Problem Set IV Solution

Tobias Bodentien Philipp Grunenberg Alexander Haas Osama Warshagha

29-01-2026

Task 1

In this task we design an reduced experimental design to study the impact of different car setup parameters and strategy-related factors. As seen in the last problem set we have several variables. Some of those variables are predetermined and track specific. Since they do not vary within the practice or the race and are fixed we cannot include them in the experimental design. The setup variables however can be influenced by the racing team. In this problem set we additionally have strategy relevant variables. We can now choose tire compound, fuel load and how many pitstops we do and when we do them. Since we do not have any previous data on the strategy relevant variables it is important to learn about the influences of those on the lap time and overall race time. It is of significant importance to find out how durable the tires are and how they influence the lap time. The same applies to fuel load. It is also important to do some pit stops to estimate how much time is lost per pit stop. Since the pit stops are determined by tire changes we do not explicitly include them as a factor in the experimental design as we learn about them indirectly. The fuel load will decrease during the practice. Therefore we do not include it explicitly as a factor in the experimental design and just fuel up the car to the maximum after every tire change.

The LASSO results of the previous problem set indicate that lap time is strongly influenced by several car setup parameters and their interactions with track and environmental conditions. Every retained variable involves at least one controllable car feature. In particular, brake balance, differential, engine, and front and rear wing settings appear repeatedly with sizable coefficients, suggesting that drivetrain and aerodynamic configuration are key performance drivers. Therefore, we choose all setup variables as strong drivers of performance.

From the previous problem set we know that the relationship between car setup variables and lap times is non-linear. While a two-level design can only identify general trends, the inclusion of a medium value allows the model to capture curvature and diminishing returns, which are critical for identifying the “sweet spot” in a complex mechanical system. However, a three-level design may oversimplify the complex physical interactions of racecar dynamics. A response surface with only three points assumes a symmetric curvature, which risks missing the true optimum if the performance curve is skewed or asymptotic. By extending the design to five levels, we gain the resolution necessary to capture higher-order non-linearities and subtle changes in vehicle behavior. This finer granularity is critical for distinguishing between a broad performance plateau and a narrow, sensitive peak, ensuring that the final optimal setting is precise rather than just an approximation between two extremes.

Regarding the tyres we expect that a softer compound results in faster lap times but higher degradation. For the qualifying the extra-soft compound therefore makes sense. It is still important to use the other three compounds to learn when it makes sense to use a softer tire with an additional pit stop as opposed to a harder tire. The resulting factors that we believe influence the lap time are:

- Rear Wing: 10 / 130 / 250 / 370 / 500
- Front Wing: 10 / 130 / 250 / 370 / 500
- Engine: 10 / 130 / 250 / 370 / 500

- Brake Balance: 10 / 130 / 250 / 370 / 500
- Differential: 10 / 130 / 250 / 370 / 500
- Suspension: 10 / 130 / 250 / 370 / 500
- Tire: super-soft / soft / medium / hard

From the Team Analytics Website we get the following information from Gunnar: "One thing we did was to debunk the myth that setup should be changed according to fuel load or the tyres." This implies that setup factors can be optimized independently of tires and fuel load. And strategy factors can be evaluated holding setup fixed. This allows us to separate the experiment into two phases.

In the first phase we estimate the main effects of the six setup variables, holding tire compound and fuel strategy constant. As a full factorial design is not feasible we use D-optimality to come up with an experimental design, which maximizes the precision of the car setup coefficients given a limited amount of experimental runs (70=120- 50 laps for phase 2).

The resulting design achieves a solid D-efficiency of 42,1% and maintains low correlations between the factors with the largest absolute correlation being 0,16. This ensures that the influence of each individual parameter can be statistically isolated during analysis. Consequently, this approach maximizes the information gathered about the vehicle's performance while significantly minimizing the total number of practice laps required for testing when compared to a full factorial design.

In the second phase the setup is fixed and we compare tire compounds to understand lap time and degradation trade-offs as well as implicit pit stop costs. Because the number of strategy factors is small and the factor levels are few, a full factorial design is feasible and preferred, as it allows unbiased and transparent comparison of all tire compounds without confounding. This full factorial design ensures that differences in lap time and degradation across tire types can be directly attributed to the tire choice, providing a clear basis for race strategy decisions. We assume that softer tire compounds degrade faster but achieve shorter lap times than harder compounds. To efficiently learn about degradation behavior under a limited lap budget, we run longer stints on the extreme compounds (super-soft and hard), which are expected to bracket the degradation patterns of the intermediate compounds. Shorter stints on the soft and medium tires are still included to directly observe their performance levels, while the longer stints on the extremes improve identification of degradation dynamics. Therefore we decide to run 15 laps on the super-soft, 25 laps on the hard tire and 5 laps on the soft and medium tire each.

```
##
## super_soft       soft     medium       hard
##         15          5          5         25
```

Initial testing with the 25-lap stint on Hard tires with a 120L fuel load revealed that both tire degradation and fuel consumption follow a linear trend. Interestingly, the data suggested that fuel weight has a more significant impact on lap times than tire wear, as lap times consistently decreased as the fuel load lightened and the tyres degraded.

We discovered that the Super-Soft compound could not be utilized, probably due to high ambient temperatures.

Based on those findings, we have refined our approach to focus more on the influence of fuel.

Testing during a 2-lap stint on Soft tires confirmed that fuel consumption remains linear regardless of the compound, while the degradation rate is significantly higher for softer tires.

Given that fuel and tire degradation operate independently, we adopted a "single-lap stint" strategy to isolate variables. By running multiple one-lap stints, we can hold the tire condition constant (fresh tires every time) to precisely measure the influence of fuel load on performance. Consequently, we conducted stints with fuel levels of 100, 80, 60, 40, and 20 liters. Repeating this process for both Soft and Hard tires allowed us to verify if the fuel-to-performance relationship remains consistent across different grip levels.

To finalize the tire data, we dedicated a longer stint to the Soft (15 laps) and a short stint to Medium (3 laps) compounds to map their specific linear degradation coefficients.

Critical Evaluation of the Design

The experimental design was structured to balance statistical rigor with the practical constraints of a 120-lap simulation budget. While the approach successfully isolated key performance drivers, it involved risks regarding error variance and sample distribution.

Decoupling based on Domain Expertise: By following Gunnar's insight, we successfully decoupled the car setup from the strategy variables. This allowed for a two-phase approach that reduced the complexity of the design space, preventing the "curse of dimensionality" that would have occurred in a combined experiment.

D-Optimality Efficiency: Using a D-optimal design for 6 factors at 5 levels was a sophisticated choice. It allowed us to test a large "search space" (over 15,000 possible combinations) in just 70 laps while keeping the correlation between factors low (<0.16).

Variable Isolation: The decision to switch to 1-lap stints to isolate fuel effects was a good way to estimate the benefit of less fuel on lap times.

