

Baseline-relative evaluation of lap-time and tyre-degradation models for Formula 1 race strategy

T. Shakila

Bharath Institute of Science and Technology

K. Baalaji

baalaji.kadarkarai@hotmail.com

Bharath Institute of Science and Technology

Research Article

Keywords: Formula 1, lap-time prediction, tyre degradation, baseline-relative evaluation, persistence forecast, gradient boosting

Posted Date: October 23rd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11179114/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

Abstract

Race-strategy decision support in Formula 1 rests on forecasts of next-lap time and tyre degradation. Published models report small errors but rarely report what a trivial forecast achieves on the same laps, so their accuracy cannot be separated from the strong autocorrelation of consecutive laps. This study supplies that comparison using 167,083 laps from the public FastF1 interface, and finds that constructing the baseline fairly matters as much as reporting it. Gradient-boosted models were trained on 2019–2022, frozen, and evaluated with the race as the unit of inference. Against a persistence forecast the advantage is 15.1%, but 3.2% of comparisons span a pit stop, where persistence compares laps on different tyres: the gain reaches 30.6% in a stint's first five laps against 10.2% later. Against a stint-aware baseline carrying a compound offset across stops it is 12.1% (race-level $p = 0.0065$), and 13.2% on all green-flag laps without outcome-based filtering. A nested ablation shows the advantage comes from predicting a residual from the previous lap: all three residual configurations beat the baseline by race-level test, and no absolute one does. For degradation, a stint-referenced target subtracts a within-stint constant, so its persistence forecast is identical to the lap-time one and any lap-time model's degradation error equals its lap-time error; the target cannot distinguish degradation knowledge from lap-time knowledge at a one-step horizon. With the evaluation season fixed and five seeds averaged, pre-regulation and same-era training perform alike (11.9%, 12.2%) and pooling both is best (16.1%).

1 Introduction

Formula 1 race strategy is a sequential decision problem under hard time constraints. A strategist must decide when to pit and which tyre compound to fit, and those decisions are priced by two forecasts: how fast the car will run the next lap, and how much pace it is losing as the tyre ages [1, 2]. Both are routinely modelled with machine learning on publicly accessible timing data, and decision-support layers are built on top of them [3, 4]. If the forecasts are weak, the strategy advice derived from them is weak, however sophisticated the optimiser above them.

The accuracy of those forecasts is therefore the load-bearing quantity in this literature, and it is reported in a way that cannot be audited. Studies of Formula 1 lap-time prediction report mean absolute errors, coefficients of determination or classification scores against other learned models [1, 5, 6], but with rare exceptions they do not report what a trivial forecast achieves on the same observations. Consecutive green-flag laps by the same driver differ by a few tenths of a second, so a forecast that repeats the previous lap is already close. Without that number, a reported error is uninterpretable.

Outside motorsport this is settled practice. The mean absolute scaled error of Hyndman and Koehler [7] divides model error by the error of a naive forecast precisely so that accuracy is reported relative to a floor, and the M4 competition [8] scored 61 methods against a naive benchmark. The practice keeps earning its place: Zeng et al. [9] found a one-layer linear model competitive with or better than transformer forecasters on long-horizon benchmarks, and Makridakis et al. [10] found statistical

methods outperforming machine-learning ones on the M3 series. Hewamalage et al. [11] catalogued the evaluation practices that produce spurious conclusions, and Cerqueira et al. [12] showed how partitioning choices in correlated series inflate apparent accuracy. That norm has not propagated into motorsport analytics.

This study applies it, and finds that applying it carefully is harder than it first appears. Reporting a naive floor is necessary but not sufficient: the floor must also be *constructed fairly for the task*, and the target must be *capable of testing the claim made about it*. Both conditions fail in the obvious implementations, and correcting them changes the answers. The contributions are:

1. **1. A baseline-relative evaluation of Formula 1 lap-time forecasting** on public data, with every error accompanied by a naive forecast on identical held-out laps and the race as the unit of inference (Sections 5.2, 6.1).
2. **2. A demonstration that a naive baseline can be unfairly handicapped, and a fair replacement.** The persistence forecast spans pit stops, comparing laps on different tyres; the model's advantage is three times larger where that happens. A stint-aware baseline that carries a compound offset across stops reduces the measured advantage from 15.1% to 12.1%, which we take as the defensible figure (Section 6.2).
3. **3. A demonstration that the target formulation, not the feature set, decides whether the baseline is beaten.** All three residual configurations beat persistence by race-level test; no absolute one does, the worst by a factor of seven (Section 6.3).
4. **4. A structural result about the degradation target.** A stint-referenced target — lap time minus a within-stint reference pace — subtracts a within-stint constant, so its persistence forecast is identical to the lap-time one and any lap-time model's degradation error equals its lap-time error. At a one-step horizon the target cannot test degradation knowledge at all (Section 6.4).
5. **5. A released pipeline with verified causal features.** Two tests — future-invariance under race truncation and invariance to perturbation of the predicted lap's own outcome — are released and pass for all 28 features. No manuscript number is transcribed by hand; a released script checks the text against the archived results (Sections 4.3, 5.4).

Section 2 reviews related work, Section 3 states the problem, Section 4 the method, Section 5 the setup. Section 6 reports results, Sections 7 and 8 discuss and bound them, and Section 9 concludes.

2 Related work

2.1 Lap-time prediction in Formula 1

Patro et al. [5] applied machine-learning regressors to Formula 1 lap times. Kara and Küçükmanisa [1] predicted lap times with LSTM networks. Mistry and Smith [6] trained recurrent networks to simulate strategic decision-making across a race. Zhao [13] forecast fastest qualifying lap times with a deep neural network. None of these reports the error of a naive forecast on its own evaluation laps, which is

the comparison this study supplies. Outside Formula 1, Tulabandhula and Rudin [14] built a real-time prediction and tyre-change decision system for professional car racing and stressed that racing structure — tyre changes, clean air, caution periods — had to be built into the statistical models.

2.2 Tyre degradation modelling

Cappello and Hoegh [2] model tyre degradation with a state-space formulation, the closest published work to the degradation target used here and — unusually for this subfield — one that compares against a structurally matched time-series alternative rather than only against other learned models. Heilmeier et al. [15] built a lap-discretised race simulation incorporating degradation, fuel-mass loss, pit stops and overtaking; its deterministic precursor [16] fitted the degradation parameters from publicly accessible lap times. With team telemetry, Todd et al. [17] forecast tyre energy, a driver of degradation, with deep learning and gradient boosting and explained the forecasts. In our own earlier survey of big-data analytics for tyre and brake performance in Formula 1 [18] we catalogued the predictive models applied to tyre-change and brake-cooling decisions; we did not encounter a study reporting a comparison against a naive forecast, and we offer that as our prior observation rather than as the result of a systematic search.

2.3 Downstream decision support

Sasikumar et al. [3] built pit-stop decision support from telemetry with deep learning models. Thomas et al. [4] introduced Race Strategy Reinforcement Learning for pit-stop strategy, and in related work [19] added explanations to that policy for Formula One race strategy. Heilmeier et al. [20] automated race-strategy decisions with neural networks trained on historical timing data. Avella et al. [21] present a simulation framework for Formula 1 race strategy built on pit-stop optimisation. Outside Formula 1, Liu and Fotouhi [22] combined neural lap-time surrogates with Monte Carlo tree search for Formula-E energy strategy, and Aguad and Thraves [23] optimised pit-stop strategy with dynamic programming and game theory, extending the deterministic and stochastic dynamic programs of Carrasco Heine and Thraves [24].

Each of these consumes lap-time or degradation forecasts as an input, which is why the credibility of those forecasts bounds the credibility of the decision support built on them. Table 1 positions the present study against the closest of these works.

Table 1

Positioning against the closest published works. “Naive baseline reported” requires a trivial forecast evaluated on the same observations as the model

Reference	Technique	Setting	Naive baseline reported	Difference here
Kara and Küçükmanisa [1]	LSTM	F1 lap times	No	Persistence scored on identical laps
Patro et al. [5]	Machine-learning regressors	F1 lap times	No	MAE against a naive floor
Mistry and Smith [6]	Recurrent networks	F1 strategic simulation	No	Forecast skill, not simulation fidelity
Zhao [13]	Deep neural network	F1 qualifying lap times	No	Race laps, not qualifying; naive floor reported
Cappello and Hoegh [2]	State-space model	F1 tyre degradation	Yes — a matched time-series alternative	Baseline fairness analysed; nested ablation
Sasikumar et al. [3]	Deep learning	F1 pit-stop decision support	No	Tests the upstream forecasts it would rely on
Thomas et al. [4, 19]	Reinforcement learning	F1 pit-stop strategy	Not applicable	Supervised targets with baselines
Heilmeier et al. [15, 20]	Simulation; neural strategy engineer	Circuit motorsport	Not applicable	Strategy layer, not forecast evaluation

Gap. No published study we are aware of evaluates Formula 1 lap-time forecasts against a naive baseline on identical held-out laps, examines whether that baseline is fairly constructed, or establishes whether reported accuracy depends on the target formulation. Cappello and Hoegh [2] are the nearest, and address degradation alone. The contribution is the application of an established evaluation standard [7, 8, 11] to a field that has not adopted it, and the findings that follow.

3 Problem statement and research questions

Let a driver's green-flag lap at index t within a race have duration $y(t)$. A lap-time model predicts $y(t)$ from information available before lap t . A degradation model predicts $d(t)$, pace lost relative to fresh-tyre pace within the current stint, defined in Section 4.2. The persistence forecast is $\hat{y}(t) = y(t - 1)$, and $\hat{d}(t) = d(t - 1)$.

RQ1 Does a gradient-boosted lap-time model outperform a naive forecast on identical held-out laps, under inference that respects race-level dependence?

RQ2 Is that naive forecast fairly constructed, and does the answer to RQ1 survive a fairer one?

RQ3 Which components — feature blocks, target formulation — are responsible for any advantage?

RQ4 Can a stint-referenced degradation target test degradation knowledge, and does the lap-time advantage depend on the technical-regulation era of the training data?

