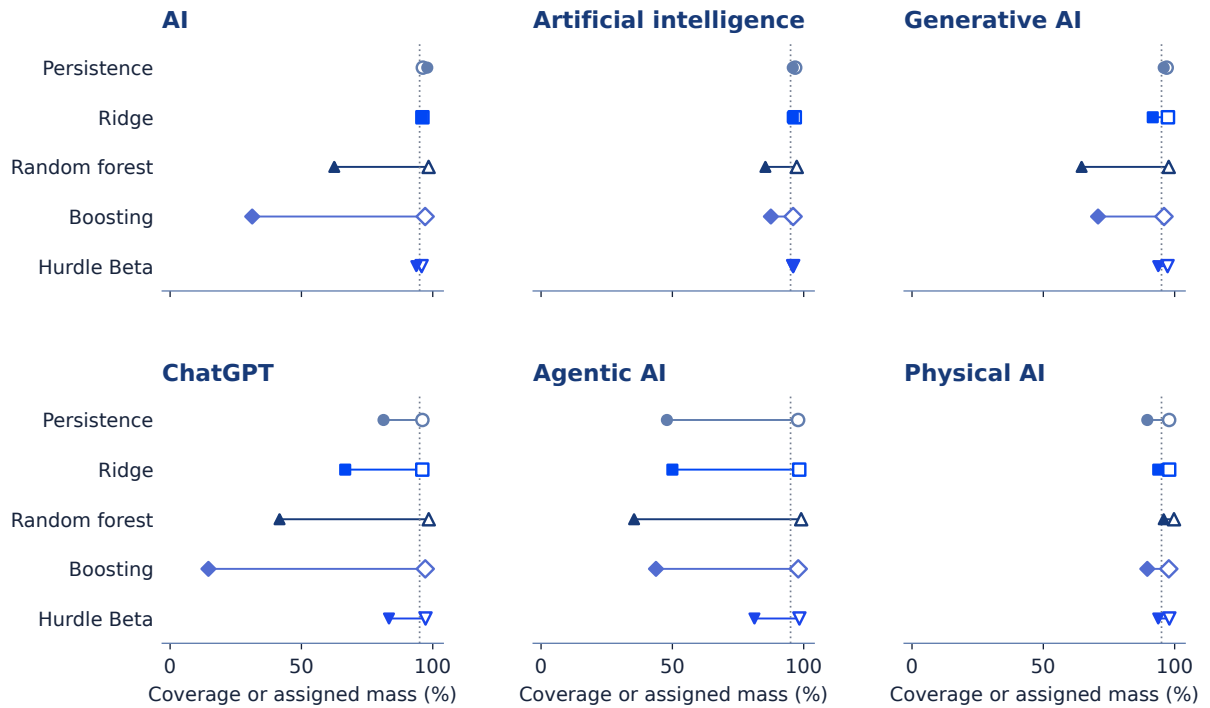
For Agentic AI, Ridge records mean RPS of 4.141, 37.6% below persistence's 6.631; the corresponding MAEs of their predictive medians are 4.771 and 7.521. GMR records a close RPS of 4.153. Ridge's nominal 95% intervals cover 24 of 48 targets, with mean width 4.77 points. Hurdle Beta has higher RPS (5.140) and wider intervals (30.88 points), covering 39 targets. Both have coverage below the nominal level.

Both trees have lower mean RPS than persistence for the artificial-intelligence phrase and Physical AI, and higher RPS for the other four queries. Boosting's nominal 95% intervals cover 7 of 48 ChatGPT targets. Comparison with assigned interval probability shows that discreteness alone does not explain the coverage

Table 2. Four-week forecast comparison: mean ranked probability score (lower is better), with 48 matched targets per query. Bold marks the lowest observed score in each column among these fourteen procedures.

Forecast	AI	AI phrase	Generative	ChatGPT	Agentic	Physical
<i>Feature-based learning procedures</i>						
Ridge Gaussian	3.205	4.154	6.214	6.816	4.141	6.665
Random forest leaf	4.883	4.162	6.798	9.393	12.223	5.095
Gradient boosting	5.347	4.150	6.611	10.385	10.657	7.961
GMR	4.086	4.316	7.754	7.899	4.153	6.021
Beta without atom	3.557	4.420	7.481	10.250	10.943	5.254
Hurdle Beta	3.651	4.220	6.790	5.981	5.140	5.115
Hurdle logit-normal	3.657	4.099	6.153	6.461	4.619	5.122
Local empirical	6.646	3.997	7.527	8.122	8.302	5.210
<i>Statistical, frequency, and zero references</i>						
Persistence	3.407	4.580	5.467	6.135	6.631	9.040
Kalman trend	3.481	6.004	7.762	6.372	5.138	9.275
Smoothed marginal	19.819	6.457	20.847	21.183	18.352	5.128
Discounted marginal	4.773	4.439	6.499	8.853	18.352	5.128
Two-rate empirical	4.773	4.439	6.500	8.850	18.352	5.128
Zero-only	78.792	82.896	80.333	59.854	20.792	5.292

AI phrase is the quoted artificial-intelligence query. Random forest leaf and Gradient boosting denote the two probability constructions described in the text. Hurdle Beta and Hurdle logit-normal use fallback rules throughout Physical AI; neither supervised hurdle component is fitted there.