Range Bracketing: Running longer stints on the Hard and Soft tires while keeping the Medium stint short was a quick way to learn about the degradation without wasting too much of the 120-lap budget.

D-Optimality Design (Phase 1): Utilizing one lap per setup is statistically efficient but risky. This approach makes the results highly sensitive to "noise" or random errors in a single lap. While increasing the number of laps per setup would improve reliability, it would significantly degrade the D-Efficiency (Dea), requiring more practice time than available.

Full Factorial Design (Phase 2): This is highly effective for a small number of configurations (e.g., the three viable tire compounds). However, because of the fuel experiments the resolution for the Medium tire remains limited. The 3-lap stint was too short to provide a robust statistical baseline compared to the 15-lap and 25-lap stints used for other compounds. This creates an unbalanced model that might be very accurate for long stints on the hard tire but inaccurate for stints on the soft and medium tire.

Task 2

For the race in England, the multi-armed bandit (MAB) method should be used exclusively. A very large parameter space must be explored using only a few draws from the underlying distributions and without applying more involved models, in order to find an optimal setup and strategy for the race. The limited number of practice laps (120) poses a significant challenge, as MAB problems usually involve many more pulls to identify the best-performing arm.

Due to the independence of the setup from the tyres (a setup that performs better on "soft" also consistently performs better on "hard"), forming joint tuples from the different setup and strategy parameters is not necessary. Accordingly, the problem is divided into two subproblems as follows:

1. Determining the best setup from a predefined set of possible setups.
2. Derivation of the best possible strategy from a predefined set of strategies.

MAB methods are typically used for problems in which exploration ('finding the best arm') and exploitation ('earning the rewards of the current best arm') must be balanced. However, the present problem is a classical 'ranking and selection' problem, as our sole aim is to find the best setup and strategy. While experimenting, there is no need to accrue high rewards. In the MAB community, this is referred to as a 'best arm identification' or 'pure exploration' problem [1, p. 2]. In this setting, simple algorithms can perform substantially better than classical MAB algorithms such as epsilon-greedy, UCB1 or Thompson sampling [3, p. 81]. Accordingly, algorithms that place a strong emphasis on exploration, or that exclusively explore, are well suited to the setting considered here.

1. Determination of the Setup

Due to the limited number of practice laps, a trade-off must be made between the number of bandit arms (i.e. setup-parameter tuples) and the number of practice laps allocated to each arm. If too many different tuples are explored, there will be too few practice laps allocated to each arm, resulting in highly noisy performance estimates that are not very informative. The setup tuples are selected in an attempt to achieve the broadest possible coverage of the parameter space, taking the track characteristics into account.

The track characteristics of England are as follows: cornering is low (23/100), grip is very high (79/100) and the track is high-altitude (77/100) and smooth (1/100).

The following findings from previous analyses, together with consideration of information from the "Analytics GP" online tool, suggest a tendency in the selection of setups for the bandit algorithm.

Aerodynamics

"Another point that we found was the importance of aerodynamic balance, indicating that downforce components must be tuned in concert rather than in isolation to maximize speed" (our analysis). This implies not using different values for the rear and front wing. "Car wing angles should be set higher in tracks with more corners" (Analytics GP). This corresponds with the traditional recommendation of a low-downforce setup for tracks with few corners.

Engine

"High engine output induces detrimental wheel spin in low-grip environments, while proving advantageous on high-traction tracks" (our analysis). Therefore, selecting higher engine output on high-traction tracks such as England might translate into improved acceleration. Nevertheless, "pushing the car at high altitude might be suboptimal" (Analytics GP). Accordingly, two substantially different engine settings are used here.

Break Balance

No insights have yet been obtained regarding brake balance, and the direction of the brake-balance effect remains unclear. Therefore, testing of different values is taking place here.

Differential

"High-cornering circuits require frequent shifting, thereby amplifying the utility of the differential. The adverse impact of larger differential settings mitigates—and eventually vanishes, as track cornering intensity increases" (our analysis). Accordingly, the differential for the England Grand Prix should be set rather conservatively. To simplify the search, it is kept constant at 100 for almost all tuples.

Suspension

In France, a trend towards very low suspension settings was observed. This could be consistent with the characteristics of the relatively rough track. Since the track in England is not rough at all, the following setup variants place a stronger focus on stiffer suspension settings. However, one variant includes an intentionally low value as a safeguard.

Arm	Rear Wing	Front Wing	Engine	Brake	Differential	Suspension
A1	250	250	300	250	250	250
A2	10	10	400	250	100	450
A3	10	10	100	250	100	450
A4	100	100	400	500	100	200
A5	100	100	100	50	100	200
A6	150	150	400	50	100	500
A7	300	300	350	500	100	50

The seven specified setup variants include a balanced baseline (A1), several low-downforce variants (A2–A6) with parameter choices guided by the above findings, and a medium-downforce setup (A7). These are intended to cover the parameter space as broadly as possible, while maintaining a focus on low downforce.

Successive Rejects Algorithm

The Successive Rejects (SR) algorithm is used to identify the best arm (i.e. the optimal setup) of the MAB problem, as described in Audibert, Bubeck, and Munos [1]. It is easy to implement, parameter-free and achieves an essentially optimal exponential rate for identifying the best arm (equivalently, simple regret/error probability), up to a logarithmic factor [1, p. 6]. In summary, in each phase of the algorithm, the worst-performing arm is eliminated (in our case, the arm with the largest average lap time). The last remaining arm is considered the best according to the algorithm. The number of pulls per arm in each phase is chosen so that the optimal convergence rate is achieved [1, p. 6]. For a detailed description of the algorithm, see Figure 3 in Audibert, Bubeck, and Munos [1, p. 6].

In order for the number of pulls per arm to depend exclusively on the setup, the stint length is set to 1, the fuel load to 120 and the tyre selection to ‘hard’ for all arms. The number of pulls per arm across the six phases is as follows:

```
## Pulls phase 1: 4
## Pulls phase 2: 1
## Pulls phase 3: 1
## Pulls phase 4: 1
## Pulls phase 5: 2
## Pulls phase 6: 4
```

```
## Total pulls:  57
```

The iterative execution of the algorithm via simulation in Analytics GP yields the following:

```
## Phase 1 - eliminated arm: A3
## Phase 2 - eliminated arm: A2
## Phase 3 - eliminated arm: A4
## Phase 4 - eliminated arm: A7
## Phase 5 - eliminated arm: A5
## Phase 6 - eliminated arm: A6
## Selected arm:  A1  - Lap-time:  109.0824
```

2. Determination of the Strategy

After the A1 setup was selected as the best of the seven candidates, 63 practice laps remain available. Three additional laps are required to determine the linear constants for fuel decrease and tyre wear for each compound (extrasoft, soft and medium). Based on these constants, suitable strategies for the England Grand Prix can then be specified.

The estimated per-lap fuel decrease is z = 3.23. The compounds degrade at the following rates per lap: w_extrasoft = 12.67, w_soft = 5.51, w_medium = 4.46 and w_hard = 3.67. As the maximum fuel load is 120 and z is smaller than all the compound degradation constants, fuel is never the limiting factor in determining the length of a stint.