4 Proposed methodology

The pipeline comprises acquisition and model fitting, followed by four evaluations that each compare the fitted model against a naive forecast on identical laps. Figure 1 shows the arrangement.

4.1 Data and lap filtering

Lap records for 2019–2025 were retrieved with FastF1 version 3.8.2 [25], an unofficial open-source interface to the public Formula 1 live-timing service whose underlying data remain subject to the terms of use of their rights holder; the retrieved records are provided in Online Resource 1 and deposited in Zenodo [26]. A lap enters the primary analysis if it is flagged timing-accurate, carries no pit-in or pit-out timestamp, has green track status, and is no slower than 110% of an expanding median over the driver's preceding laps in that race. The expanding reference removes look-ahead, but it does not remove outcome selection: a lap is excluded when its *own* time exceeds the threshold, so the retained set is chosen partly by the quantity being predicted. Because that could flatter both forecasts, Section 6.2 reports the full comparison on **all green-flag laps with no such exclusion**, and we treat that as the robustness check on every headline number. The expanding median requires three prior laps before it applies; earlier laps are retained unconditionally.

4.2 Degradation target, and a structural property of it

Degradation is not defined consistently across this literature. We use a stint-referenced target: for a stint S ,

$$d(t) = y(t) - r(S)$$

1

$$r(S) = \text{median}\{y(u) : u \in S, 2 \leq \tau(u) \leq 4\}$$

2

where $\tau(u)$ is tyre age in laps. Referencing within the stint removes circuit, car and fuel-load-at-stint-start effects. Stints with fewer than five green-flag laps are excluded. We have not found prior work using

exactly this construction — Cappello and Hoegh [2] take a state-space approach to the same underlying quantity — so we present it as our own definition rather than a community standard. Section 6.4 shows that the definition, and not the model fitted to it, is what limits the conclusions available from it.

The definition has two consequences that govern what a degradation comparison can show. Because $r(S)$ is constant within a stint,

$$d(t) - d(t-1) = y(t) - y(t-1)$$

3

so degradation persistence makes the same error, lap for lap, as lap-time persistence. And if a degradation forecast is formed from a lap-time forecast as $\hat{d}(t) = \hat{y}(t) - r(S)$, then

$$d(t) - \hat{d}(t) = y(t) - \hat{y}(t)$$

4

so its degradation error *is* its lap-time error. Section 6.4 confirms both identities empirically. Together they mean that at a one-step horizon this target cannot distinguish a model that has learned about tyres from one that has learned about lap times: any lap-time model, including one with no tyre features, inherits the same comparison. This is a limitation of the target, and it applies to any degradation definition formed by subtracting a within-group constant.

4.3 Feature blocks and causal availability

Twenty-eight features fall in four blocks: **condition** (12: tyre age, compound, stint index, lap number, race progress, fuel proxy, air and track temperature, humidity, rainfall, wind speed, previous-lap track position, fresh-tyre indicator); **recent pace** (6: previous lap, 3- and 5-lap moving averages, 5-lap exponentially weighted average, 3-lap rolling standard deviation, deviation of the previous lap from the 5-lap trend); **traffic and track evolution** (6: gap to the car ahead and behind, an in-traffic indicator at 2.0 s, the field's median lap time, pace relative to it, the five-lap change in field median); and **identity** (4: driver, constructor, circuit, compound as categorical).

Three constructions required care. FastF1 reports track position at the *end* of a lap, which is a function of that lap's duration, so only the previous lap's position is admissible. Race progress and fuel proxy cannot divide by the maximum lap number observed in a race, which is known only once the race has finished; both use a per-circuit expected distance frozen from the training seasons. The opening-lap fallback for the persistence base likewise uses a frozen per-circuit prior rather than the race's own median. For circuits absent from training — Las Vegas in 2023 — the global prior across training circuits is used, and the same applies to any new venue. FastF1 also reports three venues under different names across seasons (Monte Carlo and Monaco, Singapore and Marina Bay, Miami and Miami Gardens), which the circuit feature and priors treat as distinct. Harmonising the names changes the five-seed hold-out MAE by +0.0025 s (race-level $p = 0.062$), so the released models keep the reported names.

Two released tests establish these properties rather than asserting them. `test_causality.py` truncates a race after lap k , rebuilds the features and asserts that the row for every lap $\leq k$ is unchanged, which detects dependence on *later* laps. Because that cannot detect a feature computed from the predicted lap's *own* outcome — end-of-lap position is exactly such a feature — a second test perturbs $y(k)$ and asserts that row k 's own features do not move. All 28 features pass both.

4.4 Naive baselines

Two naive forecasts are reported throughout.

Persistence is the previous *retained* lap. After filtering this is the immediately preceding lap 94.5% of the time and lies in the same stint 96.8% of the time; the remainder span a pit stop, where the comparison is between laps on different tyres.

A stint-aware baseline repairs that. Where the previous retained lap is in the same stint it uses that lap; otherwise it uses the previous lap plus a compound-transition offset, the median lap-time change across a stint boundary for the newly fitted compound, estimated on training data only. A strategist knows which compound has just been fitted, so this uses nothing unavailable at prediction time. Section 6.2 reports both.

4.5 Residual target formulation

The model predicts a change from the persistence base rather than an absolute value; the reported prediction is base plus learned residual. Gradient-boosted trees approximate a continuous identity mapping by piecewise-constant splits, so a model asked to output absolute lap time — spanning roughly 60–120 s across circuits — expends capacity rediscovering “approximately the same as last lap”. Section 6.3 measures the consequence across all six cells of the ablation.

4.6 Algorithm

Algorithm 1

Baseline-relative training and evaluation

Require: training seasons T_{train} , evaluation seasons T_{test} , feature block B , target form $m \in \{\text{absolute}, \text{residual}\}$, seed s , baseline b

Ensure: MAE of the model and of the baseline on identical laps

- 1: $\text{priors} \leftarrow$ circuit priors fitted on T_{train} only
- 2: $L_{\text{train}} \leftarrow \text{filter}(\text{load}(T_{\text{train}}))$; $L_{\text{test}} \leftarrow \text{filter}(\text{load}(T_{\text{test}}))$
- 3: $X_{\text{train}}, y_{\text{train}} \leftarrow \text{features}(L_{\text{train}}, B, \text{priors})$
- 4: $X_{\text{test}}, y_{\text{test}} \leftarrow \text{features}(L_{\text{test}}, B, \text{priors})$

- 5: freeze categorical level sets C from X_{train}
- 6: split T_{train} into inner-fit and inner-validation *by race* using seed s
- 7: **if** $m = \text{residual}$ **then** $lab \leftarrow y_{\text{train}} - base_{\text{train}}$ **else** $lab \leftarrow y_{\text{train}}$
- 8: $f \leftarrow \text{fit}(X_{\text{train}}[\text{inner-fit}], lab[\text{inner-fit}])$, early stopping on inner-validation
- 9: $pred \leftarrow base_{\text{test}} + f(\text{align}(X_{\text{test}}, C))$ if $m = \text{residual}$, else $f(\text{align}(X_{\text{test}}, C))$
- 10: $M \leftarrow \{\text{laps of } L_{\text{test}} \text{ where the baseline } b \text{ is defined}\}$
- 11: **return** $\text{MAE}(pred[M], y_{\text{test}}[M]), \text{MAE}(b[M], y_{\text{test}}[M])$

Step 10 of Algorithm 1 is the methodological core, and Section 6.2 shows that the choice of b in step 10 is as consequential as the model in step 8.

Complexity. Training is $O(n \cdot d \cdot K \cdot \log n)$ for n laps, d features and K boosting rounds, dominated by histogram construction in LightGBM [27]. Both baselines are $O(1)$ per lap.

4.7 Models and hyperparameters

Table 2 lists every hyperparameter of both models together with the reason for its value. Two properties of that table matter more than the values themselves: no hyperparameter search was performed, and the settings are identical in every configuration of the ablation, so Section 6.3 compares formulations rather than tuning effort.

Table 2

Hyperparameters. No hyperparameter search was performed; values are conventional defaults for this model class, fixed before any evaluation and identical across every configuration, so the ablation compares formulations rather than tuning effort

Component	Parameter	Value	Justification
LightGBM	objective	L1	Matches the reported MAE metric
LightGBM	learning rate	0.05	Low rate with early stopping
LightGBM	num leaves	96	Capacity for circuit and driver interactions
LightGBM	min data in leaf	40	Guards against leaves fitting single stints
LightGBM	feature, bagging fraction	0.85	Decorrelates trees; provides seed-to-seed variation
LightGBM	boosting rounds	≤ 3000 , early stop 100	Stopping set on inner validation, never the hold-out
GRU	hidden units, window	64, 5 laps	Parameter economy; shortest window spanning warm-up into wear
GRU	optimiser, learning rate	Adam, 1×10^{-3}	Standard default for recurrent networks
GRU	loss, batch size, epochs	Smooth L1, 512, ≤ 40 (patience 6)	Robust to outlier laps; early stopping on inner validation
GRU	input scaling	Standardised with training statistics	No hold-out information enters the scaler
Protocol	seeds	5 (ablation)	Enables mean $\pm$ standard deviation
Protocol	split granularity	race	Laps within a race are correlated

5 Experimental setup

5.1 Corpus

Table 3 sets out the corpus by season, with the role each season plays.

Table 3
Corpus after filtering. Seasons 2019–2022 train, 2023 is the hold-out, and 2024–2025 are reserved as an out-of-time check reported separately by season

Season	Races	Drivers	Green-flag laps retained	Role
2019	21	20	19,600	Training
2020	17	23	14,567	Training
2021	22	21	19,880	Training
2022	22	22	18,240	Training
2023	22	22	19,886	Hold-out
2024	24	24	22,605	Out-of-time
2025	24	21	21,864	Out-of-time
Total	152	—	136,642	—

A further 30,441 laps were removed by the filters of Section 4.1, from 167,083 retrieved. Training therefore uses 72,287 laps across 2019–2022. Because 2023 is the hold-out, 2024 and 2025 are also out-of-sample for the frozen model, and Section 6.6 reports them separately rather than pooled: they differ in regulation stability and in the number of unseen circuits.

