Pace Deficit & Starting Grid → Finishing Position in F1

Suneel Chandra Vanamu

2025-10-06

This paper asks a practical question in modern Formula 1: how do **on-day pace** (best-lap deficit) and **track position** (starting grid) each relate to where a driver **finishes**? The motivation is simple—teams invest heavily in qualifying for clean air, and race commentary constantly points to raw speed—so a simple, single-number relationship can be useful without complex models. Using the public **F1DB** CSV release (2014–present), I fit two single-predictor linear regressions: best-lap deficit → finishing position, and grid position → finishing position. The grid model shows a stronger, steadier link; the pace model is meaningful but noisier. I report **R²/adj-R²**, **residual standard error (positions)**, **slope + 95% CI**, and basic diagnostics.

1 Introduction

Formula 1 results can hinge on many moving parts—strategy, traffic, weather—but two simple signals are always front and center: **raw pace** and **track position**. This paper asks a single, practical question: How do a driver's **best-lap pace deficit** and **starting grid position** each affect where they **finish**? In plain terms, if a driver is slower on their best lap, or if they start farther back, do they tend to finish farther back as well?

This question matters because teams pour effort into **qualifying** for clean air, and commentators constantly point to **speed** as the difference-maker. A clear, one-number relationship helps analysts discuss race outcomes without building complex models. To make fair comparisons across circuits and seasons, I focus on the **2014–present hybrid era** and use **finishing position** as the common outcome (**1 = winner**). Pace is measured as the gap between a driver's best race lap and the race's fastest lap (**"best-lap deficit," in seconds**). **Starting grid position** is the driver's place on the grid at lights out.

Using the **F1DB CSV** release, I assemble a driver–race table for this era and fit two **one-predictor linear regressions**: (1) **best-lap deficit → finishing position**, and (2) **starting grid position → finishing position**. For each model I report how tightly the predictor lines up with finishing order (**R²/adjusted R²**), the model's typical miss in positions (**residual standard error**), and the average change in finishing place for a one-unit change in the predictor (**slope with a 95% confidence interval**). I also include simple **scatterplots** with a **regression line** and basic **residual checks**.

The remainder of this paper is organized as follows. **Section 2** describes the dataset and key variables. **Section 3** provides bivariate analysis. **Section 4** builds and evaluates the best-lap-deficit and starting-grid models. **Section 5** discusses the results and their interpretation. **Section 6** summarizes main takeaways, practical implications, and brief robustness ideas.

Project repository: https://github.com/Suneel1508/MATH261A-project1

2 Data and Variables

I use the public F1DB dataset (F1DB contributors 2025). Files are read with readr (Wickham, Hester, and François 2024), tidied with janitor (Firke 2023) and tidyr (Wickham and Girlich 2024), and string fields parsed with stringr (Wickham 2023). Visuals use ggplot2 (Wickham 2016) within R (R Core Team 2024). To keep things comparable across tracks and seasons, I focus on the 2014–present hybrid era. For each driver in each race, I pull three core fields:

- Finishing position (response): integer rank, where 1 = winner.
- Best-lap pace deficit (sec) (predictor A): how much slower a driver's best race lap was compared with the race's fastest lap (small, continuous values in seconds).
- Starting grid position (predictor B): integer rank at the start lights (1 = pole, higher = farther back).

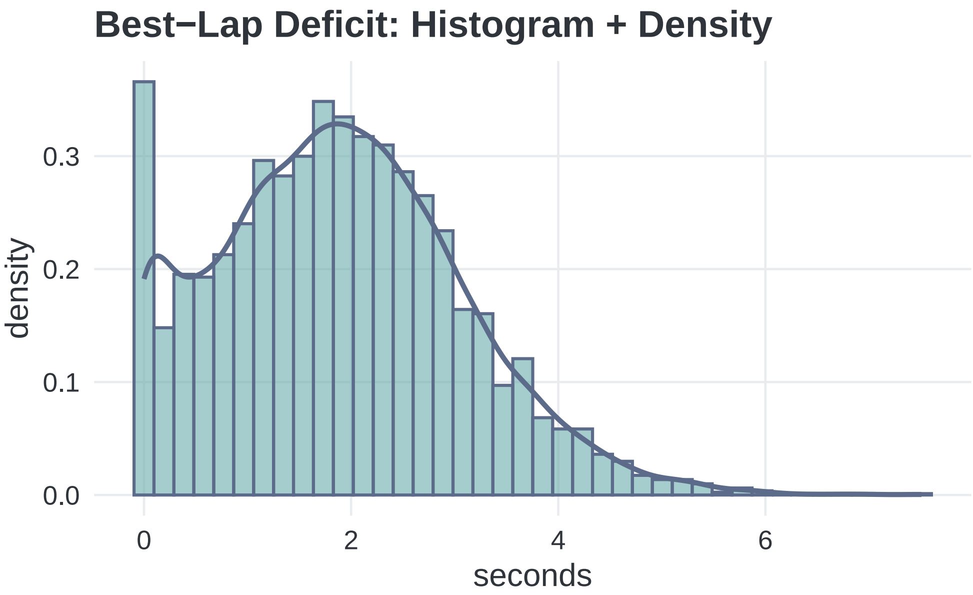

Figure 1: Distribution of best-lap deficit (2014–present). Most drivers sit within a few seconds of the race's fastest lap; a long right tail reflects incidents and traffic.