Based on this information and the assumption of linearly decreasing fuel and compound values (cf. the French Grand Prix), the arms for the strategy bandit have been specified. A key insight from previous races is that driving with low fuel values is highly advantageous. To facilitate selection, only tyre compound and the number of stops are permitted as variables. Each strategy uses one compound and allocates laps as evenly as possible across the 63 laps of the race. Each stint starts with the minimum feasible refuelling amount. The following are selected from the set of feasible strategies implied by this:

Arm	Compound	w_comp	b	L1	L2	L3	L4	L5
A1	hard	3.67	2	21	21	21	-	-
A2	medium	4.46	3	16	16	16	15	-
A3	soft	5.51	3	16	16	16	15	-
A4	soft	5.51	4	13	13	13	12	12

The next step is to specify both the multi-armed bandit algorithm and the objective function by which the arms are evaluated. The main issue here is obtaining an estimator for the lap time of a stint that is as unbiased as possible. As no other models, such as regression, can be used, our approach is as follows: As in the search for the setup, this is a best-arm identification problem. However, here the SR algorithm is not used; instead, the extremely simple uniform allocation strategy is employed [2, p. 9 and following]. Under this approach, the number of pulls is distributed equally across arms, independently of the observed rewards. Since obtaining the best possible estimate of the stint time (and thus the race time of the current strategy) is crucial, each arm (i.e. each of the four strategies) is pulled only once. Furthermore, it is important that the stint time estimates are as comparable as possible and that estimation induces as little bias as possible. For two stops, this implies 21 laps per stint, for three stops 16 laps, and so on. Since only 60 laps remain, the following stint length scheme is used when pulling the arms.

Arm	A1	A2	A3	A4
Compound	hard	medium	soft	soft
Target stint length $L(b)$	21	16	16	13
Simulated stint length	19	15	15	11
Minimal fuel load $F(z, b)$	68	52	52	42

The minimal fuel load for the simulation is calculated as follows: Let b denote the number of pitstops and $L(b)$ be the target stint length. For per-lap fuel consumption $z = 3.23$, the minimal fuel load is

$$F(z, b) \;=\; \lceil L(b)\, z \rceil \;=\; \left\lceil \left\lceil \frac{63}{b+1} \right\rceil z \right\rceil .$$

As a side note, it is worth mentioning that an additional lap can be driven for each stint that is not the first, since the lap in which a stop is made does not affect the fuel load or tyre condition. Therefore, the resulting strategies for all pit stops can decrease the fuel load by 3. The average lap times from these stints are ultimately used as the basis for the bandit's reward. The objective function is then an estimate of the total race time, to which 30 seconds per pit stop are added, as noted in Analytics GP.

$$\widehat{\text{RaceTime}}^{\text{sim}}_{A_i}(b) \;=\; 63\,\bar{y}_{\text{sim}} \;+\; 30\,b\,.$$

The recommendation for the output of the best arm remains the arithmetic mean. In this trivial case, where there is only one pull, this consists of one observed value.

```
## [1] "Best Arm: A3. Racetime: 6434.175339596"
```

Conclusion

The MAB algorithms clearly identified the best arm in both subtasks. However, in hindsight, the SR algorithm is not the best choice for finding the optimal setup since it uses a relatively large number of pulls per arm. The inherent variance in lap times for the same setup was not large enough to justify so many draws. Therefore, using a different MAB algorithm for identifying the best arm (e.g. uniform allocation or UCB1) would have enabled a substantially larger parameter space to be explored (i.e. more arms or setups). This conjecture was also confirmed by the race results of the England GP. Our car setup performance was clearly worse than that of the other teams. However, the approach to identifying the best strategy proved to be a good choice.

In general, when the budget for the practice is limited and the parameter space is extensive, a bandit approach seems to be a better option than a reduced experimental design, as it allows a few strong candidates to be identified quickly. The algorithm can dedicate more laps to promising setups to average out the noise, while briefly sampling poor ones. Bandits are also useful when the main objective is to select a near-optimal setup by the end of the practice period rather than learning the precise effects of the parameters.

A bandit approach is particularly appealing if there are significant performance differences between the options because it can exploit early evidence and focus on the most important setups. By contrast, I would opt for a fixed experimental design if the aim is to derive inferences, e.g. to estimate the impact of setup variables such as brake balance or engine tuning.

