Universiteit Leiden
The Netherlands

Bachelor Informatica

A data-driven approach to determine personalised stage suitability

Domen van Soest s2962632

Supervisors:
Arno Knobbe & Teun van Erp (Tudor)

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

06/01/2025

Abstract

This thesis explores professional road cycling through the lens of data science, addressing tactical questions about how well specific race stages align with the physiological profile of a key rider. In professional cycling, athletes are typically categorized by their strengths across different terrains—for example, climbers excel on mountain stages, while sprinters dominate flatter routes. The focus of this study centers on a particular rider from the Tudor Pro Cycling Team who possesses an intermediate profile: someone especially competitive on semi-mountainous stages. Unlike pure sprinters, this rider can endure sustained climbs, and unlike pure climbers, he can often outpace them in finishing sprints.

The central tactical question is whether the characteristics of an upcoming race suit this rider's abilities. If the course does not play to his strengths, teammates might choose to conserve energy for future opportunities. Conversely, if the profile is favorable, the team can rally around him to maximize the chance of a stage win.

To address this, the study develops a data-driven analysis pipeline using detailed sensor data from the rider's bike and wearables. The pipeline first calculates time gaps to the front of the race, followed by spatial feature engineering. It then applies machine learning techniques to predict speed and power output throughout a given stage. The final component is a drop-off prediction model that estimates the likelihood of the rider falling behind during an upcoming race based on course characteristics and physiological data.

The findings show that it is possible to forecast specific segments of elevated terrain where the rider is likely to struggle or lose contact with the lead group. These insights offer practical value for shaping team tactics, race-day planning, and rider selection. Overall, this research contributes to the growing role of data-driven decision-making in professional cycling.

Contents

1 Introduction

Professional road cycling is one of, if not the most well-known, sports on a bike. The sport spans all continents, where cyclists must overcome steep ascents, tricky descents, and difficult environmental conditions. This combination of physiological demands, strategic complexity, and team coordination creates a sporting discipline offering opportunities for race prediction and analysis.
In this project, we focus on one specific topic of professional cycling: the moment when a specific rider cannot keep up with the pace of the fastest rider.

Professional cycling is governed by the Union Cycliste Internationale (UCI), which regulates races on the international podium. The UCI structures each race into hierarchical tiers, where World Tour (WT) races are the highest tier possible, and the Pro level, where Tudor is placed, is one tier below WT. World Tour races feature the most prestigious races each year, such as the Tour de France, the Giro, and the Vuelta, also known as the Grand Tours. The competitive format includes both one-day classics and multi-stage races, which can be again subdivided into five categories, including individual time trials (ITT), team time trials (TTT), flat stages, semi-mountainous stages, and mountainous stages [ELS21].

To achieve the best results in these race formats, teams have a squad of riders with different qualities. Generally the riders can be divided into two body types: aerobic and anaerobic. Sprinters are characterised by an anaerobic physiological profile featuring type II muscle fiber, which enables high-power output over a short duration [PLSH+11]. In contrast, climbers have an aerobic body type, which features type I muscle fiber. The aerobic body type enables the climbers to produce submaximal power outputs for longer time periods [TBSS93]. However, what is in between the typical climber and sprinter?

1.1 Strategy in Professional Cycling

The outcomes of professional races are not only determined by individual capabilities, but a big factor has to do with the team's strategic approach. The term ”drafting” needs to be explained to understand the basics of tactics. Drafting happens when a rider is riding behind other competitor (s) to reduce the wind resistance, which could save up to 63% of their energy when having the right posture on their bikes [vB23]. When a rider loses the wheel of the rider in front of him, the benefit of drafting is lost. This forces the rider to output significantly more power to maintain the same speed, leading to a critical moment also known as the ”drop-off moment.”

With this knowledge, the competitors mostly ride together in a group, known as the peloton, where they have the benefit of drafting to conserve energy for potentially decisive moments in the race. With this in mind, teams have a hierarchical structure to support the rider whom they think is most likely to win the race. This rider, the leader, is protected by his supporting teammates, known as domestiques, who perform actions like pace setting and drafting. The goal of these actions is to save the energy of the leader to increase the possibility of him winning the race.

During the race, not everyone stays in the peloton; Breakaway attempts always happen and involve small groups of competitors who try to gain time and distance by riding away. However, when trying to escape the peloton, it comes with sacrifices, because the aerodynamic advantages are thrown away, so riders should be careful not to exhaust themselves too early and get caught by the peloton again before the finish line. These elements form a complex system where race outcomes reflect the actions of the individual rider, but also the tactical positioning and the team coordination during the race.

1.2 Data science in Professional Cycling

Due to technology evolving rapidly, the way people look at the sporting field has completely changed. The same happened with professional cycling, where technological advancements are revolutionizing the sport in many ways. Training optimization, talent recognition, and race strategies—all of these aspects are becoming more dependent on computers and their calculations. The whole process of getting the data is made possible by computers on the bike or on the cyclist. These computers vary from measuring heart rate and GPS to obtaining power output and speed. The measurements create multi-variate-timeseries and provide valuable insights into the rider's performance, which allows predictive models to be made. However, is it possible to add new features to improve the model's accuracy? This leads to the first subquestion:

Q1: How can the original multi-variate timeseries be enriched with features that improve the final model?

Data collection forms the foundation for building predictive models. These models enable teams and coaches to optimize training programs [KNH21], and forecast race results [KDSVL20b], providing valuable insights for performance analysis and strategic decision-making. By using historical data and altitude profiles, it is possible to develop machine learning models that predict key metrics. This leads to the second subquestion:

Q2: Given a future altitude profile and historical profile data, can we predict the velocity and power output along the profile?

This approach allows for an analysis where the performance of a rider can be predicted for profile races based on both terrain characteristics and past results. In addition to predicting instantaneous performance, it is important to understand the physiological demands placed on the rider throughout the race. This is important because a rider's current performance is not only influenced by the immediate changes in terrain but also by the accumulated workload from previous efforts. Therefore, after predicting power output along the profile, the next step is to estimate the rider's recent workload based on these predictions. This leads to the third subquestion:

Q3: Given the predicted power, how can we estimate the recent workload?

This method captures how recent efforts weigh more heavily on a rider's current state, providing a more logical view of fatigue and readiness. All these elements form a complex system where the race outcome reflects the actions of the subject rider. The final aspect of the strategy is to udner

1.3 Subject of the thesis

Some riders don't exclusively fit in the aerobic or anaerobic category; they have a balanced physiological profile combining the two body types. The rider analyzed for this research embodies this profile type, demonstrating the ability to win in semi-mountainous races. However, the variability in semi-mountainous terrains, ranging from flat courses with some hills to mountainous terrain with some flat roads, makes it difficult, from the altitude profile alone, to decide if the analyzed rider could be victorious. As mentioned in Section 1.2, the challenge requires a data-driven approach that uses feature engineering and predictive modeling. The final aspect also involves predicting the overall performance by identifying critical moments during the race where the rider could drop off. This leads to the last subquestion:

Q4: Given the enriched profile and historical data, can we predict the probability that a rider has dropped at each point in the race?