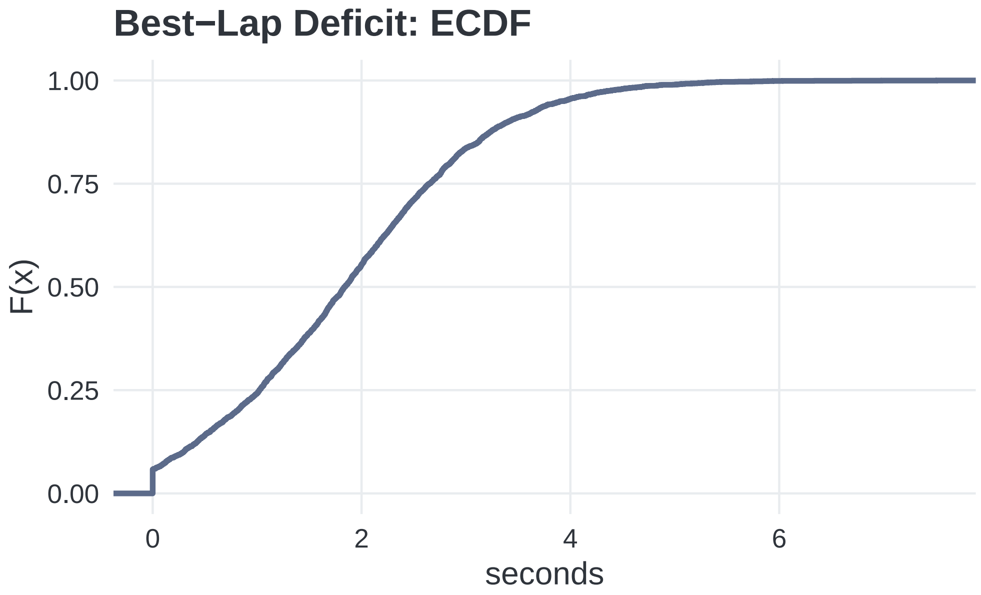

Figure 2: ECDF of best-lap deficit. About half the field is within ~1–1.5s of the fastest lap; only a small fraction exceed ~3–4s.

The **best-lap deficit** is continuous and measured in seconds, so I show its **histogram with a density curve** and an **ECDF**. The histogram suggests most drivers' best laps sit within a small band—roughly a couple of seconds—of the race's fastest lap, with a **right-hand tail** of larger deficits (cars in traffic, tire issues, etc.). The **ECDF** (Empirical Cumulative Distribution Function) reads like a running total: at any x-value on the horizontal axis, the y-value shows the **fraction of drivers whose deficit is x seconds**. The smooth S-shape you see means deficits accumulate steadily, with very few extreme outliers.