**Figure 2.** Central 95% prediction intervals. Filled points show empirical coverage; open points show mean probability assigned to the inclusive integer intervals. Connecting lines represent their difference, not confidence limits. The dotted line marks 95%. Table 3 reports Physical AI's target counts and interval widths.

gaps (Figure 2). These contrasts concern complete procedures, including uncertainty construction, and do not isolate nonlinear regression's effect.

3.2 Coverage in the sparse Physical AI series

Physical AI has 45 zero and three positive targets. Hurdle Beta covers every zero and no positive, giving 93.8% aggregate coverage. Always reporting $[0, 0]$ covers the same targets (Table 3).

Table 3. Physical AI: central 95% intervals for 45 zero and three positive targets.

Forecast	RPS	Covered	Positive	Width
Zero-only	5.292	45/48	0/3	0.00
Hurdle Beta	5.115	45/48	0/3	5.58
Random forest	5.095	46/48	1/3	78.00
Two-rate pool	5.095	45/48	0/3	39.56

Covered counts all targets; Positive counts positive targets only. Width is the mean in index points. Forest and pool RPS are 5.0955 and 5.0949, respectively; the pool is outside the fourteen-procedure ranking.

Table 4. Supervised fits at the 48 four-week evaluation origins. The tree counts apply separately to the forest and boosting.

Query	HB occurrence	HB positive	Each tree
AI	0	48	48
Artificial intelligence	0	48	48
Generative AI	27	48	48
ChatGPT	37	48	48
Agentic AI	30	28	35
Physical AI	0	0	48

HB denotes Hurdle Beta. Its columns count logistic and positive-response Ridge fits. Trees use constant-response shortcuts at the remaining origins. For AI and the artificial-intelligence phrase, all-positive training responses explain zero occurrence fits. Physical AI has insufficient positives for either supervised hurdle component.

The forest covers one additional target, with intervals averaging 78.0 points on the 0-100 scale. Its sole covered positive lies inside the full-support interval $[0, 100]$, which provides no information about that magnitude. Its observed mean RPS is the lowest among the fourteen procedures and is 0.39% below Hurdle Beta's. The separate two-rate pool has similar RPS and covers no positive target.

The nominal 95% level refers to the full predictive distribution. The 0/3 count explains aggregate coverage; it does not evaluate intervals from the conditional-positive distribution.

Positive targets contribute 5.087 to Hurdle Beta's mean RPS of 5.115; zeros contribute 0.028. Each contribution is the subgroup mean weighted by its fraction of the 48 targets. Zeros account for coverage, whereas errors on the three positives account for most of RPS.

3.3 Supervised fits and empirical fallbacks

Hurdle Beta's occurrence classifier requires at least six positive and six zero training responses; its positive-response regression requires at least eight positives. Otherwise, a declared empirical or default rule supplies the component. Table 4 reports supervised fits alongside tree counts.

Physical AI's training pools contain one to three positives. Neither supervised hurdle component is fitted at any evaluation origin, although fallbacks use the training observations. Both trees are fitted at all 48 origins; fitting alone does not establish a stable conditional relationship from sparse responses. For Agentic AI, occurrence and positive-response components are fitted at 30 and 28 origins, respectively, and each tree at 35. In the two dense queries, all-positive pools explain the absence of occurrence fits, while positive regression is fitted throughout.

Holding Hurdle Beta's saved positive distributions and settings fixed without refitting, replacing its occurrence probabilities with smoothed historical rates raises Agentic AI RPS from 5.140 to 13.086 and occurrence Brier score from 0.203 to 0.576. Both favor the retained probabilities. This agreement is empirical: an RPS gain can also compensate for error in the positive-value distribution (Eq. (13)). Neither contrast alone establishes calibration.

The substitution raises RPS from 6.790 to 8.638 for Generative AI and from 5.981 to 8.137 for ChatGPT. Their evaluation targets are all positive, so these cases concern continued reported activity.

3.4 Model choices unresolved by validation

Agentic AI and Physical AI have all-zero validation targets and all-zero responses in the supervised validation training pools. The five discounted-marginal settings tie exactly, as do the 25 two-rate empirical settings and each tree's four settings. Every tree validation forecast uses a constant-response shortcut: four

Table 5. Exactly tied empirical settings can yield different later forecasts. RPS is evaluated on the same 48 targets per query.