By addressing all these subquestions, this project aims to provide coaches with deeper insights in the rider's potential performances. Ultimately, this leads to the central research question:

Research Question: Given an altitude profile and the historical data from a rider, is it possible to predict whether he has any chance of winning the race?

This project aims to give coaches more insight into how the subject rider could perform during the race. By developing a data preprocessing pipeline and predictive models that integrate altitude profiles with key performance metrics, this research tries to improve tactical decision-making before the race. The methodology for constructing predictive models includes multiple data sources such as terrain characteristics, historical data performances, and physiological measurements during these races. When the final model is built, it provides new insight into the strategy for upcoming races and maximizes the rider's potential for success.

2 Background and related work

2.1 Physics-based models of cycling performance

As the starting point of the background and related work, this research introduces a physics-based model that describes the power demands of cycling. It establishes that overcoming various resistive forces determines the power (P) required by a cyclist. Aerodynamic resistance, wheel rotation, rolling resistance, frictional losses in wheel bearings, changes in both kinetic and potential energy and frictional loss in the drive chain are all contributing to predict the power output on certain timestamps [MMC+98].
On climbs, gravitational resistance becomes increasingly significant as the gradient increases, while on flat terrain, aerodynamic drag dominates the resistance. Because this is the case, rider position and surface area of the rider become important in minimizing the power losses due to air resistance. For instance, the elbow position minimizes drag and reduces the required power output across different velocities [FMB+20]. This framework explains why power requirements vary on terrains.

2.2 Physiological Models of Performance Capacity

Secondly, Skiba and Philip Friere (2014) validate a critical power (CP) model using data of three triathletes [Ski14]. Their work determines that the subjects possess a finite work capacity above the CP (known as W'), which becomes depleted when operating above the CP and reloaded when below it. After their model, Poole et al. (2016) created a model of the human physiological response to exercise demands that establishes a framework linking physiological parameters— VO2 max, lactate threshold and efficiency, to sustainable power output [PBV+16]. The critical power, which represents the highest power output sustainable, occurs when blood lactate increases to the limit of tolerance, which gives the riders the feeling of exhaustion.
This framework informs our research about providing the physiological aspects of cycling, it also gives an understanding of when and where the rider may lose contact with their competitors (drop off) during races. When analyzing the altitude profile of the race, and some sections require sustained power outputs above a rider's CP, it will deplete W' reserves. For example, if no sufficient recovery opportunities are available before demanding terrain segments, the W' could reach a threshold, forcing the rider to reduce power and drop off.

2.3 Machine learning

Machine learning is increasingly used in the sports sector, offering insight into modeling complex datasets. For data mining in elite sports, [OZMR13] demonstrated how integrating machine learning algorithms could result in meaningful patterns from physiological and performance data.

Machine learning algorithms have been applied in different areas of the cycling sport. One of these areas is the identification of hidden talent. The research used k-nearest neighbor imputation to handle missing data and integrate expert knowledge into predicting the young rider's potential [JBM23]. These models focus on feature engineering and data quality to identify talents in publicly

available race results. Other research has shown that past U23 results significantly affect future success in cycling, using models like linear regression and random forests [VBVWG23].
For professional cycling, studies have used a learn-to-rank machine learning technique to predict the top 10 contenders in one-day road races [KSdL+21]. Besides predictions on multiple one-day races, research had been done on a single one-day race, the Tour of Flanders, where they tried to predict the result of the whole race [KDSVL20a].

Predicting the drop-off points of a rider represents a relatively unexplored research area. A study examined pacing strategies in endurance events, and making theoretical models for energy expenditure helps show that cycling is called negative pacing [AL08]. The study resembling our research the most was done by de Leeuw et al., who developed a three-stage pipeline for rider-specific predictions based on route characteristics [dHH+20]. Their model calculated the time gained or lost compared to the direct rivals of the rider. It focused more on the time differences between the riders instead of the dropoff prediction.

Overall, this overview demonstrates the potential of machine learning in sports science, extracting meaningful patterns, and giving important insights into data. While research has already been done in this field, predicting drop points has yet to be done. This theoretical framework integrates the knowledge of current methodologies to create a rider-specific drop-off model.

3 Methods

3.1 Data collection

Collecting data has become more popular over the years, and more teams are starting to implement a data-driven approach to train for races. Computers on the body of the rider, on the bicycle, are all being used to capture important data on how a training or race went.
The data is provided in a spatio-temporal format, combining spatial (location-based) and temporal (time-based) components. It is captured within FIT (Flexible and Interoperable Data Transfer) files, which are used to record sports and fitness activities. What makes these file formats suitable for sports is that each row in the file represents one second of the activity; this provides a detailed picture of how the training or race has developed over time. The rows contain various features, the most important features within these files are:

- **Timestamp:** The data frame's index, where every row in the dataset is a second in the race.
- **Enhanced altitude (Meters):** Elevation measurement of the course that depicts the course's profile.
- **Power (Watts):** Rider's power output that shows physical exertion.
- **Speed:** The velocity of the SR reflecting his progression through varying terrain.
- **Longitude and Latitude:** Both are used to pinpoint the rider's location at that moment in the race.

It ensures that the file contains a comprehensive analysis of the athlete's performances over time and space.

3.2 Data preprocessing

3.2.1 ProCyclingStats API

The first step of the preprocessing pipeline was to retrieve general data on races, which would be used for the final prediction model. The source selected to tackle this problem is ProCyclingStats (PCS). This website collects all sorts of data, such as race information, the riders' rankings, and finishing times.

Using the PCS, all relevant data for the project can be extracted, ensuring consistency and accuracy. Because the data are easily accessible, they can be prepared for the next steps of the preprocessing pipeline.

3.2.2 List of riders

Using the PCS API, the first goal was to obtain all the different riders, including our SR, who were or are part of the Tudor Pro Cycling team between 2022 and 2024. Subsequently, all the races participated in by the riders during the same period were extracted. Races such as team time trials (TTT) and individual time trials (ITT) will not be included because they differ in racing strategies, aerodynamic positioning, and effort distribution, which makes them less representative of typical road race dynamics. After the selection procedure, the remaining races of our specific rider will be treated as primary races. Each race is checked to see if other riders from Tudor have cycled the same race. This results in a list of races participated in by our SR, accompanied by a list of riders who competed in the same race.

3.2.3 Selecting races

The selection procedure is performed with the known list of riders in each race, and five criteria are applied to exclude races. Beginning with the exclusion of races, the starting list is checked to see if the SR did not start (DNS) or did not finish (DNF). A DNS could happen when the SR had been registered for a race but could not start due to sudden changes, such as illness, before the race. A DNF occurs during the race when the SR decides to give up. After the races with DNFs and DNSs are discarded, the starting time and the FR from whom Tudor has data from the final rankings list are extracted using the API. Stages where our SR or the FR was in a breakaway are excluded due to the unique dynamics during a race with a breakaway that could bias the overall performance. Moreover, stages where the fastest rider of Tudor for a certain race, which could also be the SR, finished significantly behind the winner were excluded because of the lack of necessary data to know where the drop point was.