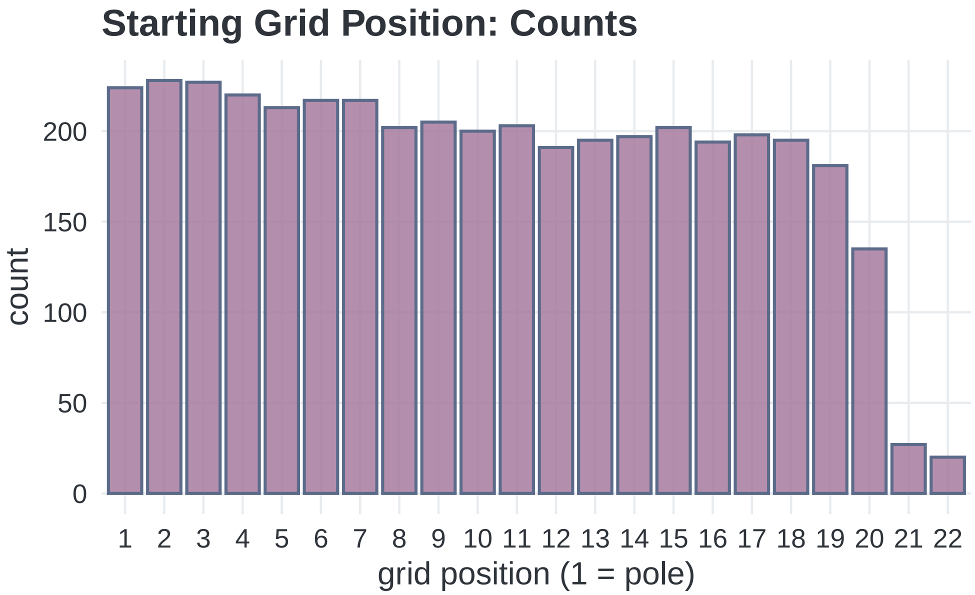

Figure 3: Starting grid positions used (2014–present). Discrete ranks produce vertical bands in later scatterplots.

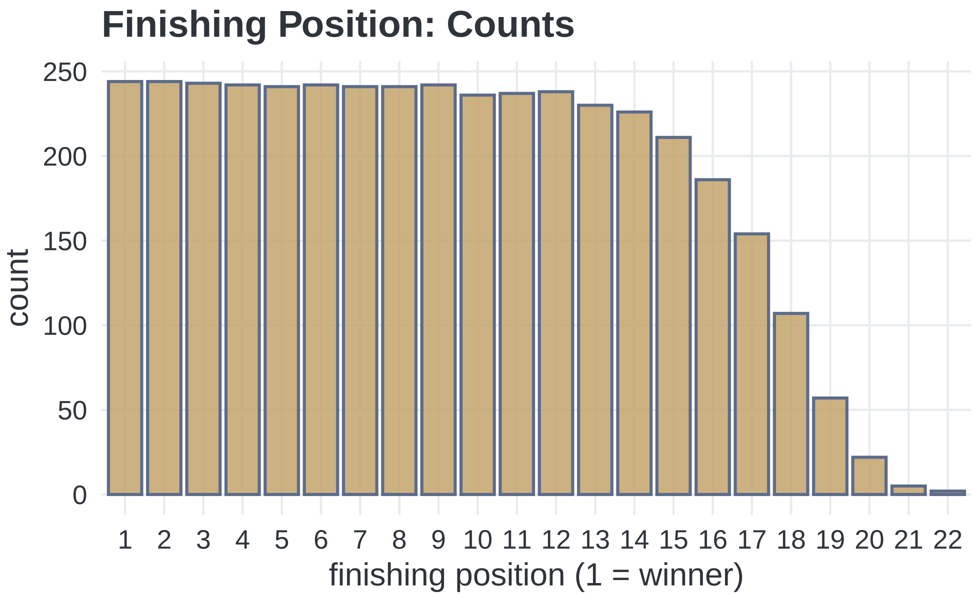

Figure 4: Finishing positions (1 = winner). Winners create a spike at 1; mass spreads gradually toward the midfield and backmarkers.

The other two variables are **discrete ranks**, so I use **bar charts** instead of box plots. The **Starting**

Grid Position plot is fairly spread across the grid, which is what we expect over many races. The **Finishing Position** plot naturally **tapers at the back** because not every race classifies all 20–22 cars; retirements and lap-downs thin out the tail. These bars make it clear we're modeling integer outcomes, and they also hint that moving from, say, P4 to P6 is not the same kind of jump as moving from P18 to P20 in terms of frequency.

A few light data steps sit behind these pictures. I join race results to the fastest-laps table to compute the **best-lap deficit** per driver per race, then keep a **single row per driver–race** with the three fields above. I also restrict to the hybrid era using the provided year field. This keeps things apples-to-apples across circuits and seasons without introducing extra adjustments.

These visuals do two jobs. First, they **set scale**: finishing position is an integer rank, grid is an integer rank, and pace deficit is a small, continuous number measured in seconds. Second, they **flag shape**: pace deficits are right-skewed, while grid/finish are count-based. With this context in place, Section 3 takes the next step—a quick look at how each predictor lines up with finishing position—before we fit the two simple regressions.

Missingness & filters. I keep one row per driver–race in 2014+ with **numeric finishing position** and the predictor needed for each model. Practically, that means I require finite values for `position_number` (removing DNS/DSQ/DNF entries lacking a numeric finish), `best_lap_delta_sec` for pace models, and `gap_sec` for gap models. Winners have `gap_sec = 0` by definition. This ensures summaries and fits are based on complete, comparable records.

3 Quick Bivariate EDA

Before fitting any models, I want to quickly visualize how each predictor relates to finishing position, as well as check for any relationship between the two predictors themselves, using ggplot2(Wickham 2016) and dplyr(Wickham et al. 2023). This isn't about proving anything yet—it's just a check for the direction of the trend, amount of spread, and any obvious curvature or outliers. I'll plot (a) **best-lap deficit → finish**, (b) **grid → finish**, and (c) **grid** **best-lap deficit** with simple rank-based correlations.