5.2 Protocol and inference

Splits are by season. Within training, the early-stopping set is 15% of *races*, never of laps: laps within a race share weather, track evolution and safety-car interruptions, so a lap-level split leaks across the boundary [12].

Race-level paired *t*-tests are the primary inference throughout, because laps within a race are not independent and the 19,463 lap-level comparisons badly overstate the effective sample. Each race contributes one mean absolute error for the model and one for the baseline, giving $n = 22$ paired observations on the hold-out. Lap-level tests, a race-clustered bootstrap (10,000 resamples of races with replacement) and Diebold–Mariano tests with Newey–West HAC variance and the Harvey–Leybourne–Newbold small-sample correction [28, 29] are reported as secondary evidence. Holm–Bonferroni correction [30] is applied across the four primary race-level tests.

5.3 Metrics

Mean absolute error in seconds is primary, matching the L1 training objective. Improvement is reported as a percentage reduction relative to the stated baseline on identical laps. Effect size is Cohen's d_z for the paired race-level differences. Root-mean-square error is reported alongside for the lap-time model.

5.4 Reproducibility

Retrieval, feature construction, training, evaluation, statistics and figure generation are separate scripts, released with the frozen model, the circuit priors and the result files. Every figure is drawn from the archived JSON. A released script, `12_check_manuscript.py`, re-reads those files and asserts every load-bearing value in this manuscript, including the annotations printed inside figures; it exits non-zero on any mismatch, and was run against the submitted text.

6 Results

6.1 The lap-time model beats a naive forecast (RQ1)

On the 2023 hold-out the frozen model attains an MAE of 0.3900 s on the 19,463 laps where a persistence forecast is also defined. Persistence attains 0.4591 s on the same laps: a 15.1% reduction. On all 19,886 retained laps, including the 423 opening laps where the model must predict without a previous lap and errs by 5.459 s, the MAE is 0.4978 s and the RMSE 1.8027 s.

By race-level paired test the advantage is significant ($t(21) = -4.52$, $p = 1.9 \times 10^{-4}$, $d_z = -0.96$), with the model ahead in 21 of 22 races; a sign test over those races gives $p = 1.1 \times 10^{-5}$ and a Wilcoxon signed-rank test $p = 2.6 \times 10^{-4}$. The race-clustered bootstrap puts the mean loss difference in $[-0.0866, -0.0439]$ s and the improvement in $[7.8\%, 20.9\%]$, both excluding zero. The lap-level test is far smaller in p (7.5×10^{-103}) and illustrates why it should not be the headline: it treats 19,463 correlated laps as independent.

Per lap, the model is closer than persistence on only 56.1% of laps, six points above a coin flip, so most of the aggregate gain comes not from being slightly better everywhere but from being much better on a minority of laps. Section 6.2 identifies which laps those are, and shows that this matters for how the comparison should be constructed.

Tyre-degradation results appear in Fig. 2 and are interpreted in Section 6.4, where the structure of the target changes what they can support.

6.2 The naive baseline is not fairly constructed, and the gain is smaller when it is (RQ2)

Persistence uses the previous *retained* lap. That is the immediately preceding lap for 94.5% of comparisons, and lies in the same stint for 96.8%. The remaining 618 laps span a pit stop, where persistence is asked to predict a lap on fresh tyres from a lap on worn ones — a comparison no strategist would make, since the compound just fitted is known.

Stratifying shows how much of the advantage lives there (Table 4). Where the previous lap is in the same stint the model gains 11.1%; where it crosses a stint boundary it gains 44.6%, cutting 1.7213 s to 0.9542 s. The gain is 30.6% in the first five laps of a stint and 10.2% from lap 11 onward. The model's advantage is concentrated where the baseline is structurally handicapped.

Table 4

Lap-time results by stint stratum, 2023 hold-out. Each stratum compares the frozen model against the persistence forecast on identical laps, tested at race level

Stratum	Laps	Model MAE (s)	Persistence MAE (s)	Improvement	Race-level p	d_z
Previous lap in same stint	18,845	0.3715	0.4177	11.1%	0.027	-0.51
Previous lap crosses a stint boundary	618	0.9542	1.7213	44.6%	2.4×10^{-6}	-1.37
Stint laps 1–5	2,670	0.5726	0.8254	30.6%	3.9×10^{-10}	-2.34
Stint laps 6–10	4,576	0.4453	0.4924	9.6%	0.024	-0.52
Stint laps 11 and later	12,217	0.3294	0.3666	10.2%	0.087	-0.38

The last row deserves emphasis: in settled running, late in a stint, the model's 10.2% advantage does **not** reach significance at race level ($p = 0.087$), even though it is the largest stratum. The model's demonstrable skill is in transitions, not in steady state.

A fair baseline should therefore know what a strategist knows. The stint-aware baseline of Section 4.4 carries a compound-transition offset — estimated on training data alone, ranging from -1.02 s for hard to -2.37 s for intermediate — across stint boundaries. Against it the model's advantage is **12.1%** (0.3900 s against 0.4438 s; $t(21) = -3.02$, $p = 0.0065$, $d_z = -0.64$, 95% CI on the difference $[-0.086, -0.016]$ s), against 15.1% for the naive version. We report 12.1% as the defensible headline and the naive 15.1% alongside it, because the naive figure is the one comparable to the literature.

The outcome-based filter of Section 4.1 is the other way the comparison could flatter itself. Repeating the test on **all green-flag laps**, with no 110% exclusion, gives model 0.4945 s against baseline 0.5695 s, a 13.2% advantage ($t(21) = -3.58$, $p = 0.0018$). Both errors rise, the advantage does not disappear, and it sits between the two filtered figures.

The general point outlives this dataset. Reporting a naive baseline is necessary, but a naive baseline can be made to lose by being denied information the practitioner has. Where the residual target is defined by differencing against that same baseline — as it is here and in any residual formulation — a handicapped baseline inflates the measured advantage twice over.

6.3 The target formulation, not the feature set, decides the outcome (RQ3)

The nested ablation crosses three feature blocks with two target formulations, five seeds each, on the identical 19,463 hold-out laps (Table 5, Fig. 3).

Table 5

Ablation on the 2023 hold-out. Learned entries are mean $\pm$ standard deviation over five seeds. Race-level p tests the seed-0 model against persistence on 22 paired race means; “beats baseline” requires $p < 0.05$ with the model ahead. R^2 is shown to demonstrate that it does not discriminate between these configurations, and is discussed in Section 7.2

Configuration	Kind	MAE (s)	R^2	Race-level p	Beats baseline
Persistence (previous lap)	Non-learned	0.4591	0.9870	—	—
Rolling mean, 3 laps	Non-learned	0.5301	0.9824	—	No
Rolling mean, 5 laps	Non-learned	0.5949	0.9796	—	No
Field median, previous lap	Non-learned	1.1211	0.8562	—	No
LightGBM, condition, absolute	Learned	3.1935 ± 0.2561	0.7613	5.0×10^{-4} (worse)	No
LightGBM, condition, residual	Learned	0.4493 ± 0.0090	0.9875	9.4×10^{-4}	Yes
LightGBM, +pace, absolute	Learned	0.5693 ± 0.0783	0.9838	0.220	No
LightGBM, +pace, residual	Learned	0.4036 ± 0.0027	0.9885	1.2×10^{-10}	Yes
LightGBM, +traffic, absolute	Learned	0.5098 ± 0.0391	0.9927	0.214	No
LightGBM, +traffic, residual	Learned	0.3849 ± 0.0040	0.9895	1.6×10^{-7}	Yes
Ridge regression, full, residual	Learned	0.7064	—	—	No
Random forest, full, residual	Learned	0.3987 ± 0.0004	—	—	Yes

The R^2 column is worth reading across. The configuration with the *highest* R^2 in the table, 0.9927, is an absolute-target model that loses to persistence; the best model by MAE scores 0.9895; and persistence itself scores 0.9870. R^2 orders these configurations almost in reverse of their usefulness.

The separation is by target formulation, not by features. All three residual configurations beat persistence at race level; no absolute configuration does, and the weakest is worse than persistence by a factor of seven. Within the residual column, features still matter — 0.4493 to 0.4036 to 0.3849 s — but the condition-only residual model already beats the baseline with no recent-pace features at all, while the

full-feature absolute model does not. A study reporting only the full-feature residual cell would attribute to its features an advantage that the target formulation supplies.

A ridge regression on the same residual target reaches 0.7064 s, worse than persistence, confirming that the residual formulation alone is not sufficient: the mapping from conditions to lap-to-lap change is not linear. A random forest reaches 0.3987 s, close to LightGBM, so the result is a property of the formulation and the data rather than of one library. Figure 4 ranks all twelve configurations of Table 5 on one axis, where the separation between the residual and absolute groups is visible directly.

6.4 A stint-referenced degradation target cannot test degradation knowledge (RQ4a)

Section 4.2 predicted two identities from the definition of the stint-referenced target. Both hold exactly in the data. Degradation persistence makes an error identical to lap-time persistence on **100.0%** of the 14,681 covered laps (mean absolute difference 0.0 s). And the frozen lap-time model, re-expressed as a degradation forecast by subtracting the stint reference, attains 0.3276 s — which is its lap-time error on those same laps, not an independent result.

This has a consequence for any study that uses such a target, which we state plainly. At a one-step horizon, this target is lap-time forecasting minus a within-stint constant. Any lap-time model inherits a degradation comparison from it, whether or not the model knows anything about tyres. A reported success on this target is therefore not evidence that degradation is forecastable beyond persistence; it is the lap-time result in different units.

The scale of the target shows why. Its standard deviation is 2.398 s, but the standard deviation of its one-step change is only 0.614 s: one-step differencing removes 93% of the variance (74% of the standard deviation). Almost all of what a model on this target appears to explain is therefore level that persistence already carries forward from the previous lap.

Two results in this study show how easily a lap-level test on this target misleads.