3.2.4 Noisy data

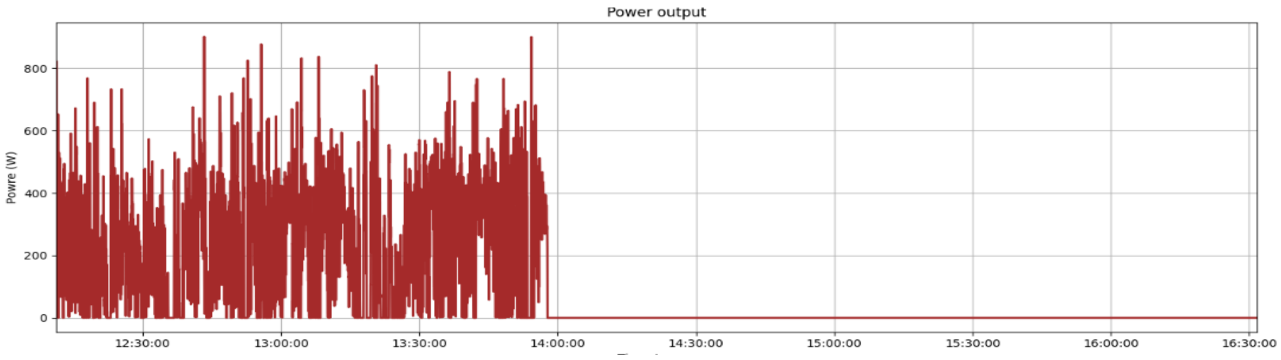

Figure 1: Power profile error

Each stage labeled as suitable for the prediction model should be checked to address potential data quality issues. The reliability of the data is of great importance in ensuring the quality and accuracy of the predictive model. Errors that regularly occur in these files can be categorized into three categories:

- **Switch of bicycle**: Mid-race bicycle changes are common during bike races. The riders could puncture their tire, crash, or change to a different bike because it gives them a strategic advantage to conquer parts of the race. It results in a sudden cut-off in the data.

- **Computer captures wrong or no data**: Sensor failures can result in data gaps, impossible values, or random data. For example, rain can change the altitude readings, affecting columns like power measurement and altitude readings. Figure 1 shows an example of a power profile error.
- **Measures with other applications**: Some riders measured themselves with other applications during a race. For example, Strava is an application used to measure training or races that records activities differently than in a FIT file. When converting the data collected by Strava to a FIT file, the values are interpreted differently and therefore not suitable for the experiments.

3.3 Calculating dropoff

Calculating the dropoff is the most important step. The problem encountered is that the dataframes of both riders have different lengths, as riders mostly activate their recording devices at varying times before the start. Therefore, the raw timestamp data cannot be directly compared and should be synchronized first.

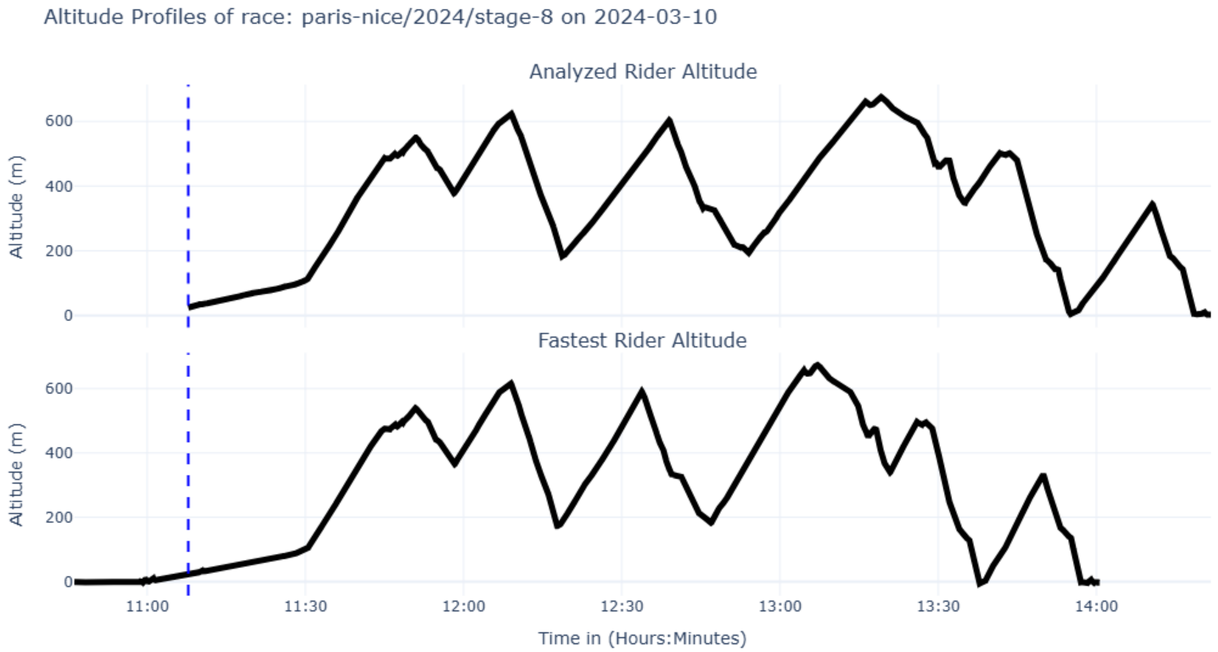

Figure 2: The bottom graph shows that the FR started his computer earlier than the SR, showing a longer segment in the beginning, which needs to be recalibrated

The time closest to the official starting time set by ProCyclingStats was selected to address this challenge. For both riders, the timestamp closest to the official start time was identified. Eventually, the dataframe was chosen with the earliest timestamp that preceded, but was closest to, the official start time. To maintain the integrity of the data, the distance values were recalibrated, meaning that both riders started at 0 meters and started synchronously. When the recalibration has finished, the dynamic time difference can be calculated. Both spatial data (distance) and temporal data (seconds) are used for the calculations. The following formula has been used to calculate the dynamic time difference:

```
function CALCULATE TIME DIFFERENCE($p_{\text{SR}}$)
    $P_{\text{FR}} \leftarrow \{p \in \text{FR data} : p.\text{distance} \geq p_{\text{SR}}.\text{distance}\}$
    if $P_{\text{FR}} \neq \emptyset$ then
        $p_{\text{FR}} \leftarrow$ earliest point in $P_{\text{FR}}$
        return $p_{\text{FR}}.\text{time} - p_{\text{SR}}.\text{time}$
    else
        return null
    end if
end function
```

For each position in p_{SR}, the algorithm tries to find the earliest point where the FR reached at least the same distance. After finding the earliest point, the formula: $p_{\text{FR}}.\text{time} - p_{\text{SR}}.\text{time}$ calculates the difference in time between the FR and SR. Positive values indicate that the rider is getting or has already been dropped, and negative values indicate that the SR has overtaken the FR, which doesn't happen often.
To demonstrate the calculation of the time difference function, the red line in Figure 3 shows the time of the SR compared to the FR. The blue dotted line around 14:45, shows the moment the SR has been dropped.