3.1 Best-lap deficit → finishing position

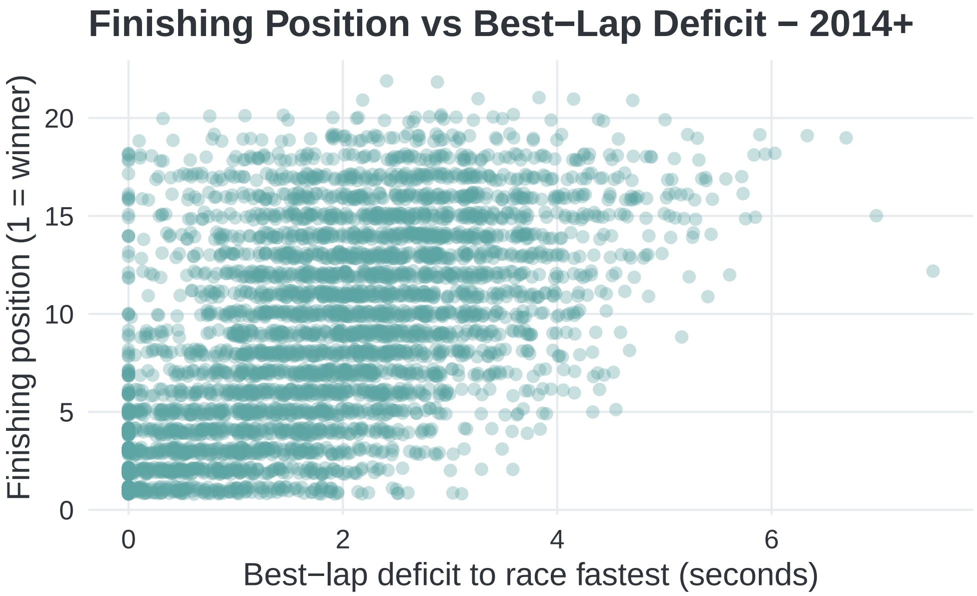

Figure 5: Finishing position vs best-lap deficit (points only). Clear upward tilt—larger deficits usually mean worse finishes—but with wide spread.

The cloud **tilts upward**: larger best-lap deficits generally pair with **worse finishing positions**. The signal is there, but the spread is wide—at the same deficit you can land anywhere from the top ten to the teens. Near **zero deficit**, many finishes cluster toward the front but it's not guaranteed (traffic/strategy can shuffle results). Past roughly **3–4 seconds**, front-running finishes are uncommon, which matches intuition. Overall: a **meaningful but noisy** relationship; a simple linear fit later should capture the direction, with only **moderate** R^2.

3.2 Starting grid position → finishing position

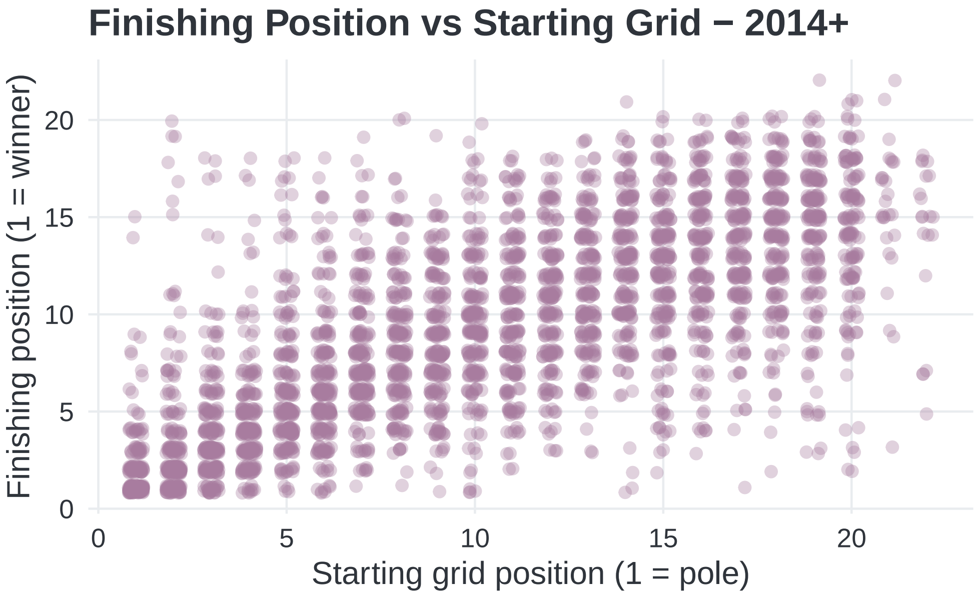

Figure 6: Finishing position vs starting grid (points only). A staircase pattern: deeper grid slots generally align with worse finishes.