Query	Family	Tied settings	Selected RPS	Later RPS range	Pooled RPS
Agentic AI	Discounted marginal	5	18.352	9.917-18.352	14.410
Agentic AI	Two-rate empirical	25	18.352	9.783-18.352	14.750
Physical AI	Discounted marginal	5	5.128	5.085-5.240	5.101
Physical AI	Two-rate empirical	25	5.128	5.082-5.208	5.095

Selected denotes the first configuration in the declared grid among exact validation minima. The later range is a diagnostic of unresolved selection. Pooling averages all tied PMFs within the stated empirical family.

configurations at twelve origins. Validation therefore never exercises the fitted branches used later at 35 Agentic-AI and 48 Physical-AI evaluation origins per tree.

Grid order makes selection reproducible but does not distinguish tied specifications empirically. Agentic AI’s five tied discounted settings later yield RPS values from 9.917 to 18.352; Table 5 reports both empirical families. These ranges reveal unresolved selection.

The separate pooling analysis averages full forecasts over all exactly tied settings within each empirical family. Equal weights were declared before calculation, after earlier results had been inspected. Agentic AI’s discounted-marginal RPS falls from the canonical selection’s 18.352 to 14.410 for the pool, still above Ridge’s 4.141. Physical-AI pooling widens intervals without covering another positive target. Tree settings are not pooled.

The squared-loss variance identity ensures that pooled RPS cannot exceed mean candidate RPS (Appendix D); improvement over a particular candidate and calibration are not guaranteed. Pooling removes candidate-order dependence in exact arithmetic, while retaining dependence on the grid and duplicate representations. It expresses unresolved choice without supplying positives absent from validation.

3.5 Recoverability under retrospective normalization

The measurement analysis asks which earlier inputs are compatible with a final rounded series. Assume a strictly positive latent path $X_1, \dots, X_T$, held fixed across request windows, and let $M_T = \max_s X_s$. Reported values are $y_s = \lfloor 100X_s/M_T + 1/2 \rfloor$. This deterministic half-up rounding model excludes sampling, suppression, filtering and other revisions. A prefix begins at the first observation and ends at a forecast origin; it is distinct from the rolling 104-week estimation window.

Proposition 1 (Compatible earlier normalization). *Let $z_t = \max_{s \leq t} y_s$ be the largest reported value in a prefix, and let $\lambda_t = M_T / \max_{s \leq t} X_s$ be its rescaling multiplier. If no reported 100 follows the prefix, $\lambda_t = 1$ and the rounded prefix is identified exactly. Otherwise, the exact feasible set is*

$$\Lambda_t = \begin{cases} (200, \infty), & z_t = 0, \\ \left(\frac{100}{z_t + 0.5}, \frac{100}{z_t - 0.5} \right], & 1 \leq z_t \leq 99, \\ [1, 200/199], & z_t = 100. \end{cases} \quad (4)$$

If the prefix contains a reported 100, rescaling changes each unrounded index by at most 0.5 points and each rounded value by at most one integer step.

Appendix E proves attainability and gives coordinate bounds with their endpoint conventions. The result states when the prefix scale is identified and gives its feasible range otherwise. For example, the final sequence $(0, 0, 100)$ permits any rounded value from 0 to 100 for the second coordinate of its two-observation prefix. Coordinate-wise feasible sets cannot generally be combined freely because all coordinates share the same normalization denominator.

Under this submodel, saved diagnostics identify the complete rounded prefix at all 48 origins for the artificial-intelligence phrase, at 24 for Generative AI, and at 26 for Physical AI, but at none for AI, ChatGPT, or Agentic AI. No origin has an unbounded multiplier. Agentic AI’s largest recorded upper bound is nevertheless 66.7; the artificial-intelligence phrase has multiplier one throughout.

The 66.7 bound describes compatibility under the stated assumptions; it neither estimates an actual historical revision nor measures the resulting forecast-score bias.

4 Discussion

The comparison supports two related interpretations. First, a lower observed distributional score does not by itself establish adequate interval coverage. Agentic AI illustrates this distinction through Ridge’s score and coverage, while Physical AI shows how zeros can supply aggregate coverage even when positive values dominate scoring loss. Subgroup counts explain the aggregate result; they do not impose the full distribution’s nominal coverage requirement on each outcome-defined subgroup.

Second, model rankings need to be read with fitting and selection records. For Physical AI, Hurdle Beta uses declared fallbacks throughout evaluation, whereas both tree procedures are fitted at every origin. Their validation forecasts nevertheless use only constant-response branches. Exact validation ties likewise leave the later performance of empirical recency settings unresolved. These records distinguish reproducible selection from validation that actually discriminates among the fitted behaviors used later.

The normalization analysis concerns a separate source of uncertainty: which earlier inputs are compatible with one retrieved series under explicit measurement assumptions. These compatibility sets are not prediction intervals and do not estimate actual Google revisions or their effect on forecast scores. Interpreting the empirical ranking therefore requires both a clear description of the executed procedure and the provenance of its inputs.

Limitations. The evidence comprises six related queries, 48 dependent targets per query and compact tuning grids, with only three positive evaluation targets for Physical AI. Model extensions and diagnostics followed inspection of earlier results, leaving no untouched confirmation period. Comparisons evaluate complete probabilistic implementations, preventing attribution of score or coverage differences to the learning mechanism alone. The October 2026 retrieval does not recover historical publication vintages. The restricted normalization model leaves actual revisions and any resulting score bias unmeasured.