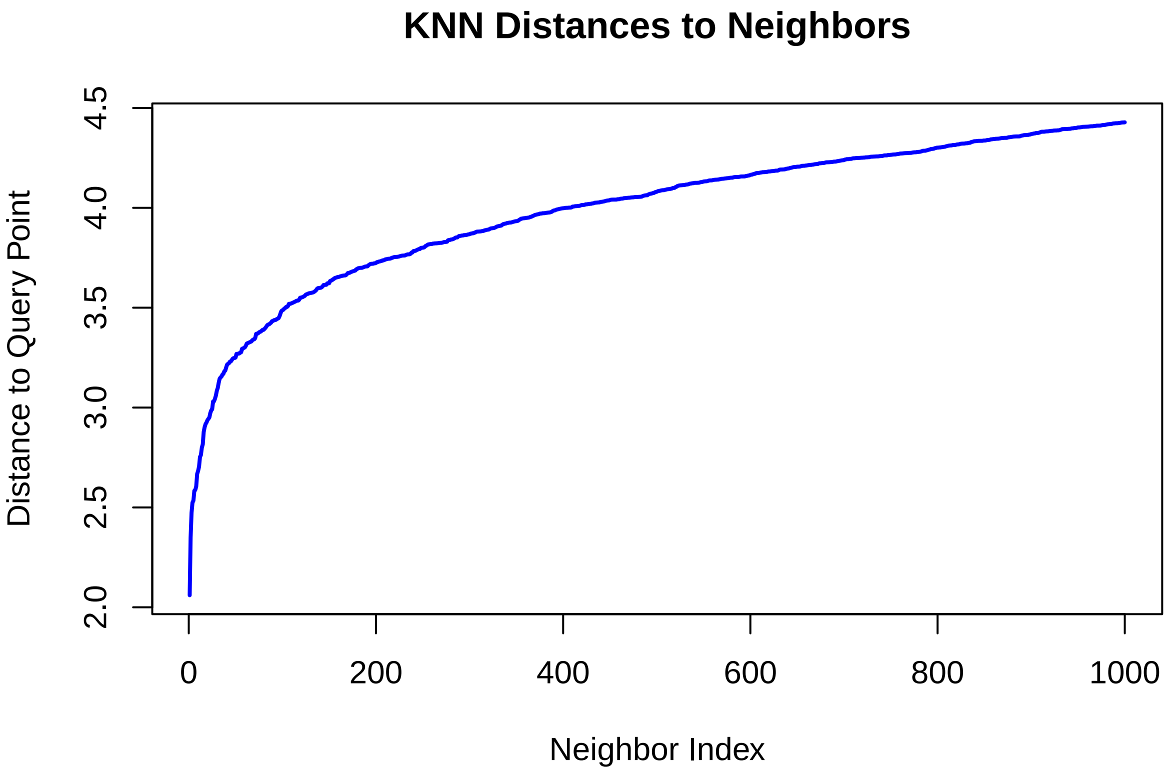
Task 3

1. Setup

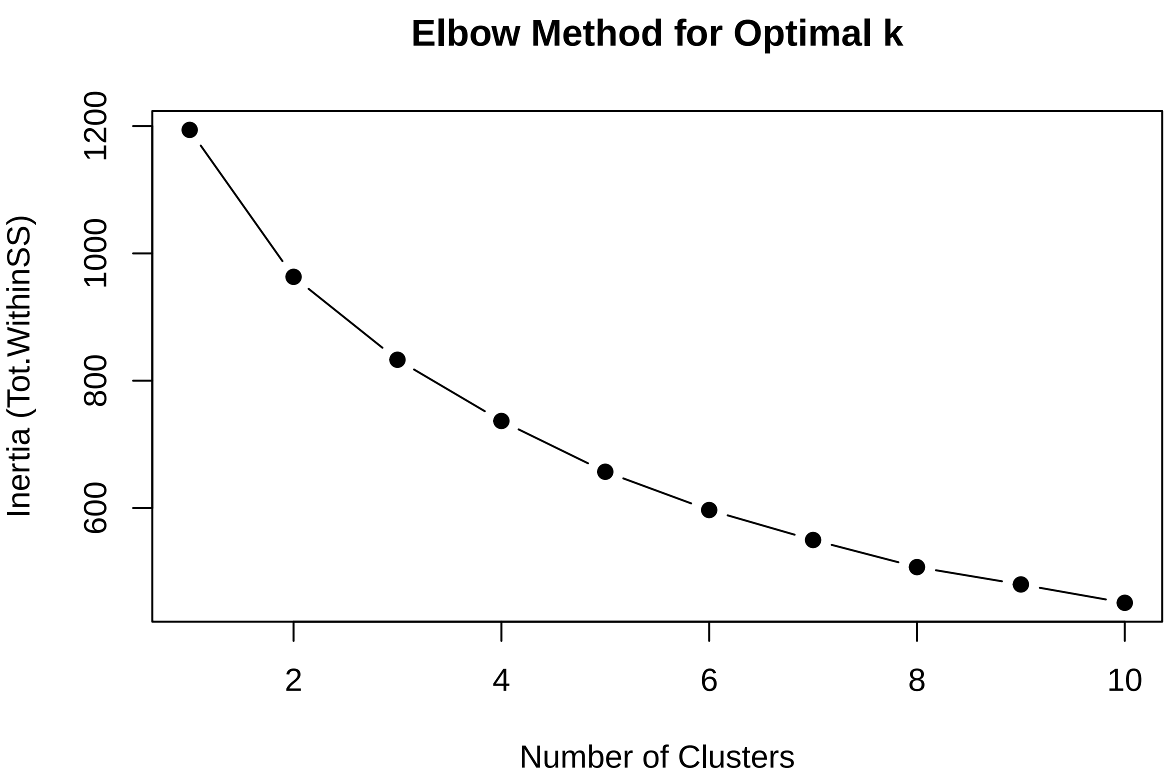
Data-Driven region search

For the Belgium race, the car setup is optimized via a data-driven approach to identify a promising region and then sample the space using information from prior tasks.

The data-driven methodology employs a K-Nearest Neighbors (KNN) algorithm to retrieve samples comparable to the Belgium circuit's conditions. These instances are subsequently clustered to identify optimal vehicle setups and define the boundaries of the search space. Initially, the parameter K is determined by analyzing the distance metric of the first 1000 points (10% of the dataset). K=200 is selected, as this value corresponds to the point where the curve flattens.

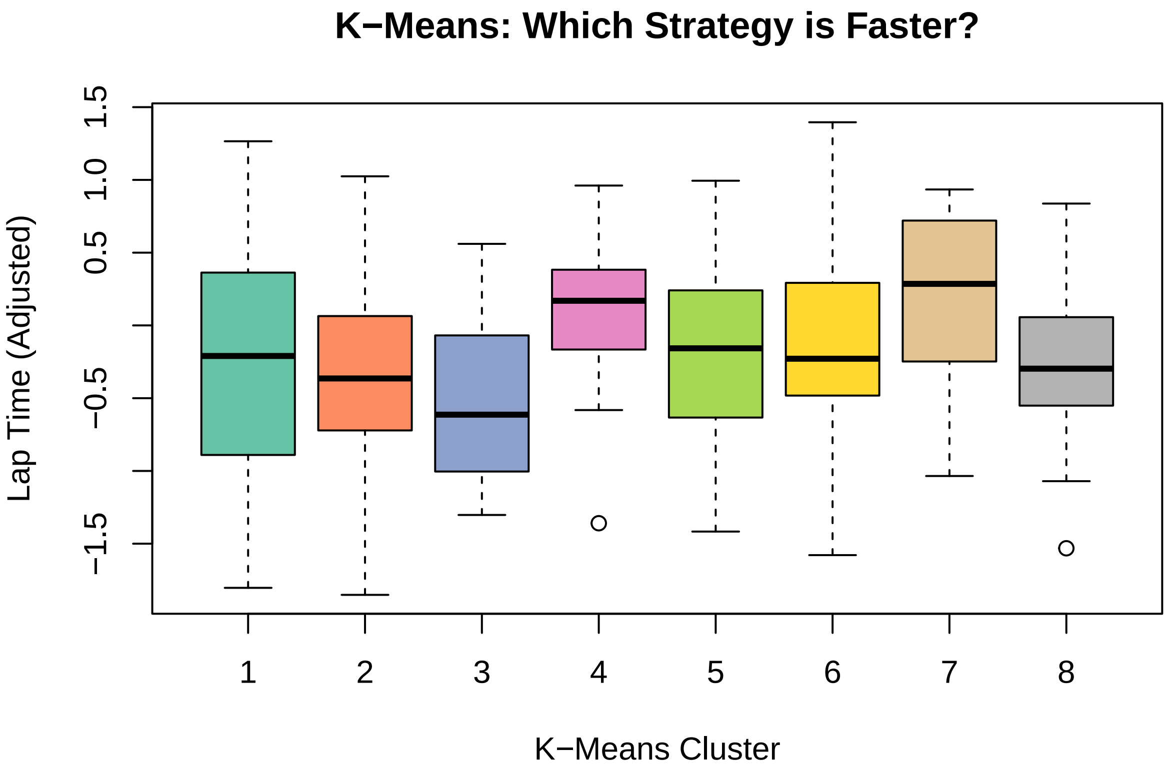


Following the identification of the 200 nearest data points, both K-Means and Agglomerative Clustering are used to identify a more comprehensive and robust data story. Promising clusters are subsequently determined by evaluating the mean adjusted lap time within each group.For K-Means, the optimal parameter K is selected by iterating through a range of 1 to 10 and analyzing both the Elbow method and the Silhouette score.