There's a clear **staircase pattern**: as **grid position increases**, finishing positions also rise. The vertical stripes are just the grid being discrete; what stands out is how the point bands climb steadily from left to right. Compared with the pace plot, the spread is **tighter**, especially near the front rows—track position really helps. I'd expect a **stronger fit** here (higher R² and a smaller residual error in positions) than in the pace-deficit model.

3.3 Are the predictors related?

```
# A tibble: 1 x 2
  Spearman Kendall
     <dbl>   <dbl>
1    0.511   0.366
```

The jitter plot **leans upward**, and the rank correlations back it up: **Spearman 0.51** and **Kendall 0.37** indicate a **moderate positive association**. In plain terms, cars that start farther back often also have a larger pace deficit—but the link isn't tight enough to call them duplicates. Each predictor carries **distinct information** about finishing order.

Why these? Spearman ( ) and Kendall ( ) are **rank-based** correlation measures: they capture **monotonic** association without assuming linearity and are robust to outliers. Here they summarize whether starting deeper on the grid tends to go with a larger pace deficit; moderate positive values mean "usually yes," but not a one-to-one link.

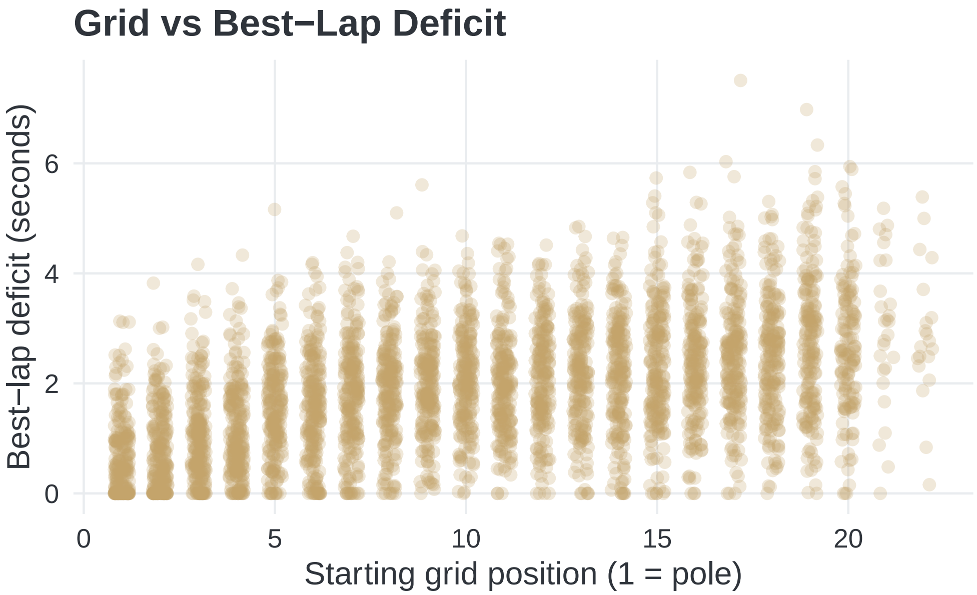

Figure 7: Starting grid vs best-lap deficit. Moderate positive association: cars starting farther back often have larger pace deficits, but not one-to-one.

Linear fits are reasonable (no obvious curvature), but we should expect **grid → finish** to explain more variance than **pace → finish**. We'll quantify that in Results with **R²**, **residual standard error (positions)**, and **slope + 95% CI**, and we'll add residual plots to check assumptions.

4 Methods

To answer the question "How do a driver's best-lap pace deficit and starting grid position each affect **finishing position**?", I fit **two separate one-predictor linear regressions** on the 2014–present data. I keep the setup simple on purpose so the effects are easy to read in plain units (positions and seconds).

4.1 Model A — Best-lap pace deficit → Finishing position

Let FinishPos_i be driver $i's$ finishing position (1 = winner) and Deficit_i the driver's best-lap pace deficit in seconds (their best race lap minus the race's fastest lap). I fit a simple line:

$$\text{FinishPos}_i = \beta_0 + \beta_1 * \text{Deficit}_i + \varepsilon_i$$

Here, β is the line's intercept (the model's expected finishing position when the deficit is zero—i.e., when a driver's best lap matches the day's fastest lap). β_1 is the **average change in finishing position per +1 second** of deficit (e.g., $\beta_1 = 1.4$ means ~1.4 places farther back for each extra second). The error term ε_i captures race noise we're not modeling (strategy, safety cars, incidents). I'll report **R²/adj-R²**, **Residual SE** (typical miss, in positions), **Residual SE²**, and the **slope with 95% CI**.