5 Conclusion

Two findings summarize this retrospective comparison of fourteen procedures. First, a lower observed forecast score need not imply adequate prediction-interval coverage: for Agentic AI, Ridge’s mean RPS is 37.6% below persistence’s, yet its nominal 95% intervals cover only 24 of 48 targets. For Physical AI, Hurdle Beta covers all 45 zeros and none of the three positives. Second, fitting and validation records are essential to interpreting a model ranking: Hurdle Beta uses only fallbacks for Physical AI, while validation does not exercise the tree branches fitted during subsequent evaluation. No procedure has the lowest observed RPS across all six queries.

Future work will evaluate independent query families using data snapshots and forecasts saved prospectively, obtain more positive outcomes for sparse-series assessment, and examine validation designs that exercise the components expected to run in use. These steps would test how well the present findings generalize beyond the retrieved series.

A Forecast origins and training information

Index the 208 observed weeks by $t = 0, \dots, 207$. A direct h -week forecast uses the predictor vector

$$Z_t = (Y_t, Y_t - Y_{t-1}, Y_t - Y_{t-4}, \bar{Y}_{t-3:t}, (t - 104)/104) \quad (5)$$

and the supervised training pool

$$\mathcal{D}_{t,h,W} = \{(Z_s, Y_{s+h}) : \max(4, t - W + 1) \leq s \leq t - h\}. \quad (6)$$

Each training response is therefore dated no later than the origin within the retrieved series. Sunday labels mark the start of weekly intervals: the first main origin, 31 December 2023, includes the week ending 6 January 2024. These labels refer to observation weeks, not historical publication times. The nominal window W bounds training-origin indices; lagged predictors can extend four weeks before its lower bound. Standardization uses the current training pool. Persistence and Kalman use up to W available raw observations. Each query is fitted separately, without external predictors.

The primary design sets $W = 104$ and $h = 4$, with validation origins $\text{range}(104, 152, 4)$ and evaluation origins $\text{range}(156, 204)$. There are 12 validation targets and 48 evaluation targets, with 100

supervised pairs at each evaluation origin. The first validation fit has 97 pairs because four initial lags are required. A configuration is ineligible if any validation forecast fails; a run stops if none is eligible. Each query and method selects the eligible configuration with the lowest mean validation RPS, retaining the first declared configuration in an exact tie. These settings remain fixed during refitting at successive evaluation origins.

Only the original twelve procedures have $h = 1, 8$ checks, the separately tuned 2023 diagnostic, and the $W = 52, 78$ comparisons. The supporting materials supply their schedules and candidate grids. Every analysis uses the 8 October 2026 retrieval, so chronological fitting does not reconstruct the information available from Google in 2024. The model extensions and diagnostics followed inspection of earlier results and remain exploratory.

B Probability forecasts and fallback rules

B.1 Common outcome grid

Every forecast is a PMF on $0, \dots, 100$. Gaussian distributions are integrated over nearest-integer bins; bins 0 and 100 absorb the exterior tails. Persistence, Ridge, Kalman, boosting, GMR, non-atom Beta, both hurdle procedures and local analogues floor final bin masses at 10^{-12} and renormalize. Hurdle procedures also floor and normalize their positive components. Empirical references and the forest use their explicit prior without an additional floor; zero-only is the exact atom δ_0 . All scores use the stored PMFs. Log loss is infinite when the observed outcome has zero assigned probability.

B.2 Regression, mixtures, trend and analogues

Persistence centers a Gaussian at Y_t , with spread given by the root mean square historical h -step difference, floored at 1.5 index points. Ridge regresses the future index on standardized Z_s and uses its training residual root mean square, also floored at 1.5. These plug-in spreads omit coefficient uncertainty.

The non-atom Beta method fits Ridge to the logit-transformed response using all observations, after clipping unit-scale values to $[0.005, 0.995]$. The fitted location and selected precision determine Beta shapes, each floored at 0.05. The zero bin receives the continuous mass below 0.005; there is no separate occurrence model.

GMR fits a full-covariance Gaussian mixture to standardized $(Y_s, Y_s - Y_{s-4}, Y_{s+h})$ and conditions on the two predictors [2]. Its grid contains one to three components and two covariance regularizers. Fitting uses two expectation-maximization (EM) initializations, a fixed seed and a 10^{-4} floor on conditional standardized variance. For n training observations and K components, it falls back to persistence when $n < 3K$ or the response standard deviation is below 0.05. The shared Gaussian binning routine floors its input scale at 0.7; Ridge and persistence apply the stronger floor specified above.

The Kalman reference has level and slope states, diagonal process covariance and Gaussian observation noise [17]. Each origin starts from the first window observation, zero slope and covariance $\text{diag}(1000, 1000)$. It then refilters the history using the Joseph covariance update and propagates h steps. Eight variance settings are compared.