$K = 8$ is choosen for the K-means as there the silhouette score starts to increase again and we do not overfit the number of clusters for a sample size of 200.

```
##   Front Wing Rear Wing Brake Balance Suspension    Engine Differential
## 1  417.41667  438.3056      90.08333   37.22222  58.80556    136.75000
## 2  335.67742  457.5161     321.54839   27.29032  16.90323    146.32258
## 3   85.73333  200.6000     127.20000   79.53333  50.73333     47.33333
## 4  276.89286  399.8214     272.96429  181.25000  40.00000    238.28571
## 5  185.30000  400.9333     127.73333   11.83333  31.30000     46.40000
## 6  442.40000  469.1333     334.26667   37.86667 108.80000    335.40000
## 7  212.87500  420.5000     214.87500   49.81250 206.81250    146.93750
## 8  316.55172  458.4828     159.27586  153.89655  18.03448     54.20690
##   Cluster_Size Mean_Adjusted_Lap_Time
## 1           36            -0.21708277
## 2           31            -0.31665884
## 3           15            -0.47812990
## 4           28             0.09834299
## 5           30            -0.21876011
## 6           15            -0.11363261
## 7           16             0.24206572
## 8           29            -0.22673828
```




```
##
## Best K-Means Cluster is #3. Use this for further optimization.
```

```
##
## Bounds for optimization (based on best K-Means cluster):
```

```
## Front Wing: 1.00 - 194.40
## Rear Wing: 130.20 - 267.70
## Brake Balance: 16.40 - 232.70
## Suspension: 1.00 - 165.90
## Engine: 1.00 - 138.00
## Differential: 1.00 - 135.40
```

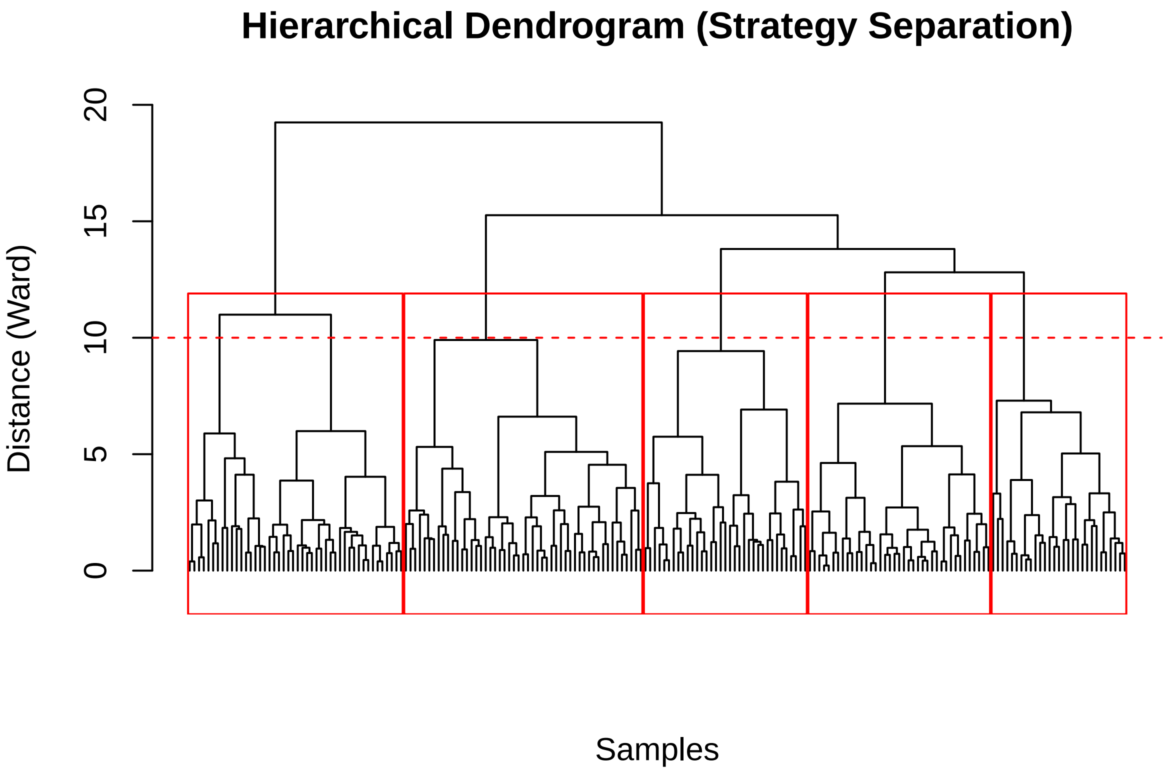
The cluster summary, showing the mean of each car-feature parameter and the average adjusted lap time, identifies clusters 2 and 6 as the most promising candidates. Cluster 6 exhibits a slightly superior lap time and a larger sample size, suggesting a more robust estimate. Both clusters display very low values for Engine and Suspension, while Brake Balance and differential fall within the middle range. However, the wing parameters present a contradiction: while cluster 6 indicates high values, particularly for the Rear Wing, cluster 2 suggests mid-range values for the Front Wing. This discrepancy highlights the necessity for an additional clustering algorithm to extract a coherent story and identify the final region.