4.2 Model B — Starting grid position → Finishing position

Let FinishPos_i be finishing position and Grid_i the starting grid slot (1 = pole; higher = farther back). The model is:

$$\text{FinishPos}_i = \gamma_0 + \gamma_1 \,\text{Grid}_i + \eta_i$$

Here, γ_1 is the **average change in finishing position per one grid place farther back** (we expect a positive slope: deeper grid → worse finish). γ_0 is the intercept; since grid starts at 1, it mainly anchors the line and is most interpretable near the observed range (e.g., predicted finish at grid 1–2). As with Model A, I'll summarize **R²/adj-R²**, **Residual SE** (positions) and its square, plus the **slope with 95% CI** to show effect size and uncertainty.

I use ordinary least squares via `lm()` in R(R Core Team 2024); report rendered with **knitr** (Xie 2024). These one-variable models assume a roughly linear average effect and approximately constant spread of residuals; I'll check standard residual plots in Results. Because we use just one predictor at a time, the goal is clarity: quantify how much **one simple number** (pace deficit or grid slot) moves finishing position, in **positions per second** or **positions per grid place**, while acknowledging unmodeled race factors.

5 Results

We report results for two simple regressions on the 2014–present era:

- (A) **Best-lap pace deficit → finishing position**
- (B) **Starting grid position → finishing position**.

For each, I summarize the slope (effect size), how much variance is explained (R^2/adjusted R^2), and the model's typical miss in positions (Residual Standard Error, RSE). The fitted red-line scatters and residual diagnostics appear below the tables.

5.1 Best-lap deficit → finishing position

According to the model, the average change in finishing place per +1s of pace deficit is **2.349** positions (95% CI [**2.238, 2.461**]). The fit explains **0.29** of the variation in finishing order (adjusted R^2 = **0.29**), with a Residual SE of **4.33** positions (variance **18.71**). In plain terms, if the line predicts P9, results often land roughly **P9 ± 4.3**. This matches the earlier points-only plot: pace is meaningful but noisy.

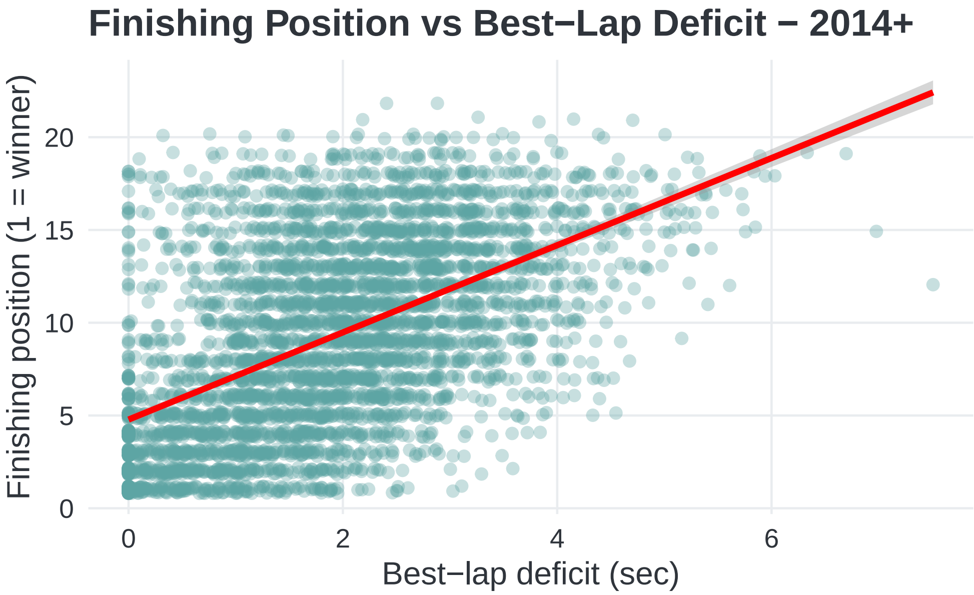

Figure 8: Finishing position vs best-lap deficit with linear fit. Upward slope quantifies positions lost per second of deficit; variability reflects race dynamics.

5.2 Starting grid position → finishing position

Here the slope is **0.655** positions per one grid place farther back (95% CI [**0.637, 0.673**]). The model explains **0.554** of finishing-order variation (adjusted R^2 = **0.554**), with a Residual SE of **3.41** positions (variance **11.66**). That tighter error and higher R^2 are consistent with the staircase we saw in the points-only plot: track position is a strong, steady signal.