The GRU attains 0.3604 s against degradation persistence at 0.3616 s — a 0.3% margin. That margin reaches $p = 0.008$ at lap level, which, read in isolation, would suggest a significant gain. At race level it does not: $t(21) = -1.19$, $p = 0.248$, with the model ahead in only 12 of 22 races and a sign test at $p = 0.832$. **We make no claim that the GRU beats persistence on degradation.**

The LightGBM – r(S) comparator looks stronger at 0.3276 s against 0.3616 s, a 9.4% margin at $p = 1.4 \times 10^{-32}$ at lap level. At race level it is ahead in 20 of 22 races (sign test $p = 1.2 \times 10^{-4}$, Wilcoxon $p = 3.0 \times 10^{-4}$), yet the paired t -test gives $p = 0.140$, because one race — Spa-Francorchamps, round 12, where it is worse by 0.347 s — dominates the variance of the 22 differences; without that race $p = 2.1 \times 10^{-10}$. None of this bears on degradation. By Eq. (4) its degradation error equals its lap-time error on every one of the

14,681 covered laps, so the 9.4% is the Section 6.1 lap-time result restricted to those laps and re-expressed in different units.

The degradation results therefore neither reverse nor confirm the lap-time conclusion: on this target the comparison is uninformative by construction.

Both learned degradation forecasts are far better than the structural baselines — linear-in-tyre-life per compound at 1.0985 s and predicting zero at 1.0744 s, on the same 14,681 laps — which shows only that within-stint pace is not a smooth function of tyre age, something persistence already exploits.

Testing degradation knowledge requires a target that cannot be reduced to a lagged lap time: a multi-lap horizon, where the within-stint constant no longer cancels between consecutive predictions, or a stint-level integral of pace loss. Section 8 sets this out as the primary methodological consequence of this study.

6.5 Training era matters less than a confounded design suggests (RQ4b)

Formula 1 changed technical regulations for 2022, introducing ground-effect aerodynamics, 18-inch wheels and a new Pirelli range. Whether a model trained before that change transfers across it is a practical question for a team deciding whether to retain historical data.

The design must hold the evaluation set fixed. A tempting design compares 2019–2021 → 2022 against 2022 → 2023, but that changes the evaluation season along with the training era; the apparent collapse it produces is a property of 2022 as an evaluation season, not of the regulation boundary. We report that confounded comparison in the last row of Table 6 only to show the size of the artefact.

Table 6

Cross-era transfer. The first three rows differ only in training data and are evaluated on the identical 19,463 laps of 2023. Improvement is against persistence; intervals are race-clustered bootstrap. The within-era control (2020 → 2021) is evaluated on a different season, so its level is not comparable with the first three rows, and its 12.1% is unrelated to the stint-aware figure of Section 6.2. The last column refits each configuration under five seeds and gives the mean improvement with its range across seeds; the other columns are single fits

Training	Evaluation	Train laps	Model MAE (s)	Improvement	95% CI	Race-level p	Improvement, five seeds (range)
2019–2021 (pre-regulation only)	2023	54,047	0.3965	13.6%	[7.1, 18.7]	1.4×10^{-4}	11.9% [10.5, 14.2]
2022 (new regulation only)	2023	18,240	0.3939	14.2%	[11.6, 16.9]	2.1×10^{-10}	12.2% [11.0, 14.3]
2019–2022 (full corpus)	2023	72,287	0.3900	15.1%	[7.8, 20.9]	1.9×10^{-4}	16.1% [14.8, 17.2]
2020 (within-era control)	2021	14,567	0.4580	12.1%	[9.0, 15.2]	1.1×10^{-7}	13.4% [11.2, 14.3]
2019–2021 (confounded design)	2022	54,047	0.4910	6.7%	[–8.5, 15.2]	0.562	5.4% [2.7, 9.2]

Single fits are not precise enough to rank these rows. The same configuration varies by up to two points of improvement with the random seed, as much as the gaps between rows, so every configuration was refitted under five seeds (inner split and LightGBM seeds varied, as in Section 6.3) and configurations were compared by paired race-level test on seed-averaged per-race errors (last column of Table 6). Three findings follow. First, training entirely before the regulation change costs nothing measurable: 11.9% against 12.2% for same-era training ($t(21) = -0.14$, $p = 0.89$). Second, pooling both is better than either alone: the full corpus reaches 16.1%, ahead of 2022-only training in 16 of 22 races ($t(21) = -2.41$, $p = 0.025$) and of 2019–2021-only training in 19 of 22 ($t(21) = -2.46$, $p = 0.023$), so three seasons of superseded regulations were useful rather than harmful. The frozen model used throughout (15.1%) lies near the lower end of the full-corpus seed range (14.8–17.2%), so the headline figures of Sections 6.1 and 6.2 are, if anything, conservative. Third, the within-era control reaches 13.4%, comparable to the cross-era rows, so nothing distinguishes crossing the boundary from not crossing it. The confounded row gives 5.4% over five seeds (6.7% in the single fit, with an interval spanning zero) – an artefact of evaluating on 2022.

The evidence therefore does not support the intuition that a model must be re-fitted promptly after a rules change rather than accumulate history; if anything it points the other way (Fig. 5).

The likely reason is Section 6.3. What the model learns is a mapping from recent pace and current conditions to a *change* from the previous lap. Regulations alter absolute lap times substantially; they alter lap-to-lap dynamics much less. A residual target is, by construction, partly insulated from the shift – which also predicts that an absolute-target model would transfer far worse, though we have not tested that.

6.6 Out-of-time seasons

The frozen model was applied to 2024 and 2025 without refitting. Table 7 reports them separately as well as pooled, because they are not interchangeable: 2024 shares 2023's regulations, while 2025 adds circuits and driver changes.

Table 7

Out-of-time application of the frozen 2019–2022 model, by season, against persistence on identical laps. Races favouring the model, and the sign test over them, are shown because the paired t-test is sensitive to a single discrepant race

Season	Comparable laps	Model MAE (s)	Persistence MAE (s)	Improvement	Races favouring model	Race-level t-test p	Sign test p
2024	22,147	0.4196	0.4895	14.3%	23 / 24	1.5×10^{-7}	3.0×10^{-6}
2025	21,411	0.3918	0.4345	9.8%	23 / 24	0.120	3.0×10^{-6}
Pooled	43,558	0.4059	0.4624	12.2%	46 / 48	7.1×10^{-5}	8.4×10^{-12}

Pooled across both seasons the race-level test gives $t(47) = -4.36$ ($p = 7.1 \times 10^{-5}$, $d_z = -0.63$).

The advantage persists two and three seasons past the training window, but it is not constant: 14.3% in 2024 against 9.8% in 2025. Reporting the seasons only pooled would give 12.2% and conceal that decline.

The 2025 row also shows why more than one test is needed. The model is ahead in 23 of 24 races, yet the paired t-test returns $p = 0.120$, because one race – round 13, where the model is worse by 0.429 s – dominates the variance of the 24 differences. The sign test ($p = 3.0 \times 10^{-6}$) and the Wilcoxon signed-rank test ($p = 9.1 \times 10^{-5}$) both reject decisively. We report the t-test as primary throughout for consistency, but read 2025 as a real advantage with one large adverse race, not as a null result.

6.7 Statistical tests

Table 8 collects the four primary comparisons and the secondary tests that accompany them.

Table 8

The four primary race-level comparisons with Holm–Bonferroni correction. The sign test and the Diebold–Mariano test are secondary and shown for completeness. Degradation rows compare the GRU degradation model against degradation persistence

Comparison	Race-level p	Holm-adjusted p	Significant	Sign test p	DM p
Lap time, 2023 hold-out	1.9×10^{-4}	5.6×10^{-4}	Yes	1.1×10^{-5}	1.4×10^{-91}
Lap time, 2024–2025 out-of-time	7.1×10^{-5}	2.8×10^{-4}	Yes	8.4×10^{-12}	1.0×10^{-86}
Degradation, 2023 hold-out	0.248	0.495	No	0.832	0.031
Degradation, 2024–2025 out-of-time	0.509	0.509	No	0.665	0.030

Holm correction is applied to the race-level p -values, which Section 5.2 designates as primary. The lap-time conclusions survive correction and are corroborated by the sign test. Neither degradation conclusion survives.

The degradation rows show the two secondary tests disagreeing with the primary one, and the disagreement is itself informative. Diebold–Mariano rejects at $p = 0.031$ and 0.030 , and so does the lap-level test of Section 6.4; read alone, either would suggest a result. But the model is ahead in only 12 of 22 races on the hold-out and 22 of 48 out-of-time, and the sign test returns 0.832 and 0.665: barely better than a coin toss over races. Diebold–Mariano corrects for serial correlation within a series, not for the clustering of laps inside a race, so with tens of thousands of laps it retains far more apparent power than the data support. This is the concrete form the paper's general argument takes: a p of 0.03 on 19,463 laps and a p of 0.83 on 22 races describe the same comparison, and only the second answers the question a strategist would ask.

6.8 Where the model is weak

Per-circuit errors (Fig. 6) span 0.251 s at Montréal to 0.905 s at Spa-Francorchamps, a factor of 3.6. The worst circuits are long-lap, weather-volatile venues — Spa, Zandvoort, São Paulo — where lap-to-lap variance is intrinsically high. Training exposure does not explain the ordering: Spa appears in four training seasons and is worst, while Las Vegas, wholly unseen and served by a global prior, attains 0.582 s.

Traffic raises the error of both forecasts without removing the model's advantage. On the 44.1% of comparable laps that begin within 2.0 s of the car ahead — measured at the end of the previous lap, the same causal quantity the model sees — the model errs by 0.4508 s against 0.3430 s in clear air, and by 0.5375 s within 1.0 s. Persistence degrades in the same way (0.5377 s in traffic, 0.4013 s in clear air), and the advantage holds in both conditions: 16.2% in traffic ($t(21) = -4.69$, $p = 1.2 \times 10^{-4}$) and 14.6% in clear air ($t(21) = -3.47$, $p = 0.0023$), with the model ahead in 21 of 22 races in each. Had traffic laps been

forecast at the clear-air error, the overall MAE would fall by about 12%, which bounds what better traffic features could recover if traffic laps are otherwise comparable. We do not use thresholds on a lap's own time as a proxy for clean air, because they select on the outcome (Section 4.1).