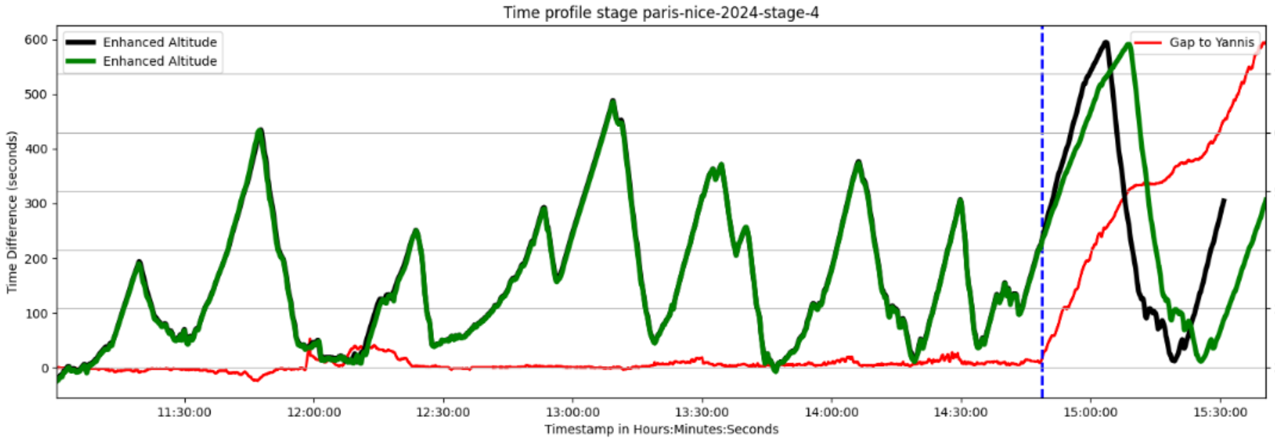

Figure 3: The red line shows the time gap in seconds between the SR and the FR. The dotted blue line shows the moment the SR cannot keep up with the FR and drops off

After the time difference is calculated, the points where the FR has a greater than or equal to 60-second advantage towards the SR get converted to binary numbers 1. The other numbers get converted to 0, which results in a binary array with 0 saying the rider is not dropped and 1 saying the rider is dropped. This is visualized in Figure 4, where the dark red points on the altitude represent the points where the SR has dropped off.

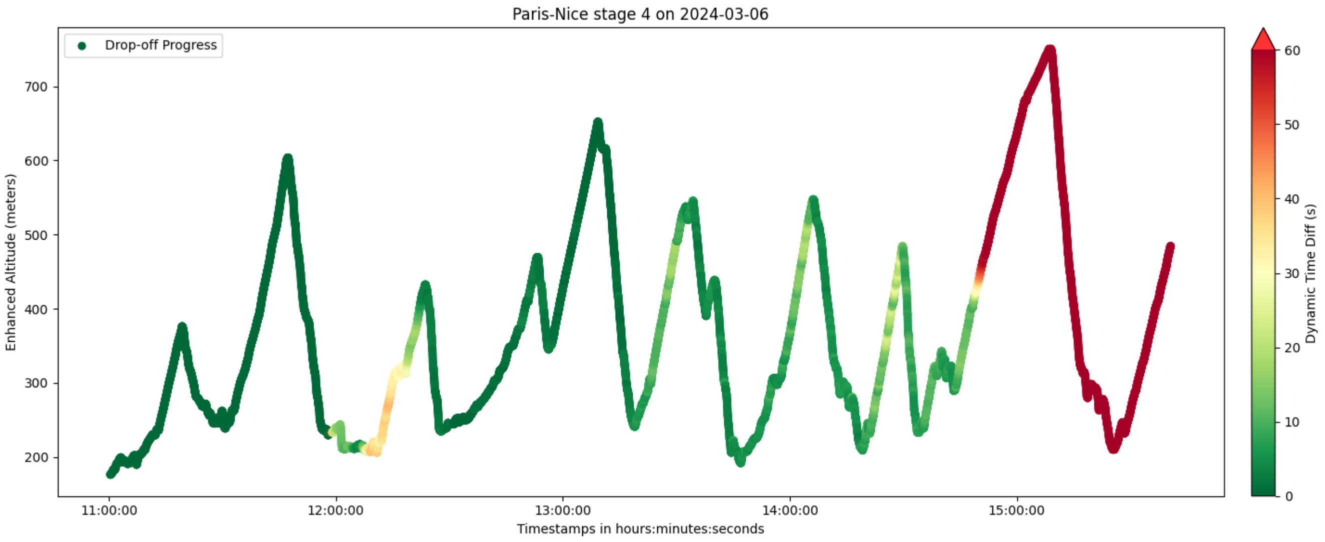

Figure 4: Converted drop visualized on the altitude profile

3.4 Feature Engineering and Temporal Aggregation

To improve the predictive modeling, the raw time series data collected during the races were enriched with engineered features. This process was designed to capture both the spatial and temporal dynamics of the cycling performance, as well as the physical characteristics of the racecourses. By modifying the original sensor measurements into new descriptors, the models could recognize patterns and relationships relevant to the rider better than with only raw data.

3.4.1 Core features

Because the measurements varied in features collected, only the core features were considered and served as a base for feature engineering. The core features can be divided into two classes: the historical and the pre-race ones. The historical features contain power output (in watts) and speed (in kilometers per hour), which can be used to inform the model about athlete's recent or long-term performance. The pre-race feature refer to information the DataFrame contains before the race, this includes the geographical profile of the race, such as distance covered (in meters), altitude (in meters), the geographical positions longitude and latitude. The pre-race features are delivered in a .FIT file format From these core features, several categories of features were derived:

3.4.2 Temporal and Spatial Features

To explore and label each data point within the race, two features were constructed that explain how far the race has progressed: stage completion and distance to the finish line. The percentage of stage completion determines how much of the race has been covered, enabling the model to take higher power outputs at the final stage of the race into account. Distance to finish provides an absolute measure, allowing the model to relate performance to remaining racing distance.

3.4.3 Elevation Features

Elevation gain and loss are critical for knowing how much effort has been put in climbs and descends. For this reason, cumulative elevation gain and loss were calculated at each point, representing the total climbing and descending efforts. When certain thresholds of high cumulative elevation gain have been reached, the predictive model could increase the risk of dropping off, because large climbing efforts can lead to fatigue. However, periods of significant elevation loss could provide opportunities for recovery and reduce the effects of the previous climbs. By incorporating both metrics, the predictive model can potentially evaluate the balance between exertion and recovery throughout the race.

3.4.4 Climb and descent features

To further characterize the terrain, features that describe the length and height of climbs and descents were introduced. A climb (or descent) was defined as a sequence of 15 consecutive points of elevation gain or loss. The total distance of the climbs was also calculated, providing the model of the scale and difficulty of the key sections of the course. Last but not least, the instantaneous altitude difference was included to capture rapid elevation changes that may trigger tactical responses.

3.4.5 Gradient features

The topographical characteristics significantly influence the performance of the SR. Capturing the road steepness (gradient) gives a critical context for performance predictions. Thus, the gradient was quantified for multiple segments ranging from 10 to 200 meters. The gradient is defined as the ratio between the vertical and horizontal distance, expressed as a percentage.

$$\text{gradient} = \frac{\Delta h}{\Delta d} \times 100\%$$

Where:

- Δh is the change in altitude
- Δd is the change in distance

- Segments from 10 to 20 meters: capture specific changes in terrain that could affect the power requirements on certain parts of the course.
- Segments from 50 to 100 meters: provide balanced insights into parts of the course where small changes in the surface are neutralized.
- Segment of 200 meters: captures the global outline of the race; it depicts the broader challenges that the riders will face.