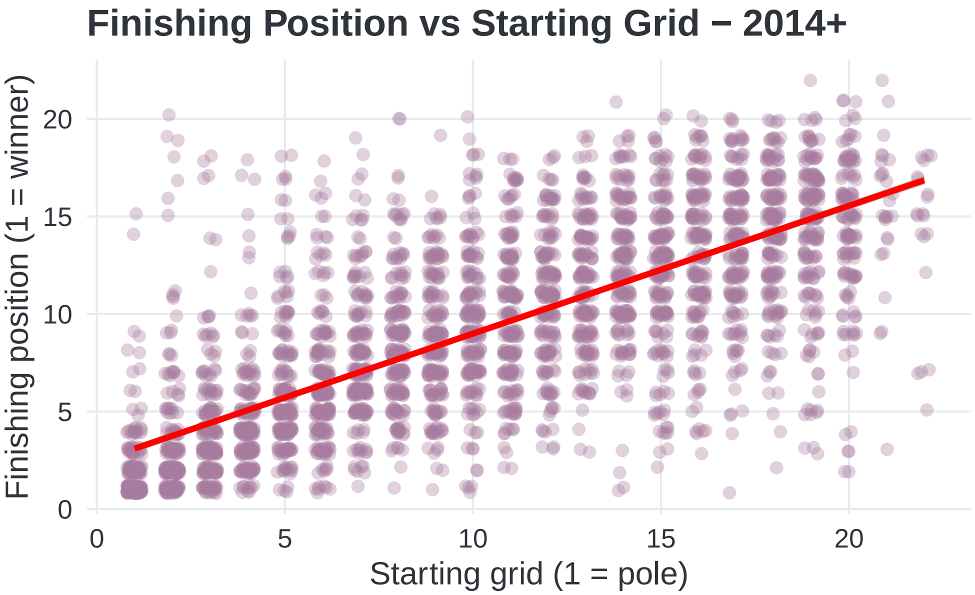

Figure 9: Finishing position vs starting grid with linear fit. Stronger, steadier upward trend than the pace plot; discrete grid slots create vertical bands.

The residuals-vs-fitted plots should look like centered bands without strong funnels.

Table 1: Model summaries (n, R², adjusted R², residual SE in positions, residual variance).

Model	n	r2	adj_r2	rse_pos	rse_sq
A: Best-lap deficit → finish	4175	0.290	0.290	4.325	18.708
B: Grid → finish	4091	0.554	0.554	3.415	11.662

Table 2: Slope estimates (effect on finishing position) with 95% CIs.

Model	Term	Estimate	Std. Error	CI low	CI high
A: Best-lap deficit → finish	Best-lap deficit (sec)	2.349	0.057	2.238	2.461
B: Grid → finish	Starting grid position	0.655	0.009	0.637	0.673

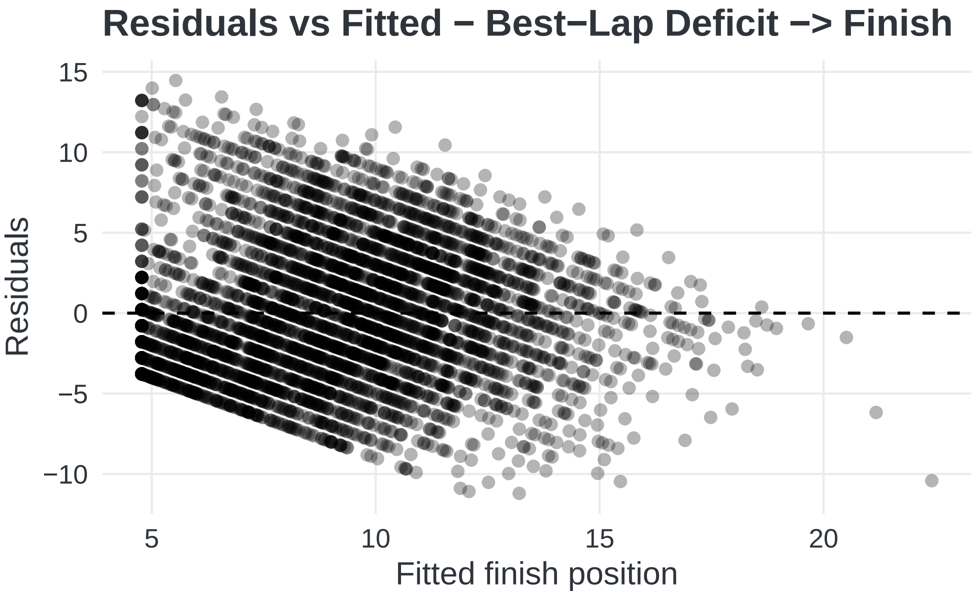

Figure 10: Residuals vs fitted — Best-lap deficit model. Band centered near zero; mild structure at extremes consistent with bounded ranks.