7 Discussion

7.1 What this means for practice

A strategist choosing a pit lap needs a next-lap forecast. The frozen model supplies one at 0.3900 s, 12.1% better than a stint-aware baseline that carries a compound offset across stops and 15.1% better than a naive one. The 0.05 s reduction in mean error per lap is small against typical pit-stop time losses of 20–25 s: even if it accumulated without any cancellation over a 60-lap race it would amount to about 3 s, an upper bound, and far less than the gain from choosing the right strategy at all.

The stratification of Section 6.2 sharpens where the model earns that. It is worth most in the first five laps after a stop (30.6%) and when the comparison spans a stop (44.6%); in settled running after lap 11 its 10.2% advantage does not reach significance at race level. A deployment that consults the model at pit-window decisions and in out-laps is justified by this evidence; one that consults it every lap in steady-state running is not.

Two tempting uses of the model are not supported. Deriving a degradation forecast from the lap-time model does not establish degradation skill: Section 6.4 shows that the derived forecast's degradation error simply is its lap-time error. And re-fitting promptly after a rules change rather than accumulating history is not indicated: Section 6.5 shows that accumulated pre-era seasons improved the model (race-level $p \approx 0.02$ over five seeds).

7.2 What this means for the literature

Table 1 lists studies reporting accuracy without a naive comparison. Our own results show concretely what such numbers can conceal, and the clearest demonstration is a coefficient of determination.

The persistence forecast — repeat the previous lap, no model, no features — attains $R^2 = 0.9870$ on the 2023 hold-out. Our full-feature absolute-target configuration attains $R^2 = 0.9927$, a figure that would read as near-perfect in isolation, while being *worse* than that one-line forecast in mean absolute error (0.5098 s against 0.4591 s). On a target this autocorrelated, R^2 above 0.98 is the floor rather than an achievement, because almost all the variance being explained is variance between circuits. An R^2 reported without a naive comparison is therefore close to uninformative, and this is not a hypothetical: it is measured on our own models. Published figures are of exactly this size — Patro et al. [5] report a test R^2 of 0.9927 — without the naive comparison that would show whether they exceed the floor.

But reporting a baseline is not enough on its own. Three further conditions apply, each of which changed a conclusion here. The baseline must be *constructed with the information the practitioner has* (Section 6.2: 15.1% became 12.1%). The target must be *capable of testing the claim made about it* (Section 6.4:

the degradation comparison tests nothing). And the design must *vary one thing at a time* (Section 6.5: an era effect disappeared when the evaluation season was held fixed). A negative or positive result that fails any of these is an artefact of the protocol.

Cappello and Hoegh [2] are the closest existing practice, being the one study in this subfield we found that scores a degradation model against a structurally matched time-series alternative rather than only against other learned models. Whether their target shares the reducibility property of Section 6.4 is worth checking directly: a state-space formulation with a latent pace term need not differ to a lagged lap time as cleanly as ours does, and we make no claim about it here.

7.3 Why the residual formulation matters so much

Gradient-boosted trees partition feature space and fit a constant per leaf. Asked for absolute lap time across circuits spanning 60–120 s, the model must spend its first splits separating circuits, then fuel loads, then compounds, before any capacity reaches lap-to-lap variation. Given the residual, the constant is supplied exactly by the baseline and every split addresses the quantity of interest. The 7.1-fold error difference between the two condition-only cells measures that.

This is not specific to Formula 1. Any strongly autocorrelated target invites the same choice, and the ablation suggests the choice dominates feature engineering.

8 Threats to validity and limitations

Construct. The degradation target is reducible to lap time at a one-step horizon (Section 6.4). This is the most consequential limitation in the paper: no result here bears on whether degradation is forecastable. Resolving it requires multi-lap horizons or stint-level integrals of pace loss, and that is the first item of future work.

Internal. The 110% filter excludes laps by their own outcome. Section 6.2 repeats the headline comparison without it and finds 13.2%, so the conclusion is not an artefact of filtering, but every stratified figure in Table 4 does carry it. Features were built once and no hyperparameter search was run; the ablation compares formulations at fixed settings rather than at each cell's optimum, so a tuned absolute-target model might close some of its gap. Causal availability is established by two released tests (Section 4.3); they establish invariance to future laps and to the predicted lap's own outcome, which is not a proof of correctness of every feature but is a stronger guarantee than assertion alone.

External. Findings are specific to Formula 1 dry and mixed-condition racing 2019–2025 under FastF1's public timing feed. Weather fields are circuit-level, not car-level. No tyre-temperature, fuel-load or setup telemetry is available publicly, and team models with that access should do better. The out-of-time result (Section 6.6) bounds the decay across two seasons but says nothing about the 2026 regulation change.

Statistical. Moving the unit of inference from the lap to the race is the right correction for clustering, but it costs power: with 22 races the 10.2% steady-state advantage in Table 4 is non-significant and might

well be real. The paired t -test is also sensitive to a single discrepant race — 2025 has 23 of 24 races favouring the model and still returns $p = 0.120$ (Section 6.6) — which is why sign and Wilcoxon tests are reported alongside it. Four primary comparisons are Holm-corrected; the stratified, ablation-cell and per-season tests are exploratory and are not, so their individual p -values should be read as descriptive. We report a test that disagrees with our preferred conclusion (Diebold–Mariano on degradation, $p = 0.031$) rather than omitting it, and give our reason for preferring the race-level result in Section 6.7; a reader who weights the tests differently would reach a different conclusion about degradation, though Section 6.4 argues the target makes that conclusion uninterpretable either way.

Scope. This is a supervised forecasting study. It does not evaluate a strategy optimiser, and the mapping from forecast error to position gained is left to the simulation literature [15, 20, 21].

9 Conclusion

Formula 1 lap-time and tyre-degradation models are widely reported without the one comparison that makes their accuracy interpretable. Supplying it on 167,083 public laps, with the race as the unit of inference, gives three results.

A gradient-boosted lap-time model does beat a naive forecast: 15.1% against persistence, and **12.1% against a stint-aware baseline that carries a compound offset across pit stops**, which is the defensible figure because persistence is otherwise asked to compare laps on different tyres. The advantage is concentrated in transitions — 30.6% in the first five laps of a stint, 10.2% and not significant at race level from lap 11 — and survives removing the outcome-based lap filter (13.2%).

The advantage is a property of the target formulation. All three residual configurations beat the baseline; no absolute configuration does, the worst by a factor of seven. A study attributing such an advantage to its features would be attributing it to the wrong component. Applied without refitting, the frozen model retains the advantage two and three seasons on — 14.3% in 2024 and 9.8% in 2025 — though not at a constant level.

A stint-referenced degradation target cannot test degradation knowledge at a one-step horizon, because it differences to lap-time forecasting exactly: its persistence forecast is identical to the lap-time one on 100% of covered laps, and any lap-time model's degradation error is its lap-time error. We make no claim in either direction about degradation forecastability. The property holds for any target formed by subtracting a within-stint constant, and should be checked before such a target is used to claim degradation skill.

Training era matters less than a confounded design suggested. Holding the 2023 evaluation set fixed and averaging five seeds, pre-regulation training reaches 11.9% against 12.2% for same-era training, indistinguishable at race level, and the full corpus including three superseded seasons reaches 16.1%, better than either — accumulating history helped.

The practical recommendation is narrow and follows from the evidence: consult such a model at stint transitions, use a stint-aware baseline to measure it, and do not treat a one-step degradation result as evidence about tyres. The broader one is that reporting a naive baseline is necessary but not sufficient – it must also be fairly constructed, the target must be able to test the claim, and the design must vary one thing at a time.

Declarations

Supplementary information. Online Resource 1 (ESM_1_Data and Code.zip) contains the analysed lap records for 2019–2025, the frozen model artefacts and circuit priors, all archived result files, and the analysis and verification code, with a note (Note on Resource.docx) describing data provenance and how to reproduce every result.

Competing interests.

The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval.

Not applicable. This study uses publicly available motorsport timing data and involves no human or animal participants.

Consent to participate

and consent for publication. Not applicable.

Funding.

The authors did not receive support from any organization for the submitted work.

Author Contribution

****T.Shakila.**** conceived the study, conducted the data analysis, developed and evaluated the models, and prepared the manuscript. ****K.Baalaji.**** supervised the research, contributed to the interpretation of the results, and reviewed and revised the manuscript. ****All authors reviewed and approved the final manuscript.****

Data Availability

The datasets generated during and/or analyzed during the current study are available in the Zenodo repository, <https://doi.org/10.5281/zenodo.23079502>

Code availability.

All code is included in the supplementary information (Online Resource 1): acquisition, feature construction, training, evaluation, statistical testing, figure generation, the two causal-availability tests, the baseline-fairness and traffic analyses, and the manuscript-verification script.