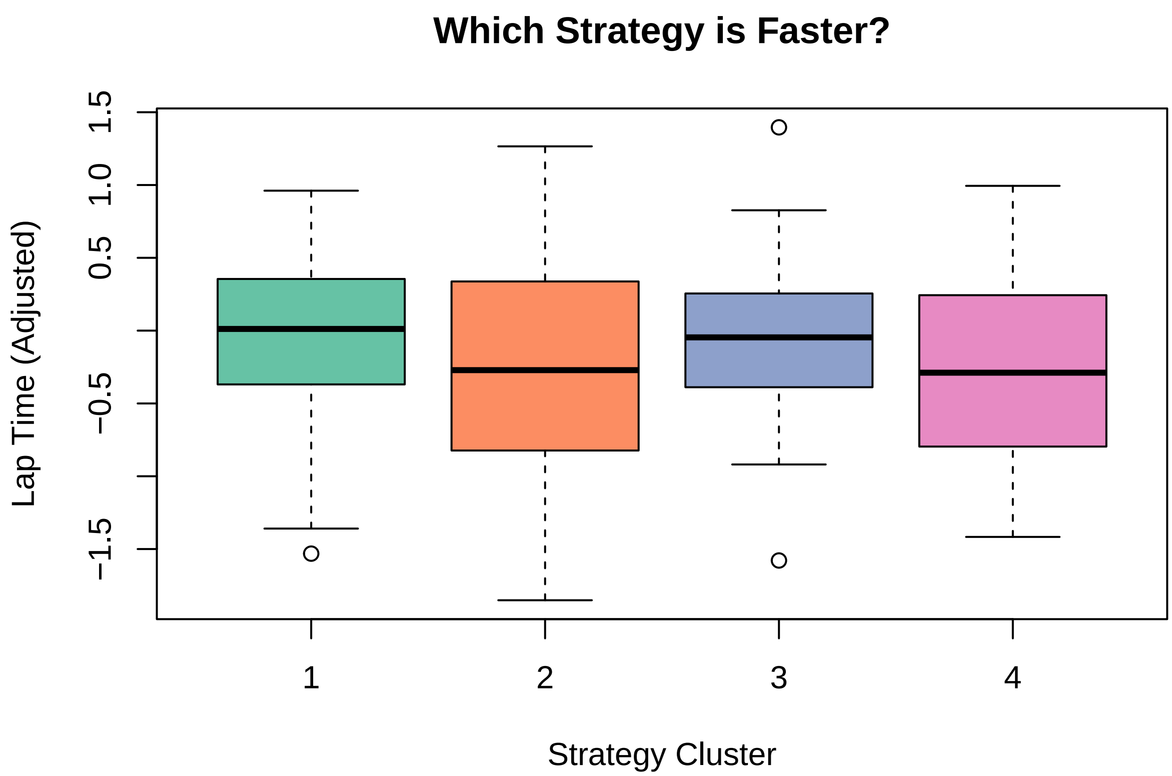
```
print(cluster_summary)
```

```
##   Front Wing Rear Wing Brake Balance Suspension    Engine Differential
## 1  417.41667  438.3056      90.08333   37.22222  58.80556    136.75000
## 2  335.67742  457.5161     321.54839   27.29032  16.90323    146.32258
## 3   85.73333  200.6000     127.20000   79.53333  50.73333     47.33333
## 4  276.89286  399.8214     272.96429  181.25000  40.00000    238.28571
## 5  185.30000  400.9333     127.73333   11.83333  31.30000     46.40000
## 6  442.40000  469.1333     334.26667   37.86667 108.80000    335.40000
## 7  212.87500  420.5000     214.87500   49.81250 206.81250    146.93750
## 8  316.55172  458.4828     159.27586  153.89655  18.03448     54.20690
```

```
##   Cluster_Size Mean_Adjusted_Lap_Time
## 1           36            -0.21708277
## 2           31            -0.31665884
## 3           15            -0.47812990
## 4           28             0.09834299
## 5           30            -0.21876011
## 6           15            -0.11363261
## 7           16             0.24206572
## 8           29            -0.22673828
```

To determine the number of clusters for Agglomerative Clustering, a dendrogram is utilized. Visual inspection suggests a range of 4 to 5 clusters; ultimately, 4 is selected to maintain more robust clusters compared to the K-means.






```
## [1] "Hierarchical Cluster Summary:"

##   Cluster Front Wing Rear Wing Brake Balance Suspension   Engine Differential
## 1       1   292.1569  418.2549      206.3137  151.90196 18.56863    116.88235
## 2       2   343.0588  459.1176      169.5147   29.33824 80.23529    134.92647
## 3       3   376.2000  456.4286      346.6000   85.88571 81.22857    264.68571
## 4       4   170.8261  320.1957      119.2391   41.06522 36.50000     60.58696
##   Cluster_Size Mean_Adjusted_Lap_Time
## 1           51            -0.07415756
## 2           68            -0.21334932
## 3           35            -0.04364497
## 4           46            -0.28700078

## Best cluster is #4. Use this for further optimization.

##
## Bounds for optimization (based on best cluster):

## Front Wing: 5.00 - 349.25
## Rear Wing: 140.75 - 480.50
## Brake Balance: 1.00 - 240.00
## Suspension: 1.00 - 159.75
## Engine: 1.00 - 132.75
## Differential: 1.00 - 179.00
```

Clusters 2 and 4 emerge as the most promising candidates. They are consistent with the hypothesis regarding Differential, Suspension, Brake Balance, and Engine. However, the wing parameters again show the largest discrepancy between the two groups; notably, the larger and thus more robust cluster suggests higher values for these settings.

```
print(hc_summary)
```

```
##   Cluster Front Wing Rear Wing Brake Balance Suspension   Engine Differential
```

```
## 1       1   292.1569  418.2549      206.3137  151.90196 18.56863    116.88235
## 2       2   343.0588  459.1176      169.5147   29.33824 80.23529    134.92647
## 3       3   376.2000  456.4286      346.6000   85.88571 81.22857    264.68571
## 4       4   170.8261  320.1957      119.2391   41.06522 36.50000     60.58696
##   Cluster_Size Mean_Adjusted_Lap_Time
## 1           51            -0.07415756
## 2           68            -0.21334932
## 3           35            -0.04364497
## 4           46            -0.28700078
```

Region Sampling

Based on the prior cluster analysis, the sampling space is defined as follows:

Feature	Min	Max
Front Wing	170	500
Rear Wing	360	500
Brake Balance	40	480
Suspension	1	80
Engine	1	110
Differential	1	300

Notably, the Front Wing exhibited significant outliers in the lower range, while Brake Balance consistently appeared in the mid-range, implying considerable uncertainty. Furthermore, as the Differential displayed isolated higher values, an upper bound of 300 has been incorporated. Overall, these limits are established conservatively, as the subsequent sampling methodology facilitates the evaluation of a greater number of configurations compared to previous tasks.

Testing Confiurations

Knowledge from prior tasks is utilized to evaluate different configurations. Previous analysis revealed large time discrepancies between setups, enabling the rapid elimination of unpromising candidates. Therefore, 30 configurations are sampled, with the optimal setup identified within a total budget of 80 single-lap stints. Strategy parameters remain fixed at 100l fuel and Soft tires throughout.

Initially, the 30 configurations are simulated for two single-lap stints each. After that, the bottom 15 are eliminated, while the top half proceeds to a third lap. Finally, the best five candidates undergo a fourth lap, after which the optimal configuration is selected.