Local empirical analogues weight training responses by standardized distance in current level and four-week change, together with exponential recency decay. Their forecast mixes discretized Gaussian response kernels and adds 2% uniform mass. A total unnormalized analogue weight at most 10^{-20} triggers uniform weights. All candidate grids are included in the supporting materials.

B.3 Hurdle learning and sparse-data defaults

The two hurdle models separate reported occurrence from the positive value [22]. Before numerical flooring,

$$p_{\pi,q}(k) = \begin{cases} 1 - \pi, & k = 0, \\ \pi q(k), & 1 \leq k \leq 100. \end{cases} \quad (7)$$

The mixture weight describes positive occurrence and the positive component its conditional PMF. For n training responses with n_+ positives, let $r = (n_+ + 1/2)/(n + 1)$. Logistic regression is fitted only with at least six positive and six zero responses. Its fitted probability $\hat{\pi}$ gives $\pi = \text{clip}(0.8\hat{\pi} + 0.2r, 0.001, 0.999)$; otherwise $\pi = r$. Hurdle logit-normal additionally clips the fallback probability to $[0.001, 0.999]$, a step that is inactive at the observed training sizes.

Hurdle Beta fits positive-response logit Ridge only when $n_+ \geq 8$. The inverse-logit prediction receives weight $\min(1, n_+/25)$, with the remainder on the empirical positive mean. With one to seven positives, only the empirical mean is used; with none, the unit-scale mean is 0.05. The final mean is clipped to $[0.007, 0.993]$. The selected precision defines Beta shapes with the same 0.05 floor.

Hurdle logit-normal uses the occurrence rule above and the eight-positive threshold, but fits a Gaussian on positive logits. Its residual scale is floored at 0.18. Smaller positive samples use empirical logit moments with a 0.25 scale floor; an empty sample uses location -3 and scale 1. Both positive regressions clip unit-scale training responses to $[0.005, 0.995]$ before applying the logit transform.

For fitted unit-scale positive CDF G , both methods use

$$\begin{aligned} q(1) &= G(0.015), \\ q(k) &= G((k+0.5)/100) - G((k-0.5)/100), \\ &\quad 2 \leq k \leq 99, \\ q(100) &= 1 - G(0.995). \end{aligned} \tag{8}$$

The lower tail is thus assigned to 1, rather than conditioning the continuous distribution on exceeding 0.005. Component diagnostics reconstruct $\pi = \sum_{k>0} p(k)$ and $q(k) = p(k)/\pi$ from the stored PMF, retaining the effect of numerical flooring. The matched occurrence diagnostic replaces only the reconstructed π by r , holding q fixed. It does not retrain the method or identify a causal benefit of a zero atom.

B.4 Tree ensembles

Both added methods use the raw five predictors, seed 20261020, and only the primary four-week design. The forest has 200 bootstrap trees with squared-error splits and unrestricted depth. Its ordered (minimum leaf size, feature fraction) options are $(10, 1.0)$, $(10, 0.6)$, $(5, 1.0)$, $(5, 0.6)$. Bootstrap samples determine the partitions. Every training row is then routed through each tree without bootstrap multiplicity. For terminal leaf $L_b(x)$, the conditional weights are

$$w_i(x) = \frac{1}{200} \sum_{b=1}^{200} \frac{\mathbf{1}\{L_b(x_i) = L_b(x)\}}{\sum_j \mathbf{1}\{L_b(x_j) = L_b(x)\}}. \tag{9}$$

With responses z_i , $\alpha_0 = 1/2$ and $\alpha_k = 1/200$ for $k > 0$,

$$p(k | x) = \frac{n \sum_i w_i(x) \mathbf{1}\{z_i = k\} + \alpha_k}{n + 1}. \tag{10}$$

This construction follows the conditional-distribution principle of quantile regression forests [21]. Its mean need not equal the library forest's bootstrap-weighted point prediction.

Gradient boosting uses squared-error loss, learning rate 0.05, minimum leaf size 5, full subsampling and no early stopping [7]. The ordered (depth, stages) options are $(1, 50)$, $(1, 100)$, $(2, 50)$, $(2, 100)$. For fitted mean function $\hat{f}$, the Gaussian forecast uses

$$\begin{aligned} \mu &= \hat{f}(Z_t), \\ \sigma &= \max \left\{ \sqrt{\frac{1}{n} \sum_i [z_i - \hat{f}(Z_i)]^2}, 1.5 \right\}. \end{aligned} \tag{11}$$

The location is not clipped; moments and scores use the binned PMF. Training residuals do not establish predictive calibration. For constant-response pools, both tree methods use exact distributionally equivalent shortcuts. These branches are recorded separately from estimator fits.

B.5 Empirical recency references

The smoothed marginal uses $(n_k + \alpha_k)/(n + 1)$. The discounted marginal replaces counts by weighted counts, with calendar-age weights proportional to $2^{-a_i/H}$ and normalized to sum to n . The two-rate empirical hurdle chooses half-lives H_π, H_q separately. Its occurrence probability is the weighted positive count plus $1/2$, divided by $n + 1$. Its positive histogram uses weights summing to n_+ and adds $1/200$ per positive bin, with denominator $n_+ + 1/2$. An empty positive pool is uniform.