3.4.6 Speed and power features

To capture the performance of the cyclist during the race, rolling averages are implemented for both power and speed. These features provide insight into the physiological state and performance capacity of the riders.
Four time windows were selected to represent various physiological and tactical aspects of each cycling performance.

- **Short-term averages (5-10 seconds):** Capture immediate responses to terrain changes, tactical accelerations, or attacks.
- **Medium-term averages (25 seconds):** Represents the ability to maintain power output over medium durations, corresponding to critical sections, such as short climbs.
- **Long-term averages (50 seconds):** Captures the long-term performance capacity, helping to identify patterns of fatigue development before the S.R. drops off

3.5 Predictive modeling approaches

To address the question of whether the velocity and power output can be predicted along a future profile using historical data, a supervised machine learning technique is used. First, all available race stages are processed as discussed in Section 3.4 to extract interesting features from the altitude profile and historical performance data, including terrain characteristics, gradient, and physiological metrics. These features are used as inputs for regression models to predict the speed and power output at each point along the profile.

Once all races have been retrieved and featurized, they are concatenated into a single DataFrame with flag columns distinguishing between different races. This structure allows the original format of the enriched races to be preserved while enabling a detailed and unified analysis across the dataset.

3.5.1 Cross-validation

To train and evaluate the predictive models, a Leave-One-Out cross-validation (LOO-CV) algorithm is employed, which is well-suited for datasets with a limited number of samples. In this approach, each fold consists of one stage serving as the test set, while the remaining stages are used for training. This process is repeated until every stage has been used as the test set exactly once. The approach maximizes the use of available data by ensuring that each stage contributes to the model's training. Furthermore, the computational demands are manageable due to the relatively small dataset size.

The prediction is done with a Leave-One-Out (LOO) cross-validation algorithm, which suits datasets with limited samples. The LOO algorithm works as follows:

- For each fold (iteration), one stage serves as a test set while the remaining n-1 stages are training stages.
- N represents the total number of verified race stages.

This approach maximizes the use of available data by ensuring that each stage contributes to the model training. Besides maximizing the data, it is computationally achievable because the dataset size is small.

3.5.2 Workflow of the prediction

Within each fold of the cross-validation, the data preparation and modeling workflow follows a systematic process. First, the data is split into separate test and training sets. To prevent data leakage, features that could reveal information about the target value and are included in the historical datasets, such as rolling speed and power averages, are excluded from the dataset. The feature removal is performed to restrict the model to only terrain-related attributes and non-speed- or power-related metrics, ensuring the predictions are only based on information that would be available in a real-world scenario. After the leakage prevention, the target value is isolated from the remaining futures. The resulting feature matrix and target column are formatted appropriately for the chosen models.

3.5.3 Model Architecture

RandomForest is the baseline model because it handles non-linear relationships well and is suitable for small datasets. The estimator's hyperparameter is set to 100, balancing the computational efficiency and model robustness. The maximal depth of the tree is set to 5 to prevent overfitting the training data.

The second regression model used is XGBoost, which builds trees that correct errors from other trees. This model was chosen because of its fast computational time and the capturing of subtle changes in the speed. For faster computation time, the dataframe is converted into an optimized data structure (DMatrix), which enhances the memory efficiency and training speed. The training was done with two hyperparameters, the learning rate set to 0.1, and the number of boosting rounds was set to 100. This ensured a balancing model complexity with generalization capacity.
Following the model training, the prediction process applies the training model to the test stage for that specific fold. The predictions are added to the dataframe of the test stage. Afterwards, we evaluate the performance of the model based on three metrics

- Root Mean Squared Error (RMSE): Measures the prediction accuracy with sensitivity to large errors
- Mean Absolute Error (MAE): Measures average speed prediction errors without overemphasizing the outliers.
- Coefficient of Determination (R^2): Indicates the proportion of speed variance.

3.6 Workload of the rider

The power output prediction model follows the same framework as the speed prediction model mentioned in Section 3.5. An extended DataFrame with the predicted speed variable incorporated in it is given as a new input for the power output model to predict it with a higher accuracy. The result of including the predicted speed value will create a more comprehensive representation of the state of the SR.

Because the same method is used, data preparation, feature selection, and model construction have the same structure. The difference in the power output model comes after predicting the target value, where during each cross-validation fold, the model generates a convoluted version of the power output. The convolution step is an important physiological phenomenon that shows how the body responds to extensive cycling efforts. Recent power outputs have more of a physiological impact compared to those exerted in the past, and convolution mathematically represents the decreasing influence of past data points. With convolution, we simulate how the body functions by progressively decreasing the influences of the power output by assigning different weights to past data points.
Several kernel functions can be used to define how much each past point weights in the convolution process. The kernels experimented with were the uniform, exponential, and logistic kernels

- **Uniform kernel:** This kernel assigns equal weights to all points within a specified window length; points outside the window are not counted. The method of the uniform kernel does not resemble human physiology, as it assumes that each point inside the window has the same importance.
- **Exponential kernel:** The exponential kernel applies rapidly decaying one-sided weights, giving the most recent points the highest influence. This method captures the decaying influence of the past points but tends to over-emphasize the most recent points and under-emphasize the points in the past.

- **Logistic kernel:** The logistic kernel applies an S-shaped weighting curve, gradually reducing the influence of older efforts 5. This kernel mimics human physiology the best because the impact of exertion fades progressively, not abruptly. Recent efforts are weighted more heavily, while older efforts maintain a small, non-zero influence reflecting the body's cumulative fatigue response.

For this study, the logistic kernel was chosen to simulate physiological effects on the body with the following formula:

$$K(i) = \frac{1}{1 + e^{-\alpha \cdot \frac{2(i - w/2)}{w}}}$$

- $K(i)$ is the logistic kernel function
- α is the steepness of the slope in the logistic function
- w is the window size parameter defining the temporal scope of the physiological effects
- i is the position index within the window, ranging from 0 to $w - 1$

3.6.1 Parameter optimization

Two parameters are critical for optimizing the logistic kernel:

- **Window Size (w)**: This parameter determines how long the physiological exertion impacts the rider. The window size is directly related to capturing immediate fatigue responses and longer-term adaptation. Smaller window sizes emphasize recent power outputs but may miss cumulative stress patterns, while larger windows capture long physiological effects but may forget immediate performance signals
- **Alpha Parameter: (α)**: The steepness parameter controls the transition characteristics of the logistic function. The higher alpha creates a sharper transition, while the lower alpha values produce a gentler transition that distributes influence more evenly across the windows.

The model achieves a more realistic simulation of how the cyclist's past and recent efforts combine to influence current power output and performance by applying the logistic kernel to the convolution process.