References

1. Kara MF, Küçükmanisa A. (2025) Predicting lap times in Formula 1: a deep learning approach using LSTM networks. In: 2025 IEEE 21st International Conference on Intelligent Computer Communication and Processing (ICCP). IEEE, pp 1–5. <https://doi.org/10.1109/ICCP68926.2025.11427134>
2. Cappello C, Hoegh A. A state-space approach to modeling tire degradation in Formula 1 racing. *J Sports Analytics*. 2026;12:22150218261446170. <https://doi.org/10.1177/22150218261446170>.
3. Sasikumar A, Leema AA, Balakrishnan P. Data-driven pit stop decision support for Formula 1 using deep learning models. *Front Artif Intell*. 2025;8. <https://doi.org/10.3389/frai.2025.1673148>.
4. Thomas D, Jiang J, Kori A, Russo A, Winkler S, Sale S, McMillan J, Belardinelli F, Rago A. Race strategy reinforcement learning: optimising pitstop strategy with emergent tactics in Formula One. *Mach Learn*. 2026;115(6):146. <https://doi.org/10.1007/s10994-026-07081-3>.
5. Patro AR, Sahana R, Gautam S, Sapna R. (2025) A predictive analysis of lap time in Formula 1 using machine learning. In: 2025 IEEE 6th India Council International Subsections Conference (INDISCON). IEEE, pp 1–7. <https://doi.org/10.1109/INDISCON66021.2025.11252332>
6. Mistry V, Smith WAP. (2025) AutoF1: an RNN-based approach to simulating strategic decision-making in Formula 1. In: 2025 IEEE 28th International Conference on Intelligent Transportation Systems (ITSC). IEEE, pp 4583–4590. <https://doi.org/10.1109/ITSC60802.2025.11423678>
7. Hyndman RJ, Koehler AB. Another look at measures of forecast accuracy. *Int J Forecast*. 2006;22(4):679–88. <https://doi.org/10.1016/j.ijforecast.2006.03.001>.
8. Makridakis S, Spiliotis E, Assimakopoulos V. The M4 competition: 100,000 time series and 61 forecasting methods. *Int J Forecast*. 2020;36(1):54–74. <https://doi.org/10.1016/j.ijforecast.2019.04.014>.
9. Zeng A, Chen M, Zhang L, Xu Q. (2023) Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence* 37(9):11121–11128. <https://doi.org/10.1609/aaai.v37i9.26317>
10. Makridakis S, Spiliotis E, Assimakopoulos V. Statistical and machine learning forecasting methods: concerns and ways forward. *PLoS ONE*. 2018;13(3):e0194889.

<https://doi.org/10.1371/journal.pone.0194889>.

11. Hewamalage H, Ackermann K, Bergmeir C. Data Min Knowl Disc. 2023;37(2):788–832. <https://doi.org/10.1007/s10618-022-00894-5>. Forecast evaluation for data scientists: common pitfalls and best practices.
12. Cerqueira V, Torgo L, Mozetič I. Evaluating time series forecasting models: an empirical study on performance estimation methods. Mach Learn. 2020;109(11):1997–2028. <https://doi.org/10.1007/s10994-020-05910-7>.
13. Zhao Z. Deep neural network-based lap time forecasting of Formula 1 racing. Appl Comput Eng. 2024;47(1):61–6. <https://doi.org/10.54254/2755-2721/47/20241191>.
14. Tulabandhula T, Rudin C. Tire changes, fresh air, and yellow flags: challenges in predictive analytics for professional racing. Big Data. 2014;2(2):97–112. <https://doi.org/10.1089/big.2014.0018>.
15. Heilmeyer A, Graf M, Betz J, Lienkamp M. Application of Monte Carlo methods to consider probabilistic effects in a race simulation for circuit motorsport. Appl Sci. 2020;10(12):4229. <https://doi.org/10.3390/app10124229>.
16. Heilmeyer A, Graf M, Lienkamp M. (2018) A race simulation for strategy decisions in circuit motorsports. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, pp 2986–2993. <https://doi.org/10.1109/ITSC.2018.8570012>
17. Todd J, Jiang J, Russo A, Winkler S, Sale S, McMillan J, Rago A. (2025) Explainable time series prediction of tyre energy in Formula One race strategy. In: Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing. ACM, pp 1082–1089. <https://doi.org/10.1145/3672608.3707765>
18. Shakila T, Baalaji K. (2025) Optimizing tyre and brake performance in Formula 1 using big data analytics: a survey on predictive models and strategies. In: 2025 International Conference on Emerging Trends in Industry 4.0 Technologies (ICETI4T). IEEE, pp 1–6. <https://doi.org/10.1109/ICETI4T63625.2025.11132168>
19. Thomas D, Jiang J, Kori A, Russo A, Winkler S, Sale S, McMillan J, Belardinelli F, Rago A. (2025) Explainable reinforcement learning for Formula One race strategy. In: Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing. ACM, pp 1090–1097. <https://doi.org/10.1145/3672608.3707766>
20. Heilmeyer A, Thomaser A, Graf M, Betz J. Virtual strategy engineer: using artificial neural networks for making race strategy decisions in circuit motorsport. Appl Sci. 2020;10(21):7805. <https://doi.org/10.3390/app10217805>.
21. Avella P, Borruso M, Marino M, Masone A. A simulation framework for Formula 1 race strategy based on pit-stop optimization. Simul Model Pract Theory. 2026;153:103341. <https://doi.org/10.1016/j.simpat.2026.103341>.
22. Liu X, Fotouhi A. Formula-E race strategy development using artificial neural networks and Monte Carlo tree search. Neural Comput Appl. 2020;32(18):15191–207. <https://doi.org/10.1007/s00521-020-04871-1>.

23. Aguard F, Thraves C. Optimizing pit stop strategies in Formula 1 with dynamic programming and game theory. *Eur J Oper Res.* 2024;319(3):908–19. <https://doi.org/10.1016/j.ejor.2024.07.011>.
24. Carrasco Heine OF, Thraves C. On the optimization of pit stop strategies via dynamic programming. *CEJOR.* 2023;31(1):239–68. <https://doi.org/10.1007/s10100-022-00806-4>.
25. Schäfer P. (2026) FastF1: Python package for accessing and analyzing Formula 1 results, schedules, timing data and telemetry. Version 3.8.2. Python Package Index. <https://pypi.org/project/fastf1/3.8.2/>
26. Shakila T, Baalaji K. (2026) Baseline-relative evaluation of lap-time and tyre-degradation models for Formula 1 race strategy [Dataset]. Zenodo. <https://doi.org/10.5281/zenodo.23079502>.
27. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y. (2017) LightGBM: a highly efficient gradient boosting decision tree. In: *Advances in Neural Information Processing Systems* 30, pp 3146–3154. <https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html>
28. Diebold FX, Mariano RS. Comparing predictive accuracy. *J Bus Economic Stat.* 1995;13(3):253–63. <https://doi.org/10.1080/07350015.1995.10524599>.
29. Harvey D, Leybourne S, Newbold P. Testing the equality of prediction mean squared errors. *Int J Forecast.* 1997;13(2):281–91. [https://doi.org/10.1016/S0169-2070\(96\)00719-4](https://doi.org/10.1016/S0169-2070(96)00719-4).
30. Holm S. A simple sequentially rejective multiple test procedure. *Scand J Stat.* 1979;6(2):65–70. <https://www.jstor.org/stable/4615733>.

Figures

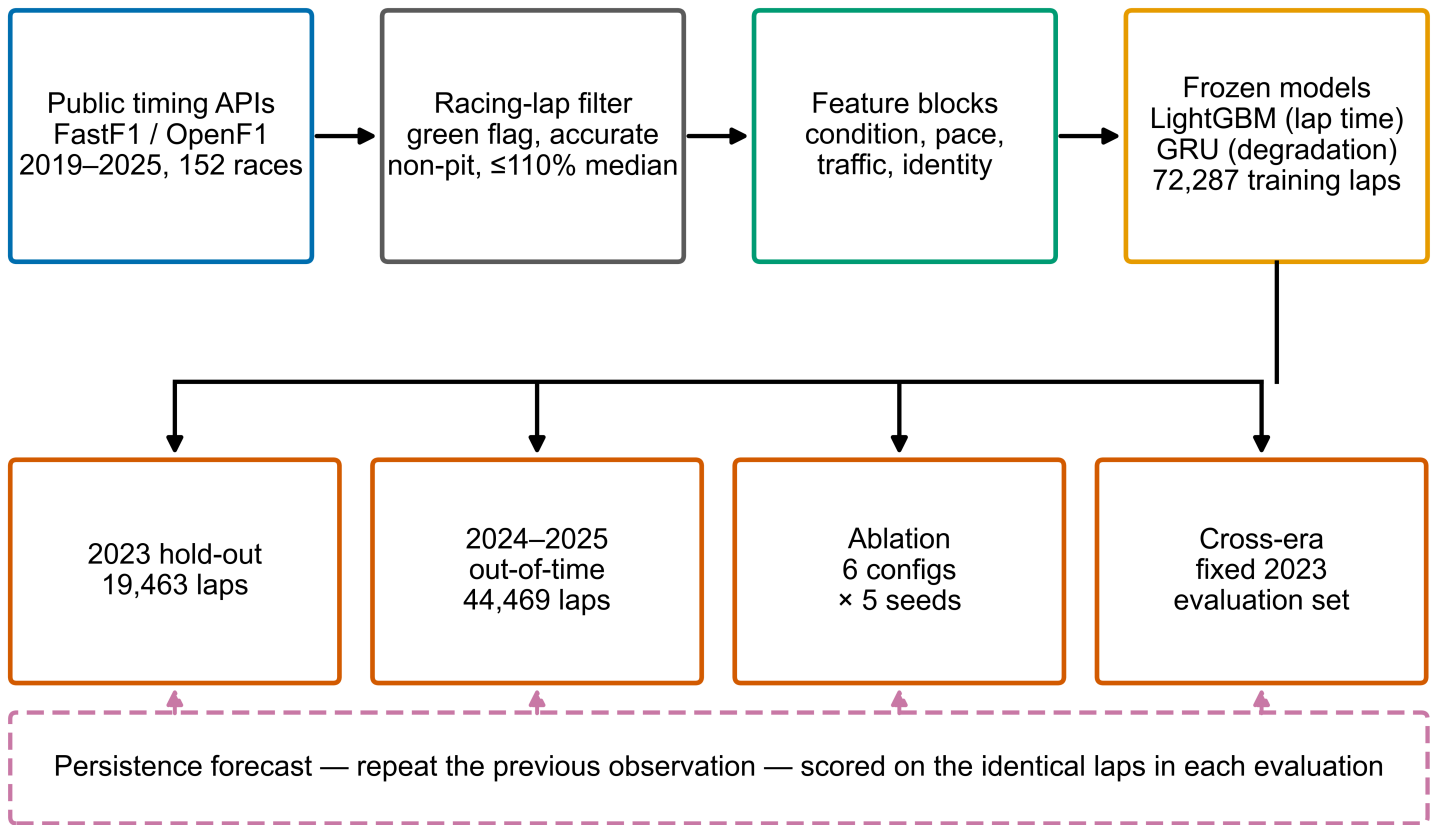


Figure 1

Study workflow. The upper row shows acquisition from the public timing interfaces, racing-lap filtering, feature-block construction and model fitting, with artefacts frozen after training on 2019–2022. The frozen models feed the evaluations of the lower row. The dashed box denotes the persistence forecast, scored on the identical laps in each evaluation. Lap and race counts are read from the archived result files