The ordered half-life grid is $\{\infty, 4, 13, 26, 52\}$ weeks. Infinite half-lives recover the smoothed marginal in both families; equal finite half-lives do not generally make the families equivalent. The grid and prior conventions are fixed. These finite-window references do not replicate the Teunter-Syntetos-Babai (TSB) point-forecast recursion [24].

C Scores, intervals and inferential limits

The unnormalized RPS sums threshold Brier scores [8]:

$$\text{RPS}(p, y) = \sum_{k=0}^{99} \{F_p(k) - \mathbf{1}(y \leq k)\}^2. \quad (12)$$

MAE uses the predictive median; RMSE uses the predictive mean. Occurrence Brier score compares $\pi = 1 - p(0)$ with $\mathbf{1}(y > 0)$. For the hurdle form, log loss is $-\log(1 - \pi)$ at zero and $-\log \pi - \log q(y)$ otherwise. Zero- and positive-target summaries explain the aggregate score without serving as separate selection objectives [19]. Each group's contribution is its within-group mean multiplied by its share of all targets.

For $S_q(k) = \Pr_q(X > k)$, define $A_q = \sum_{k=0}^{99} S_q(k)^2$ and $B_q(y) = \sum_{k=0}^{y-1} S_q(k)$, taking the empty sum to be zero. Expansion of the threshold squares gives

$$\text{RPS}(p_{\pi, q}, y) = y - 2\pi B_q(y) + \pi^2 A_q. \quad (13)$$

If the true distribution is $(1 - r)\delta_0 + r q^*$, with its conditional component also supported on the positive grid, then for fixed q the expected RPS is minimized at $[r \sum_k S_q(k) S_{q^*}(k) / A_q]_0^1$, where brackets clip to $[0, 1]$. Changing π can therefore reduce RPS by compensating for positive-shape error. The occurrence Brier score assesses occurrence separately. These identities are consequences of the standard quadratic score.

For nominal mass c , the inclusive interval is $[F^{-1}((1 - c)/2), F^{-1}((1 + c)/2)]$, with each quantile defined as the smallest integer reaching its probability. The sum of PMF entries within the interval is its assigned mass, which can exceed c . The audit reports coverage, width, assigned mass and the mean difference between coverage indicators and assigned mass. Positive-target coverage evaluates the same full-PMF intervals on the observed positive subset. It does not evaluate intervals constructed from q , and the nominal level is not a coverage requirement within each outcome-defined subgroup.

Discrete probability integral transform (PIT) histograms distribute each observation analytically over $[F(y - 1), F(y)]$, rather than drawing jitter [4]. An observed outcome assigned zero mass contributes a point at $F(y)$ and is explicitly counted. Group sizes are retained; empty groups have no estimated coverage.

The retained bootstrap covers only 36 original Hurdle-Beta contrasts against non-atom Beta or smoothed marginal across six queries and three horizons. It uses 4,000 centered circular-block replicates at lengths 4, 8 and 12, pointwise basic intervals, and Holm-adjusted left-tail comparisons within each block length [15, 18]. It conditions on saved forecasts without refitting or reselection. Abrupt changes, dependence, retrieval variation and prior outcome inspection limit interpretation. The Ridge and tree comparisons against persistence are outside its scope.

D Exact validation ties and predictive pooling

For each empirical family, let V_j be a configuration's original mean validation RPS. Define $\mathcal{T} = \{j : V_j = \min_{\ell} V_{\ell}\}$ by exact stored equality, with no tolerance. This is a set of tied grid configurations, not a confidence set; it depends on the finite grid and numerical precision. The set remains fixed across the 48 evaluation origins. Its uniform pool is

$$\bar{p}_t = \frac{1}{m} \sum_{j \in \mathcal{T}} p_{j,t}, \quad m = |\mathcal{T}|. \quad (14)$$

The two families are pooled separately, with all members using the same training cutoff. Configuration multiplicities are retained. Uniform weights express the declared grid convention, not posterior model probabilities.

For member and pooled CDFs, the variance identity gives

$$\begin{aligned} & \frac{1}{m} \sum_j \text{RPS}(p_{j,t}, y) - \text{RPS}(\bar{p}_t, y) \\ &= \frac{1}{m} \sum_j \sum_{k=0}^{99} \{F_{j,t}(k) - \bar{F}_t(k)\}^2 \geq 0. \end{aligned} \quad (15)$$

Expanding $(F_j - I)^2$ around the mean CDF proves the identity: the cross terms sum to zero. The $k = 0$ term is the variance of member occurrence probabilities and is already included in the full RPS gap. Within the pooled positive component, the weights are $\pi_j/(m\bar{\pi})$ when $\bar{\pi} > 0$, with the barred probability denoting the pool's positive mass. Members with zero occurrence probability are omitted. Directly averaging positive components generally produces a different PMF. If $\bar{\pi} = 0$, the positive component is undefined.