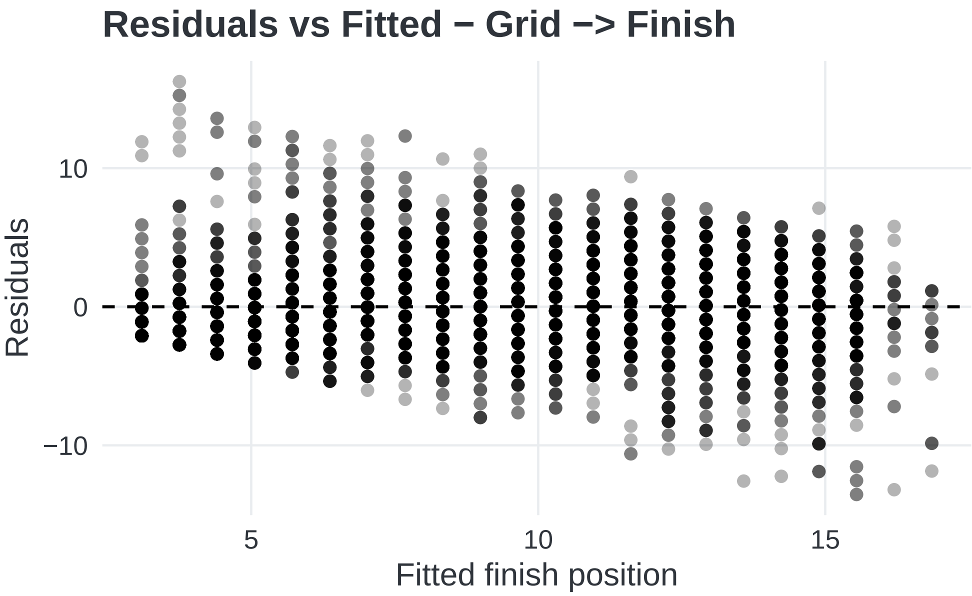

Figure 11: Residuals vs fitted — Grid model. Tighter, more horizontal band; no strong funnel, supporting a simple linear mean.

Both residual plots show the expected **striping** from an integer outcome, but the points stay **centered**

around zero overall. In the **pace → finish** model there's a **slight negative tilt**—the model is a bit optimistic for very strong predicted finishes and a bit pessimistic for weak ones—likely from the bounded scale and mild nonlinearity. The **grid → finish** band is **tighter and more horizontal**, with only small spreading at the extremes. Importantly, there's **no strong funnel or curvature**, so a linear mean with roughly constant variance is reasonable. For this project's one-predictor goal, the diagnostics look **acceptable**.

6 Discussion

This study used two simple linear regressions to see how **on-day pace** (best-lap deficit) and **track position** (starting grid slot) relate to **finishing position** in modern F1. Both signals matter, but the results are clear: **grid position is the stronger, steadier predictor**, while **pace deficit has a meaningful but noisier link**—consistent with what we saw in the bivariate plots and residual checks.

There are limits. I only analyzed the hybrid-era subset and used **one predictor at a time**, so I don't control for track layout, safety cars, tyre strategy, team/driver strength, or weather. Finishing position is an **integer and bounded**, so a straight line can't capture every wrinkle. Best-lap deficit is also a **single-lap snapshot** of pace rather than race-long speed.

Future work could fold in **track or season effects**, team/driver indicators, or **per-lap average pace**. Still, the takeaway for this project is practical and readable: **starting closer to the front reliably pulls the finish forward**, and **being even a second off the day's fastest lap tends to push it back**—quantified in plain units of **positions per grid place** and **positions per second**.

References

F1DB contributors. 2025. "F1DB: Formula 1 Database (CSV Release and Documentation)." GitHub repository. https://github.com/f1db/f1db.

Firke, Sam. 2023. *Janitor: Simple Tools for Examining and Cleaning Dirty Data*. https://CRAN.R-project.org/package=janitor.

R Core Team. 2024. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://doi.org/10.1007/978-3-319-24277-4.

———. 2023. *Stringr: Simple, Consistent Wrappers for Common String Operations*. https://CRAN.R-project.org/package=stringr.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2023. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wickham, Hadley, and Maximilian Girlich. 2024. *Tidyr: Tidy Messy Data*. https://CRAN.R-project.org/package=tidyr.

Wickham, Hadley, Jim Hester, and Romain François. 2024. *Readr: Read Rectangular Text Data*. https://CRAN.R-project.org/package=readr.

Xie, Yihui. 2024. *Knitr: A General-Purpose Package for Dynamic Report Generation in r*. https://CRAN.R-project.org/package=knitr.