To generate the 30 configurations, Latin Hypercube Sampling (LHS) is employed within the identified bounds. LHS is a Monte Carlo simulation technique designed to cover sample spaces more efficiently than independent uniform sampling.

```r
print(head(setup_df))
```

```
##   Setup_ID Front Wing Rear Wing Brake Balance Suspension Engine Differential
## 1        1        477       441           175         72     50          179
## 2        2        282       480           461          9     47           88
## 3        3        438       482           435         75    109           62
## 4        4        311       399           477         46     16          106
## 5        5        228       380           136          2    100           93
## 6        6        299       476            71         69     34          119
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

##   Round  Track Qualifying Stint Lap Rear Wing Front Wing Engine Brake
## 1     1 France          1     0   1       500        370    130   250
## 2     1 France          0     1   1       500        370     10   130
## 3     1 France          0     2   1        10        500    130   250
## 4     1 France          0     3   1       130        500    500   250
## 5     1 France          0     4   1       130        130    500   370
## 6     1 France          0     5   1       500         10    250   500
##   Differential Suspension     Fuel Tyre Remaining Tyre Choice Lap Time
## 1          250         10 10.00000      100.00000  Extra Soft 63.90551
## 2           10         10 47.75181       96.87555        Hard 68.19942
## 3           10         10 47.75181       96.87555        Hard 70.55448
## 4           10         10 47.75181       96.87555        Hard 71.12798
## 5           10         10 47.75181       96.87555        Hard 71.50342
## 6           10         10 47.75181       96.87555        Hard 69.30716
##   Lap Distance Cornering Inclines Camber Grip Altitude Roughness Width
## 1          3.3        98       73     13   18       15        74     8
## 2          3.3        98       73     13   18       15        74     8
## 3          3.3        98       73     13   18       15        74     8
## 4          3.3        98       73     13   18       15        74     8
## 5          3.3        98       73     13   18       15        74     8
## 6          3.3        98       73     13   18       15        74     8
##   Temperature Humidity Wind (Avg. Speed) Wind (Gusts) Air Density Air Pressure
## 1          39       74                48           77          14           41
## 2          39       74                48           77          14           41
## 3          39       74                48           77          14           41
## 4          39       74                48           77          14           41
## 5          39       74                48           77          14           41
## 6          39       74                48           77          14           41

## `summarise()` has grouped output by 'Rear Wing', 'Front Wing', 'Engine',
## 'Brake', 'Differential'. You can override using the `.groups` argument.

## Adding missing grouping variables: `Rear Wing`, `Front Wing`, `Engine`, `Brake
## Balance`, `Differential`

## # A tibble: 15 x 7
## # Groups:   Rear Wing, Front Wing, Engine, Brake Balance, Differential [15]
##    `Rear Wing` `Front Wing` Engine `Brake Balance` Differential Setup_ID
##          <dbl>        <dbl>  <dbl>           <dbl>        <dbl>    <int>
##  1         441          477     50             175          179        1
##  2         460          480     15             306          168       12
##  3         453          493     78             328          145       24
##  4         414          403     56             280          155       28
##  5         476          299     34              71          119        6
##  6         409          446     89             258          204        9
##  7         494          355     29             350           57       15
##  8         445          384     23             239          244       11
##  9         406          392     54             168          281       23
```

```
## 10          373          429    41             148           293        18
## 11          489          415    35             406            34         7
## 12          498          317    60             446           211        20
## 13          467          373     5             337           269        30
## 14          482          438   109             435            62         3
## 15          391          367    83             364           274        21
## # i 1 more variable: Mean_Lap_Time <dbl>
```

The results from the two-stint simulations indicate that high-downforce setups perform best. For Engine and Suspension, very low values appear superior, while Differential settings also exhibit consistency, ranging from 145 to 204. However, the Brake Balance parameter demonstrates the necessity of retaining a broad search space, as indicated by the large interval observed.

```r
print(head(top_15))
```

```
## # A tibble: 6 x 8
## # Groups:   Rear Wing, Front Wing, Engine, Brake Balance, Differential [6]
##   `Rear Wing` `Front Wing` Engine `Brake Balance` Differential Suspension
##         <dbl>        <dbl>  <dbl>           <dbl>        <dbl>      <dbl>
## 1         441          477     50             175          179         72
## 2         460          480     15             306          168         23
## 3         453          493     78             328          145         16
## 4         414          403     56             280          155         53
## 5         476          299     34              71          119         69
## 6         409          446     89             258          204         36
## # i 2 more variables: Mean_Lap_Time <dbl>, Setup_ID <int>
```

```
## `summarise()` has grouped output by 'Rear Wing', 'Front Wing', 'Engine',
## 'Brake', 'Differential'. You can override using the `.groups` argument.
```

The subsequent round supports the findings from the first two. The top four setup IDs remain consistent, with only a minor change in ranking between setups 1 and 12.

```r
print(top_5 %>% select(Setup_ID, Mean_Lap_Time))
```

```
## Adding missing grouping variables: `Rear Wing`, `Front Wing`, `Engine`, `Brake
## Balance`, `Differential`
```

```
## # A tibble: 5 x 7
## # Groups:   Rear Wing, Front Wing, Engine, Brake Balance, Differential [5]
##   `Rear Wing` `Front Wing` Engine `Brake Balance` Differential Setup_ID
##         <dbl>        <dbl>  <dbl>           <dbl>        <dbl>    <int>
## 1         460          480     15             306          168       12
## 2         441          477     50             175          179        1
## 3         453          493     78             328          145       24
## 4         414          403     56             280          155       28
## 5         445          384     23             239          244       11
## # i 1 more variable: Mean_Lap_Time <dbl>
```

```
## `summarise()` has grouped output by 'Rear Wing', 'Front Wing', 'Engine',
## 'Brake', 'Differential'. You can override using the `.groups` argument.
```

The final round again supports the trends observed in the earlier stages. The ranking of the top five setups remains unchanged, establishing Setup 12 as the optimal configuration.

```r
print(as.data.frame(top_5 %>% select(Setup_ID, Mean_Lap_Time)))
```

```
## Adding missing grouping variables: `Rear Wing`, `Front Wing`, `Engine`, `Brake
## Balance`, `Differential`
```

```
##   Rear Wing Front Wing Engine Brake Balance Differential Setup_ID Mean_Lap_Time
## 1       460        480     15           306          168       12      100.7602
## 2       453        493     78           328          145       24      101.1993
## 3       441        477     50           175          179        1      101.2353
## 4       414        403     56           280          155       28      101.4497
## 5       445        384     23           239          244       11      101.9834
```

2. Strategy

To optimize the strategy, the effects of tire degradation and fuel consumption are estimated to find an optimal tradeoff. Therefore we drive 5 stints with 8 laps each, using the remaining 40 laps of the 120 practice laps. Although decay rates are known to be fixed from prior tasks the actual impact on lap time is subject to noise. Consequently, this effect must be estimated in the forthcoming section.

Type	Decay rates per lap	MAX Laps	MIN Stints
Fuel	3.3704	35.60	2
Extra Soft	5.9822	16.72	4
Soft	3.9235	25.49	3
Medium	3.4209	29.23	3
Hard	3.1874	31.37	3

Since the fuel content can be varied independently of the tire compound, the analysis begins by estimating the fuel effect. To achieve this, four single-lap stints are executed with a 120L load, and four with 4L, the minimum permissible starting amount. With the parameters fixed to Setup 12, the average times for the 120L and 4L runs are computed to derive the time penalty per liter and per lap.