The identity requires no independence. The pooled RPS is no larger than the mean member RPS; it need not beat the canonical or best member or improve calibration, interval coverage or point-forecast accuracy [9]. Tied validation scores need not imply identical forecasts, so pooling can lower the score on those same validation targets. That reduction is an in-sample convexity consequence, not an independent validation of the pooling rule. The pool is order-invariant in exact arithmetic but can change when grid points are duplicated. The pooling rule was developed after inspecting earlier results.

E What deterministic normalization identifies

Assumptions and compatibility. Let $X_1, \dots, X_T$ be a strictly positive latent search-interest path before request-window rescaling, held fixed across normalization windows. Put $M_T = \max_s X_s$. The submodel is

$$\begin{aligned} q_s &= X_s/M_T > 0, & \max_s q_s &= 1, \\ y_s &= \lfloor 100q_s + 1/2 \rfloor. \end{aligned} \quad (16)$$

Rounding is deterministic and half-up. Sampling, suppression, filtering and other retrieval changes are excluded. The corresponding rounding cells are

$$\begin{aligned} I_0 &= (0, 0.005), & I_{100} &= [0.995, 1], \\ I_k &= [(k-0.5)/100, (k+0.5)/100], & 1 \leq k \leq 99. \end{aligned} \quad (17)$$

A full-window reported path is compatible exactly when it lies on the integer grid and contains at least one 100. A suppressed all-zero joint-query series need not satisfy this separately normalized, single-series model.

Prefix scale. Here a prefix runs from the first week in the series through the forecast origin; it is not the rolling estimation window in Appendix A. For a prefix of a compatible path, put $m_t = \max_{s \leq t} q_s$, $z_t = \max_{s \leq t} y_s$ and $\mathcal{F}_t = \{s > t : y_s = 100\}$. The exact feasible set is

$$\mathcal{M}_t = \begin{cases} \{1\}, & \mathcal{F}_t = \emptyset, \\ I_{z_t}, & \mathcal{F}_t \neq \emptyset. \end{cases} \quad (18)$$

Monotonicity of rounding places m_t in I_{z_t} . If no future value is reported as 100, a global maximum must lie in the prefix. Otherwise, assign a future reported 100 latent value 1 and any $m \in I_{z_t}$ to a prefix observation reported as z_t . The other prefix cells admit values no larger than m . Completing the suffix within its cells proves attainability.

When a future reported 100 exists, the reciprocal multiplier $\lambda_t = 1/m_t$ has the sharp set

$$\Lambda_t = \begin{cases} (200, \infty), & z_t = 0, \\ \left[\frac{100}{z_t + 0.5}, \frac{100}{z_t - 0.5} \right], & 1 \leq z_t \leq 99, \\ [1, 200/199], & z_t = 100. \end{cases} \quad (19)$$

With no future reported 100, $\lambda_t = 1$. If the prefix already contains a reported 100, each unrounded prefix-normalized coordinate $V_{s|t} = 100q_s/m_t$ satisfies

$$0 \leq V_{s|t} - 100q_s \leq 0.5. \quad (20)$$

Since $q_s \leq m_t$ and $m_t \geq 0.995$, the difference is at most $100(1 - m_t)$. Half-up rounding can therefore preserve y_s or increase it by one grid step, capped at 100. If $\mathcal{F}_t$ is empty, the rounded prefix is identified exactly.

Individual coordinates and endpoints. Sharp coordinate bounds must retain the common prefix denominator. When $\mathcal{F}_t \neq \emptyset$ and $t \geq 2$, let a, A be the numerical endpoints of I_{y_s} , and b, B the largest lower and upper endpoints among the other prefix cells. Then

$$\begin{aligned} \inf V_{s|t} &= \frac{100a}{\max(a, B)}, \\ \sup V_{s|t} &= \frac{100A}{\max(A, b)}. \end{aligned} \quad (21)$$

If $a \geq B$, the set is $\{100\}$. Otherwise its lower endpoint is included exactly when $a > 0$ and $B = 1$; its upper endpoint is included exactly when $A > b$. To see this, let m_{-s} be the maximum of the other normalized latent prefix values. The ratio $q_s / \max(q_s, m_{-s})$ is continuous on the compatible positive cells, nondecreasing in the numerator and nonincreasing in the other maximum. Its image therefore contains all interior points between the stated bounds; endpoint inclusion follows from the open and closed cell boundaries. With no future reported 100, the set is $100I_{y_s}$, except that the sole reported 100 is fixed at 100. For $t = 1$, the prefix-normalized coordinate is always 100.

Intersecting these feasible sets with $100I_k$ gives the rounded coordinate sets, with open endpoints preserved. These are marginal sets: their Cartesian product generally violates the shared-denominator constraint. They are deterministic compatibility bounds, not confidence intervals or recovered Google publication vintages. The supporting materials retain the full endpoint arguments and exact arithmetic checks.