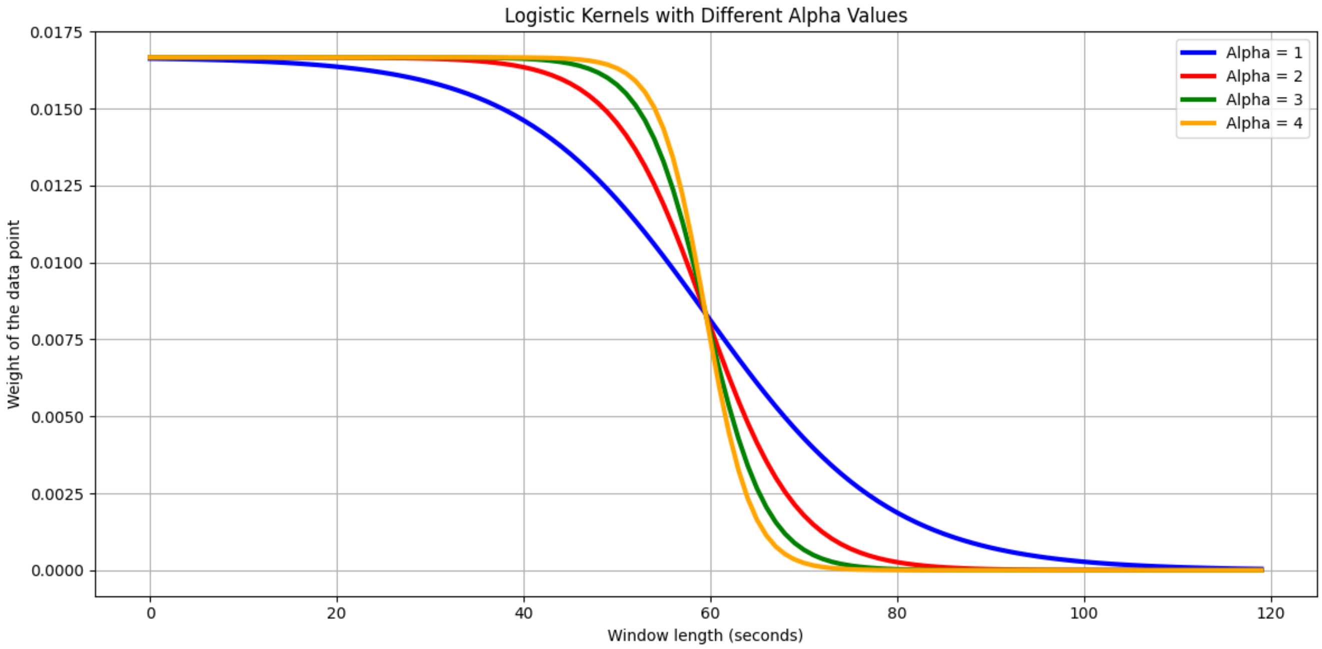

Figure 5: Logistic convolution with alpha values ranging from 1 to 4 and a window length of 120 seconds

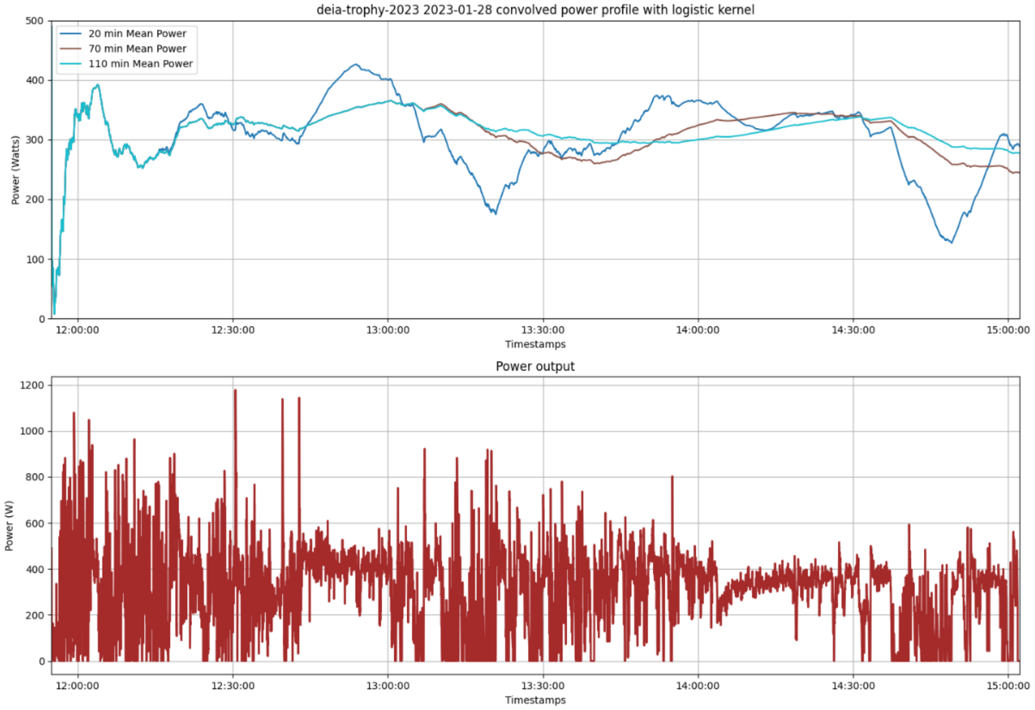

Figure 6: Example of how the power gets convoluted with different window lengths

3.7 Drop off model

Lastly, the drop-off model, where the identical LOO methods in Section 3.5 were used. The difference in the drop-off model is that classification models have been used instead of regression models. As mentioned in Section 3.3, when the FR has a ≥ 60 advantage over the SR, it is defined as 1; otherwise, 0. The objective of the drop-off model is to identify where the SR is likely to experience a drop-off moment.

3.7.1 Class imbalance

This classification problem presents a challenge due to the highly imbalanced data sets, with some having <5% of the target value, while the other points represent situations the SR can keep up with the pace. The class imbalance could lead to biased models that favor the majority class, resulting in poor detection of true drop-off points.

3.7.2 Model selection

Four different models (Random Forest, XGBoost, Logistic Regressor, LGBM) were used to predict the drop-off points.
First, the Random Forest, a learning method that constructs decision trees during training and outputs the mean prediction of the individual trees. Random Forests are resilient to noise and can capture non-linear relationships in the data.
Secondly, XGBoost, a gradient boosting framework that builds decision trees sequentially, where each tree tries to correct the errors made by previous trees. XGBoost is also known for its high performance and has techniques to prevent overfitting. This makes it a well-suited model for imbalanced classification, which applies in our dataset. Random Forest and XGBoost can handle high-dimensional data, which is common in professional cycling, and detect patterns that may precede a drop-off.
Third, LGBM (Light Gradient Boosting Machine), which is optimized for speed and memory efficiency. It grows trees leaf-wise rather than level-wise, meaning it can lead to faster training and better accuracy on large datasets.
Last but not least, the Logistic Regression, which estimates the probability that a given input belongs to the positive class by applying a logistic function to it. The logistic regression is a simple and interpretable model, especially if the relationship between the features and the target is linear. This does not apply to our data, hence it was interesting to test how such a model would perform.

4 Experimental setup

This section presents the methodological approach used to construct the predictions for cycling performance analysis. The experimental design follows a systematic approach to data preprocessing, feature engineering, and model development.

4.1 Data Collection and Filtering

With the data collection approach used in Section 3.2.2, 46 riders who had competed for the Tudor Pro Cycling team across various competitions, ranging from World-Tour events to Continental races. The diversity in the altitude profiles of these races ensures an inclusive terrain profile representation, which is crucial for developing reliable prediction models.