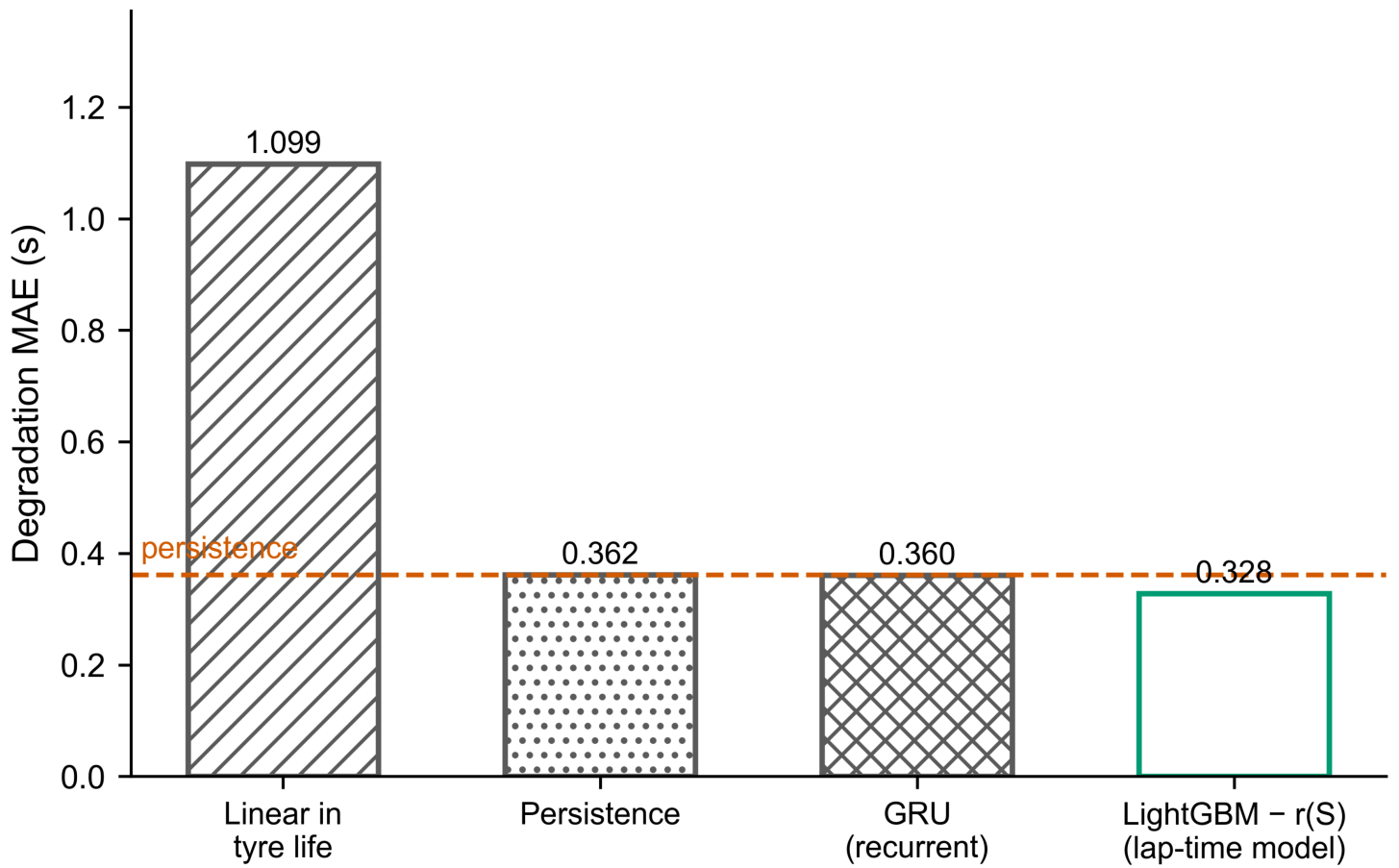


Figure 2

Tyre-degradation mean absolute error on the 2023 hold-out. The recurrent model is indistinguishable from persistence, whereas the frozen lap-time model minus the stint reference improves on it only because, by Eq. (4), its error is its lap-time error; the dashed line marks persistence

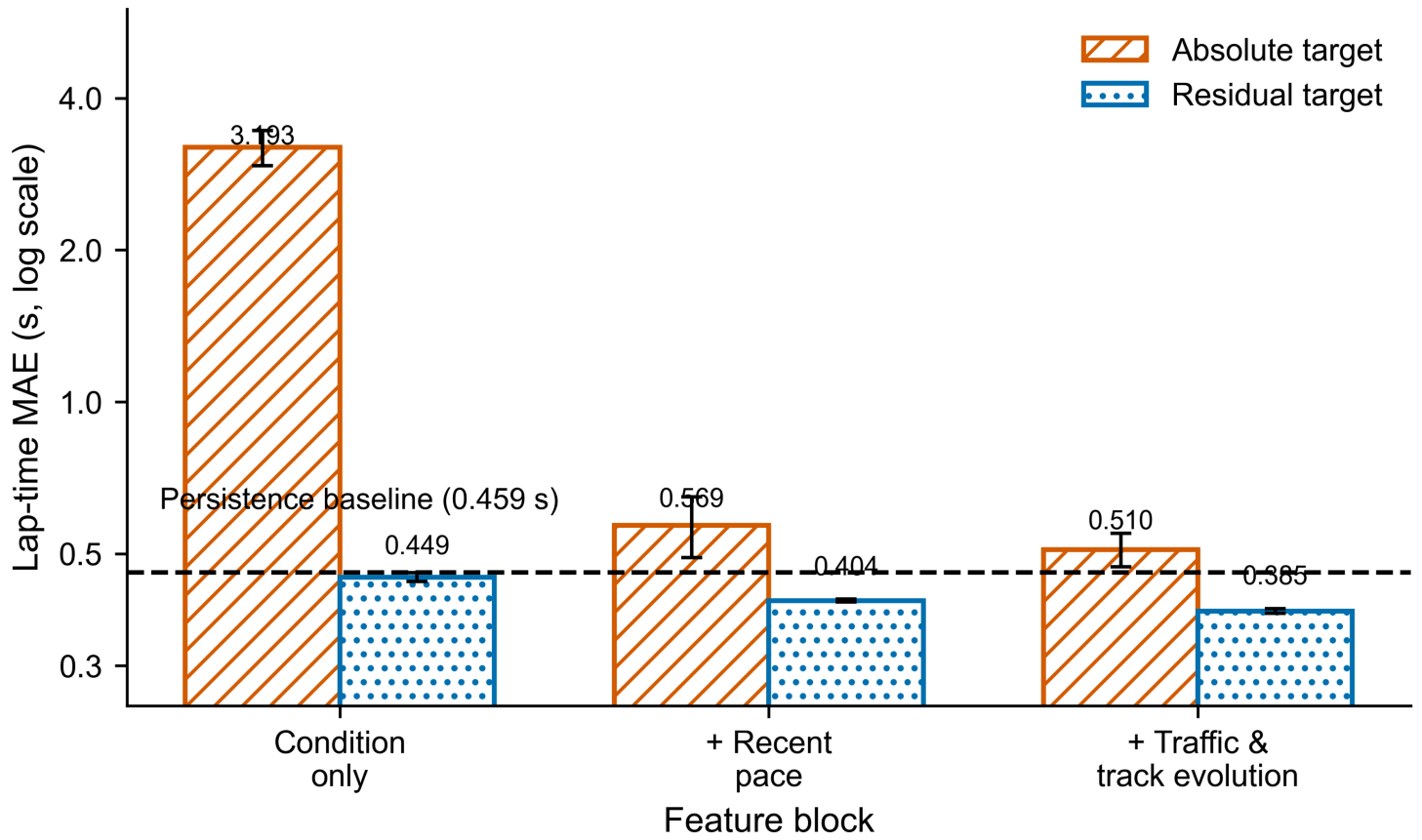


Figure 3

Lap-time mean absolute error by feature block and target formulation, five seeds per configuration, error bars showing one standard deviation. Bars are distinguished by hatch and outline; the dashed line marks the persistence baseline. The vertical axis is logarithmic