References

- [1] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1-3, 1950. doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.
- [2] Sylvain Calinon. A tutorial on task-parameterized movement learning and retrieval. *Intelligent Service Robotics*, 9(1):1-29, 2016. doi: 10.1007/s11370-015-0187-9.
- [3] Hyunyoung Choi and Hal Varian. Predicting the present with Google Trends. *Economic Record*, 88(s1):2-9, 2012. doi: 10.1111/j.1475-4932.2012.00809.x.
- [4] Claudia Czado, Tilmann Gneiting, and Leonhard Held. Predictive model assessment for count data. *Biometrics*, 65(4):1254-1261, 2009. doi: 10.1111/j.1541-0420.2009.01191.x.
- [5] Candice Djorno, Mauricio Santillana, and Shihao Yang. Restoring the forecasting power of Google Trends with statistical preprocessing. arXiv:2504.07032, 2025. URL <https://arxiv.org/abs/2504.07032>.
- [6] Vera Z. Eichenauer, Ronald Indergand, Isabel Z. Martínez, and Christoph Sax. Obtaining consistent time series from Google Trends. *Economic Inquiry*, 60(2):694-705, 2022. doi: 10.1111/ecin.13049.
- [7] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189-1232, 2001. doi: 10.1214/aos/1013203451.
- [8] Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359-378, 2007. doi: 10.1198/016214506000001437.
- [9] Tilmann Gneiting and Roopesh Ranjan. Combining predictive distributions. *Electronic Journal of Statistics*, 7:1747-1782, 2013. doi: 10.1214/13-EJS823.
- [10] Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E. Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B*, 69(2):243-268, 2007. doi: 10.1111/j.1467-9868.2007.00587.x.
- [11] Google. FAQ about Google Trends data. <https://support.google.com/trends/answer/4365533>, n.d. Accessed: October 23, 2026.
- [12] Google. Search tips for Trends. <https://support.google.com/trends/answer/4359582>, n.d. Accessed: October 23, 2026.
- [13] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York, NY, 2nd edition, 2009. doi: 10.1007/978-0-387-84858-7.

- [14] Hansika Hewamalage, Klaus Ackermann, and Christoph Bergmeir. Forecast evaluation for data scientists: common pitfalls and best practices. *Data Mining and Knowledge Discovery*, 37(2):788-832, 2023. doi: 10.1007/s10618-022-00894-5.
- [15] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2): 65-70, 1979.
- [16] Rob J Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, Melbourne, 3rd edition, 2021. URL <https://otexts.com/fpp3/>.
- [17] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1): 35-45, 1960. doi: 10.1115/1.3662552.
- [18] Hans R. Künsch. The jackknife and the bootstrap for general stationary observations. *Annals of Statistics*, 17(3):1217-1241, 1989. doi: 10.1214/aos/1176347265.
- [19] Sebastian Lerch, Thordis L. Thorarinsdottir, Francesco Ravazzolo, and Tilmann Gneiting. Forecaster’s dilemma: Extreme events and forecast evaluation. *Statistical Science*, 32(1):106-127, 2017. doi: 10.1214/16-STS588.
- [20] Marcelo C. Medeiros and Henrique F. Pires. The proper use of Google Trends in forecasting models. arXiv:2104.03065, 2021. URL <https://arxiv.org/abs/2104.03065>.
- [21] Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983-999, 2006. URL <https://jmlr.org/papers/v7/meinshausen06a.html>.
- [22] Raydonal Ospina and Silvia L. P. Ferrari. A general class of zero-or-one inflated beta regression models. *Computational Statistics & Data Analysis*, 56(6):1609-1623, 2012. doi: 10.1016/j.csda.2011.10.005.
- [23] Yair Shimshoni, Niv Efron, and Yossi Matias. On the predictability of search trends. Technical report, Google, 2009. URL <https://research.google/pubs/on-the-predictability-of-search-trends/>.
- [24] Ruud H. Teunter, Aris A. Syntetos, and M. Zied Babai. Intermittent demand: Linking forecasting to inventory obsolescence. *European Journal of Operational Research*, 214(3):606-615, 2011. doi: 10.1016/j.ejor.2011.05.018.
- [25] Ali Caner Türkmen, Tim Januschowski, Yuyang Wang, and Ali Taylan Cemgil. Forecasting intermittent and sparse time series: A unified probabilistic framework via deep renewal processes. *PLOS ONE*, 16(11):e0259764, 2021. doi: 10.1371/journal.pone.0259764.
- [26] Jamelle Watson-Daniels, David C. Parkes, and Berk Ustun. Predictive multiplicity in probabilistic classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9):10306-10314, 2023. doi: 10.1609/aaai.v37i9.26227.
- [27] Robert West. Calibration of Google Trends time series. In *Proc. 29th ACM Int. Conf. Information and Knowledge Management (CIKM)*, pages 2257-2260, 2020. doi: 10.1145/3340531.3412075.