Besides the 46 riders, 195 race stages were collected where the subject rider (SR) competed alongside at least one teammate. To ensure data quality, a filtering process was applied as outlined in Section 3.2.3. The filtering criteria included:

- **Performance proximity criterion**: Stages where the fastest rider of Tudor finished outside a 30-second window from the stage winner were excluded. This ensures that only competitive scenarios are analyzed, because that team's tactics for a certain race could be to preserve energy for upcoming races. Also the drop-off points cannot be meaningfully identified when the FR also had no chance of winning.
- **Completion status criterion**: If the SR or FR did not cross the finish line (DNF) or did not start (DNS) are excluded, because a part or no data of the race are available of the riders.
- **Breakaway exclusion**: Stages where the power output is higher throughout the race, potentially biasing the overall performance analysis.

When races suitable for the experiment were selected, an assessment of the quality of the data was made. Files containing technical errors or incomplete recordings, as described in Section 3.2.4, were removed from the final dataset. Following the application of these criteria, the dataset was reduced from 195 to 31 suitable race stages for model development.

4.1.1 Selecting correct files

The Tudor dataset contained multiple recordings per day, as riders occasionally started their bike computer before the race started. This led to the selection procedure of identifying the correct race files from both the FR and SR on the same day.

The file selection process utilized the ProCyclingStats (PCS) API to retrieve official race start times. By cross-referencing these timestamps with the recorded data, the algorithm could select the correct race file corresponding to the actual race measurements. This ensured that only competitive data was included in the analysis.

4.1.2 Calculating time difference

As part of the experimental setup, the gap between the SR and FR was calculated throughout each race stage. The gap calculation algorithm has a distance-based system, where each data point from the SR is matched to the corresponding position of the FR based on the cumulative distance covered. This distance-based approach enables performance assessment by analyzing the time compared to the FR and indicating the point where the rider drops off.

4.1.3 Dataset Characteristics

After calculating the time difference, the drop points can be assigned. As stated in Section 3.3 if the FR is 60 or more seconds in front of the SR, we consider this a drop point and mark it as 1; all the other points are marked as 0. Across the 31 analyzed racing stages, a total of 461,922 data points were collected, with 120,236 instances classified as drop-off events (26.03%) and 341,686 instances representing non-drop-off points (73.97%). This drop-off events are not uniformly distributed across all stages, with 24 out of 31 stages containing at least one drop-off occurrence. However, the individual stage-level imbalances are more present than the aggregate statistics suggest. Individual racing stages sometimes have extreme class imbalances, creating methodological challenges, as models must learn to identify drop-off patterns from stages with different class distributions while maintaining generalization capability across all racing conditions.

4.2 Feature engineering setup

A systematic feature engineering and temporal aggregation process was implemented to address the challenge of extracting meaningful information from raw, multi-variate time series cycling data. The goal was to transform the original sensor measurements into a set of features that capture both the immediate and contextual aspects of rider performance, as well as the physical characteristics of the racecourse. With the newly generated features, the performance of the final drop-off model should be improved.

4.2.1 Different Testing Setups

The evaluation process begins by establishing a baseline model that only uses core features, referred to as **Representation 1**. This representation includes solely the core features, which can be extracted from the pre-race altitude profile. The purpose of the baseline is to act as a reference point for evaluating the impact of further feature engineering steps.

Given the datasets' overall class distribution as mentioned in Section 4.1.3, any effective classification model must outperform the naive baseline approach. A majority classifier that always predicts "non-drop-off" would achieve 73.97% accuracy by simply classifying all instances as the dominant class. The expectation is that all classifiers should perform the same as or better than an accuracy score of 0.739 to be considered effective.

In the next setup, referred to as **Representation 2**, we extend the core feature seet with a range of terrain-related features derived from the original time series. These include standard statistical aggregations, such as rolling means, calculated over various window sizes. These features aim to capture short-, medium-, and long-term trends and fluctuations in the data, which may be relevant for predicting the target variable.

Representation 3 includes domain-specific features. This includes the predicted rolling averages of speed and power output, which were obtained using earlier regression models trained on features of both Representation 1 and Representation 2. These predictions are intended to reflect the rider's current physical state, supplementing the terrain and core features with estimated internal factors. When **Representation 3** has been analyzed, the best model based on metrics and feature importance is chosen to do the last test with **Representation 4**.

Finally, **Representation 4** introduces a measure of the rider's recent workload, computed by applying a convolution operation to the predicted power output time series. The best configuration of alpha size and window length is chosen to improve the performance of the model. This transformation smooths the data and emphasizes exertion trends across different time horizons. The convolution window length and exponential decay parameters were tuned to best summarize physiological stress over time.

4.2.2 Prediction Models

The experimental setup consisted of several stages. In the initial stages, I focused on the core features and then on the core features enriched with derived terrain-related features. For these stages, four different classification models were trained and tested: LightGBM, XGBoost, Logistic Regression, Random Forest.

4.2.3 Evaluating Metrics

For each model and feature set, the ROC-AUC, f1 score, accuracy, precision, and recall were calculated. This allowed for a thorough comparison of both the impact of feature enrichment and the performance of different classification algorithms. In addition, to better evaluate the predictive performance, feature importance was analyzed for each model. This analysis helps to identify which features contribute the most to the model's decisions, offering insights into the value of feature-engineered features and the effectiveness of temporal aggregation strategies.

4.2.4 Best-Performing Model

After identifying the best-performing model based on the metrics and feature importance, the model was used in the last stage of the setup. In the last stage, the featureset was enriched by including predicted speed, predicted power output and the workload of the SR. The best model from the earlier tests was then retrained and evaluated with these additional features to assess their added value.

By comparing results along all the stages, from the baseline models to models with enriched feature sets, it becomes possible to quantify the added value of each feature engineering step and modelling choice. This systematic approach ensures the impact of temporal aggregation, predictive features, workload, and model selection is tested and clearly demonstrated within this study.

4.2.5 Testing significance

To validate if the multi-variate time series can be enriched with features during temporal aggregation, is to make a statistical significance testing framework. This framework will be implemented to assess if each addition of features provides meaningful improvements beyond statistical noise. The model that performed the best, as mentioned in Section 4.2.4, will also be used to determine if enriching the dataset has a significant impact. Four tests will be conducted across the accuracy and ROC AUC performance metrics. These four tests are:

1. **Representation 1 vs. Representation 2**
2. **Representation 2 vs. Representation 3**
3. **Representation 3 vs. Representation 4**
4. **Representation 1 vs. Representation 4**

Each comparison will utilize the LOO-CV results paired t-test, accompanied by Cohen's effect size calculation to determine practical significance beyond statistical significance. Successful validation will provide empirical evidence that temporal aggregation-based feature engineering significantly improves drop-off predictions. This would confirm that the multivariate time series can be effectively enriched through geographically derived features, predicted physiological metrics, and convoluted power data.

4.3 Predicting velocity and power setup

The experimental setup is designed to evaluate the ability of machine learning models to predict the speed (in kilometers per hour) and power output (in watts) of the SR from altitude profiles and related features. The process begins with the construction of a base model, which only uses the core features mentioned in section 3.4.1. These features represent the bare minimum information available from the altitude profile.