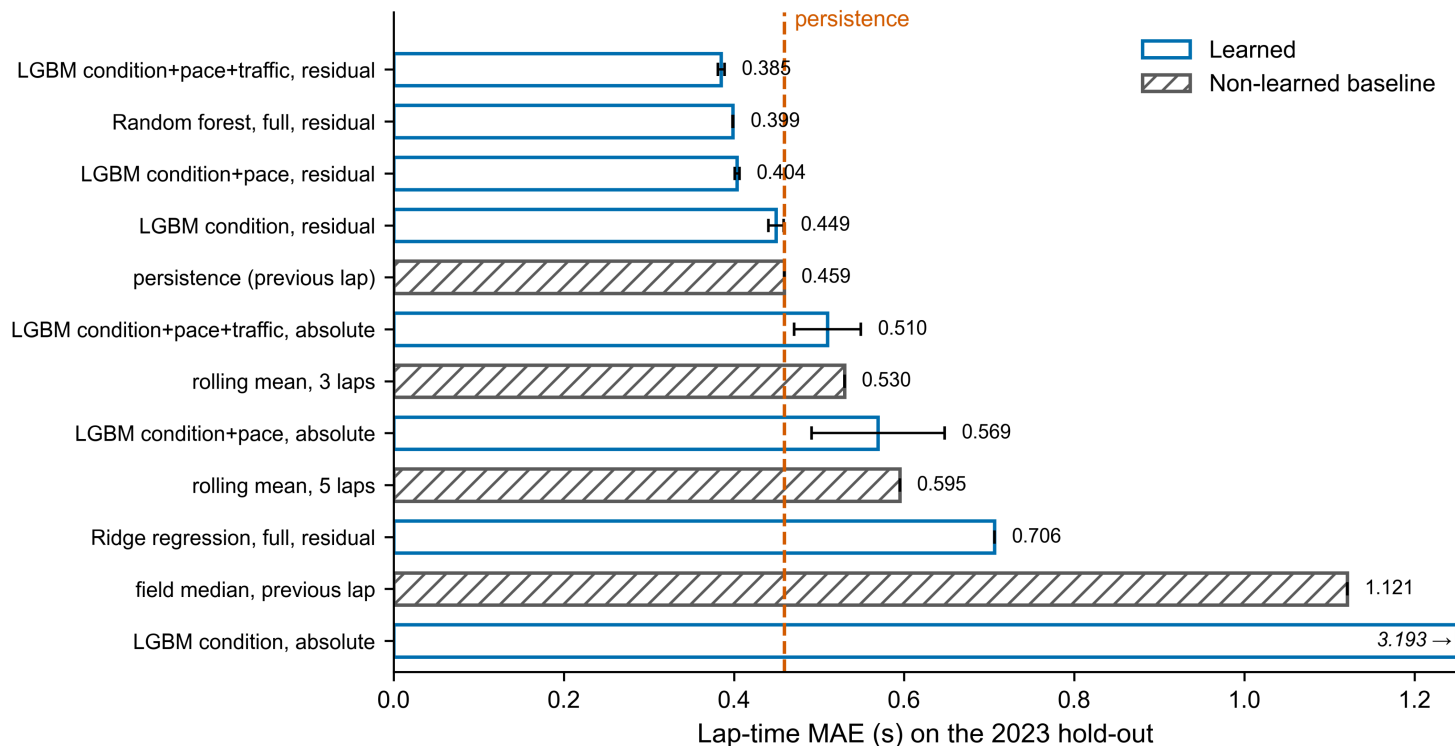


Figure 4

All lap-time configurations ranked by hold-out mean absolute error. Solid outlines denote learned configurations and hatched fills non-learned baselines; the dashed vertical line marks persistence. One configuration lies beyond the axis limit and is annotated with its value

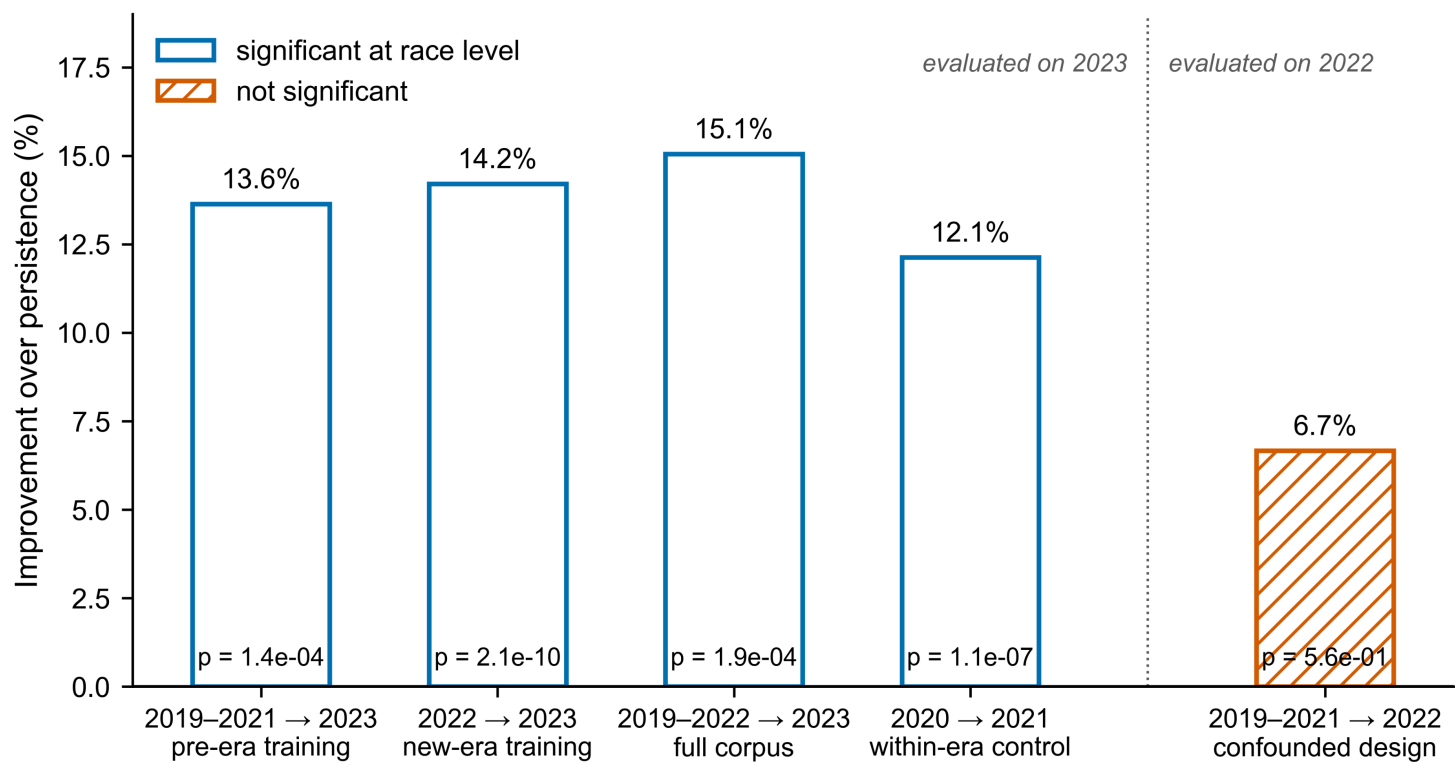


Figure 5

Improvement over persistence by training era. The four bars left of the dotted rule differ only in training data and are evaluated on the identical 2023 laps; the bar to its right is the confounded design, in which the evaluation season changes with the training era. Bars are labelled with the race-level p-value, and the hatched bar is not significant

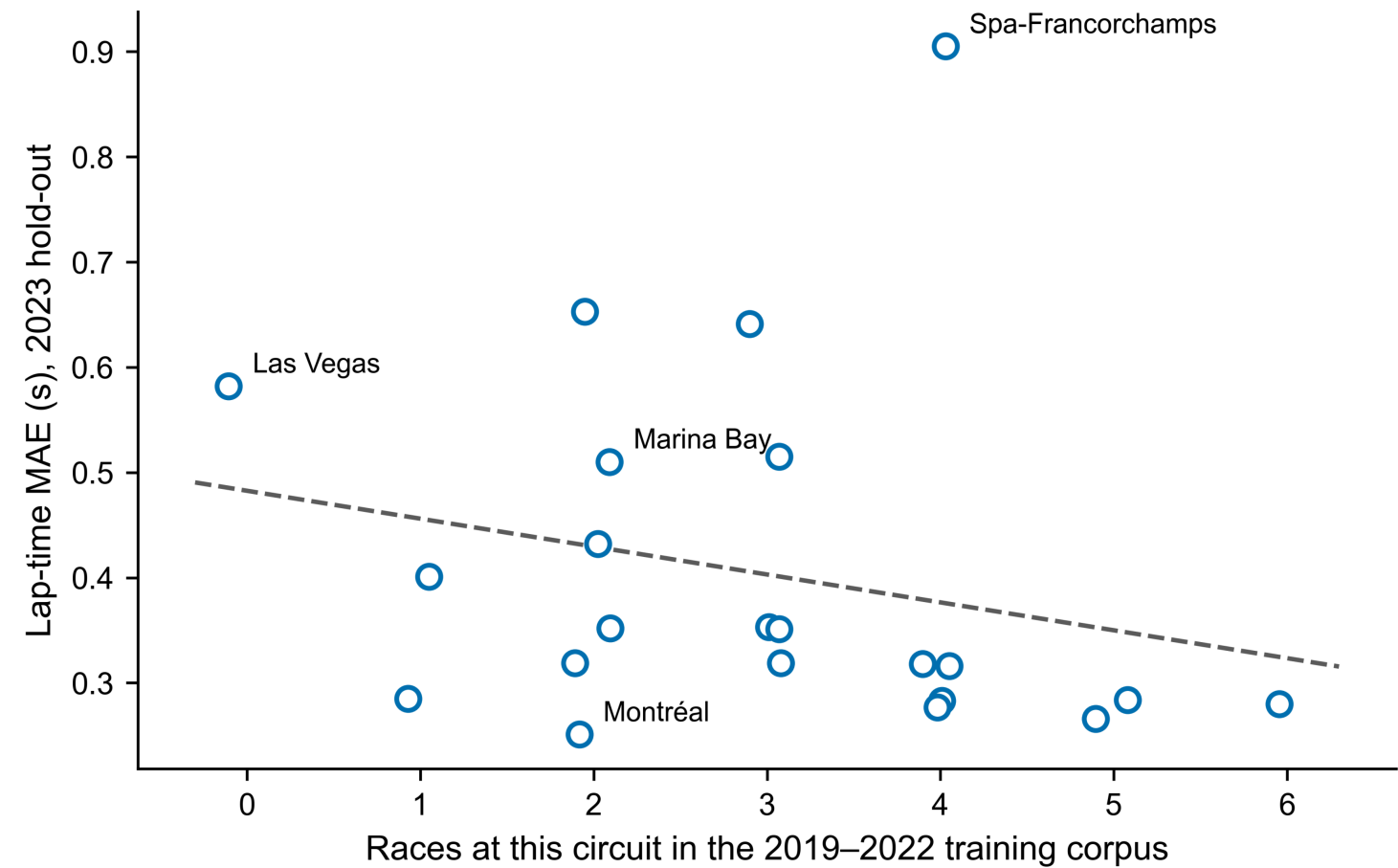


Figure 6

Per-circuit lap-time mean absolute error on the 2023 hold-out against the number of races that circuit contributes to the 2019–2022 training corpus, one marker per circuit, with a least-squares trend line. Four circuits discussed in the text are labelled