$P\_liter = \frac{\bar{T}_{120} - \bar{T}_4}{120-4}$ $C\_lap = \frac{\bar{T}_{120} - \bar{T}_4}{35.6}$

Parameter	Value
AVG Max Fuel	102.9800
AVG Min Fuel	93.5634
Fuel Diff (s)	9.4200
Fuel Penalty per Lap (s)	0.2646
Fuel Penalty per Liter (s)	0.0812

Subsequently, an 8-lap stint is performed with each of the four tire compounds to estimate the impact of tire degradation. By compensating for the known fuel effect, the specific influence of tire wear is isolated. However, given the noise within the data, no statistically significant time penalty was observed for tire degradation, a finding that drastically simplifies strategy optimization. Finally, a theoretical base pace is calculated for each compound, representing the estimated lap time at zero fuel load.

$$\begin{aligned}
&\text{1. Fuel Adjustment:} && T_{adj}^{(i)} = T_{raw}^{(i)} - \left(F_{rem}^{(i)} \cdot P_{liter}\right) \\
&\text{2. Tire Decay (Slope):} && D_{tire} = \frac{\sum_{i=1}^{N}(i-\bar{i})(T_{adj}^{(i)} - \bar{T}_{adj})}{\sum_{i=1}^{N}(i-\bar{i})^2} \\
&\text{3. Base Pace:} && P_{base} = \frac{1}{N}\sum_{i=1}^{N} T_{adj}^{(i)}
\end{aligned}$$

Tire	Seconds lost per lap	Basepace Estimation
Extra Soft	-0.1044	92.59
Soft	0.0323	93.41
Medium	-0.0755	93.43
Hard	0.1012	93.64

The total race time T_{total} is then calculated by summing the individual lap times across all stints. Since the fuel penalty increases linearly with each lap driven (or rather, the fuel load decreases linearly), the lap time for the i-th lap in a stint k is defined as the base pace of the tire plus the fuel penalty accumulated up to that lap.

Iterative Formulation (Sum of Laps):

$$T_{total} = \sum_{k=1}^{N_{stints}} \sum_{i=0}^{L_k-1} \left(P_{base}(\text{tire}_k) + i \cdot F_{penalty}\right) + N_{stints} * t$$

Where:

- N_{stints}: Total number of stints in the strategy.
- L_k: Length of the k-th stint (number of laps).
- $P_{base}(\text{tire}_k)$: Theoretical base pace of the compound used in stint k (at Lap 0 fuel load).
- $F_{penalty}$: Time cost per lap added due to fuel weight (or effectively, the time delta relative to the base pace).
- i: Lap index within the current stint $(0, 1, \ldots, L_k - 1)$. *t: stint penalyt (30s)

Two distinct three-stint strategies were evaluated to determine the optimal approach. The first configuration, utilizing exclusively Soft tires (3x Soft), resulted in a total race time of 6118.22 seconds. In contrast, the mixed compound strategy (Medium, Soft, and Extra Soft) achieved a superior time of 6111.00 seconds.

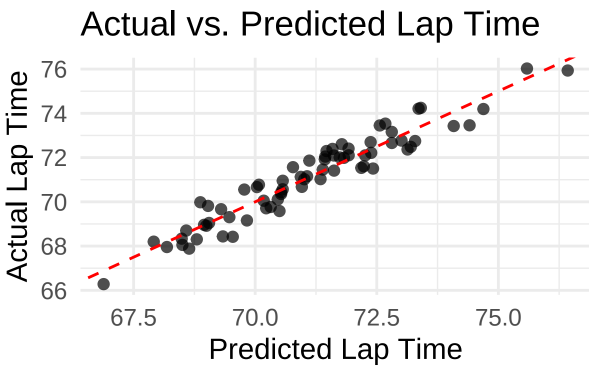
Conclusion

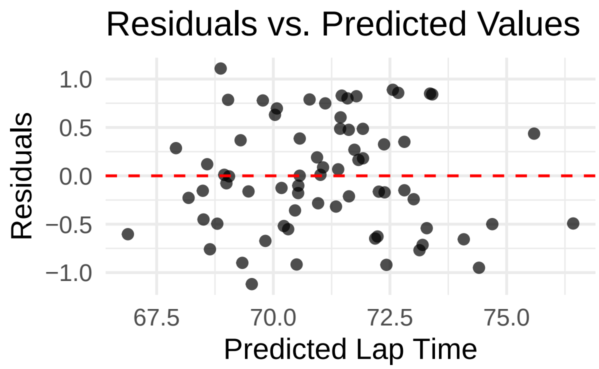
The qualifying results provide insight into the setup's relative competitiveness. While the configuration showed improved performance compared to the previous two races, it ultimately ranked last, highlighting significant residual potential for optimization. This suggests that future iterations would benefit from an expanded sampling size and a more aggressive selection process, particularly as the top four configurations remained static across evaluation rounds. Furthermore, given the established low complexity of the race strategy, computational resources could be reallocated to increase the number of evaluation rounds per setup.

Finally, the data supports the hypothesis of boundary solutions. The preference for maximum values in Front and Rear Wing combined with minimal values for Suspension and Engine suggests that the global optimum likely lies at the parameter bounds (e.g., 500 and 1). Regarding race strategy, while the selected approach proved viable, retrospective analysis indicates it was suboptimal. Replacing a Medium stint with a Soft compound would likely have yielded superior lap times, representing a missed optimization opportunity in this instance.

Task 4

Insight 1






```
## R2 = 0.9158792 adj_R2 = 0.8981696
```

```
## RMSE: 0.5652162
```

```
## (Informal) Signal-to-Noise Ratio (Model Var / Residual Var): 10.88767
```

Efficiency vs. Resolution in Single-Shot Experimental Design

To maximize the search space within the 120-lap limit, we utilized a "Single-Shot" D-optimal design, testing 70 unique configurations for only one lap each. Analysis of the Phase 1 data reveals a Signal-to-Noise Ratio of 10.89 and an R-squared of 0.916. The design successfully identified the dominant main effects and curvature where it exists. However, despite high R^2, the RMSE of 0.57 seconds indicates limited precision for ranking near-optimal setups.

The main lesson is that efficiency comes at the cost of resolution. The Single-Shot approach proved excellent for "Coarse Optimization"—effectively mapping the performance landscape and identifying the general "sweet spot" (e.g., determining that high Differential and low Brake Balance were superior).

However, it failed at "Fine Tuning." With a noise floor of ~0.6 seconds, the design lacked the statistical power to distinguish between "good" and "optimal" setups, which often differ by only 0.1–0.2 seconds. In future reduced designs, we would adopt a funnel approach: using Single-Shot testing to discard the bottom 80% of configurations, followed by multi-lap confirmation runs (3-5 laps) on the top candidates to penetrate the 0.6s noise floor and pinpoint the true optimum.

References

[1] Jean-Yves Audibert, Sébastien Bubeck, and Remi Munos. *Best Arm Identification in Multi-Armed Bandits*. Nov. 2010.

[2] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. *Pure Exploration for Multi-Armed Bandit Problems*. 2010. arXiv: 0802.2655 `[math.ST]`. URL: https://arxiv.org/abs/0802.2655.

[3] Daniel Russo et al. *A Tutorial on Thompson Sampling*. 2020. arXiv: 1707.02038 `[cs.LG]`. URL: https://arxiv.org/abs/1707.02038.