After establishing the baseline, the experiment proceeds to the enriched model. In this phase, additional features are added by featurizing the data, which are mentioned in section 3.4. These features add new insights over the data and will likely improve the performance of the regression models.

4.3.1 Prediction Models

The models used for predicting the speed and power are the Random Forest and XGBoost regressors. The RandomForest is configured with 100 estimators and a maximum tree depth of 5, while the XGBoost model uses 100 boosting rounds and a learning rate of 0.1. These parameters are chosen because they are commonly used defaults in the literature and standard implementation. These values are not highly specific or optimized for this dataset but serve as reasonable starting points for model comparison and baseline evaluation. The models are trained and evaluated using a Leave-One-Out Cross Validation (LOO-CV) approach at the stage level, as mentioned in Section 3.5.1. This method provides a robust estimate of model generalization to unseen data.

Both models are trained to predict five different rolling averages computed over different temporal windows (1, 5, 10, 25, and 50 seconds) of the target value speed and power. The approach serves

4.3.2 Evaluating Metrics

Throughout the experiment, model performance is monitored by using metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). By comparing the results of the base model and enriched model, the experimental setup allows for a clear evaluation of the impact of feature engineering and data featurization on predictive accuracy. With this setup, we can research to what extent we can accurately predict the velocity and power output along a future altitude profile and to what extent feature engineering improves this capability.

With the RMSE, MAE, and R^2 being calculated, a statistical validation is conducted through paired t-tests comparing the XGBoost and Random Forest on their best-predicted target variable across all cross-validation folds. This approach ensures fair comparison by evaluating each model's performance on the target variable where it achieves optimal predictive accuracy.

Effect size analysis using Cohen's d complements a statistical signifcance testing by measuring the magnitude of performance improvements between their optimal targets. This dual statistical approach ensures that both statistical reliability and practical relevance are established when determining the best model-target combination.

4.3.3 Stages of Predicting Velocity and Power

The making of the speed and power prediction models was done in two stages: first, models were developed to predict speed from the altitude profile and related features. Once the best speed prediction was established, it was used as an additional input feature to improve the prediction of the power output. This approach was chosen based on the intuition that speed, being a smoother and less variable signal than power, would be easier to predict directly from the available features. The main reason for this was to take advantage of the more stable speed predictions to improve the accuracy of power output estimation and make the power output estimations. By using this approach, the power output estimations will also be smoother, which could lead to the final drop-off model predicting drop-off points more easily.

This experimental design thus not only evaluates the feature engineering, but more explicitly tests the possibility of predicting the cycling performance from altitude and historical data alone.

4.4 Estimate recent workload

This part of the study aims to estimate the workload of the cyclist based on the power output. Recent workload is an important psychological indicator, as it reflects cumulative exertion experienced by the rider over a recent period. The workload could also be a strong predictor of drop-off events or fatigue.

4.4.1 Chosen Kernel

To estimate the recent workload, the predicted power is processed using a temporal convolution with a logistic kernel. The convolution operation effectively computes a weighted moving average of the power output, where more recent values are given higher importance 3.6. The logistic kernel is chosen for its S-shaped weighting, which allows for a gradual transition in the influence of past power values.

The experimental setup begins with obtaining the power output from the previously trained regression model. Then, for each time point in the time series, the recent workload is calculated by convolving the predicted power with logistic kernel over a specified window length. The kernel parameters, window size and steepness of the logistic function (alpha), are systematically varied to investigate their effect on the resulting workload feature.

4.4.2 Parameters of Window Size and Alpha

To ensure an experimental environment of the workload estimation, different window lengths and alpha values are tested. The window lengths differ from 10 minutes up to 120 minutes in increments of 10 minutes. Besides window length, the alpha parameter was tested across a range of value from 0.5 to 4 with increments of 0.5. For each configuration, the resulting workload feature is added to the input dataset for the drop-off model. The performance of the workload is than assessed using the classification metrics ROC-AUC, f1, precision, accuracy, and recall. These metrics will be visualized in a graph plot to see which window length and which alpha value are optimal.

By comparing the predictive performance across different workload configurations, the optimal window length and kernel parameters can be identified. This approach allows for a detailed investigation of how recent workload contributes to the prediction of drop-off points in cycling races. The results provide insight into the temporal dynamics of exertion and their relationship to rider performance and fatigue.

4.5 Final Model Setup

The objective of this subquestion is to predict, at each point of the race, the probability that a rider will drop off, using the fully enriched feature profile. This profile includes not only the terrain-related features, but also advanced engineered features such as predicted speed, predicted power, and recent workload.

4.5.1 Final Enriched Dataset

The experiments begin by assembling the final dataset, where each row corresponds to a time point in the race and contains all available features. This includes raw sensor data, rolling averages, terrain features, predicted features, and workload derived from the predicted features. The target variable is a binary indicator denoting whether the rider has dropped off at that specific time point. To estimate the drop-off probability, a classification model is trained for each time point. Based on the earlier results, the best model is chosen for this task.

4.5.2 Evaluation Metrics Final Dataset

The model's perfomance will be evaluated with the same metrics done in Sections 4.4 and 4.3, with ROC-AUC, precision, recall, F1-score, and accuracy. These metrics will provide a comprehensive view of the model's ability to distinguish between drop-off and non-drop-off points, as well as its effectiveness in handling class imbalance.

In addition to overall performance, the predicted probabilities are analyzed to assess which segments have a high probability of drop points. Feature importance analysis is also conducted to identify which features contribute the most to the model's predictions, offering insights into which feature is the most important.

Finally, the results are interpreted in the context of the study's objectives. The analysis focuses on how the inclusion of features imporves the model's ability to predict drop-off points. This approach not only quantifies the predictive power of the enriched profile, but also demonstrates the practical value of advanced feeature engineering in sports analytics.

5 Results

Given a future profile and historical profile data, this section addresses the question of whether we can predict the velocity and power output along the profile. Importantly, the raw values plus four different rolling average window sizes (5, 10, 25, and 50 seconds) were used for both speed and power as prediction targets. This approach was used to identify the optimal window size for predictive modeling while accounting the variability in cycling data.

5.1 Speed prediction

The base models are trained using only the core features (**Representation 1**), to establish a performance baseline.

5.1.1 Base speed prediction models

Window Size	RMSE (km/h)	MAE (km/h)	R^2
Raw Speed	15.54 ± 2.73	12.55 ± 2.32	-0.081 ± 0.171
5-second avg	15.41 ± 2.76	12.44 ± 2.34	-0.072 ± 0.165
10-second avg	15.29 ± 2.84	12.34 ± 2.39	-0.075 ± 0.180
25-second avg	14.82 ± 2.85	11.97 ± 2.43	-0.066 ± 0.180
50-second avg	14.31 ± 3.00	11.53 ± 2.54	-0.070 ± 0.196

Table 1: Prediction Performance of Base Model Speed using XGBoost

Window Size	RMSE (W)	MAE (W)	R^2
Raw Power	14.86 ± 2.52	12.08 ± 2.19	0.011 ± 0.140
5-second avg	14.78 ± 2.54	12.01 ± 2.20	0.012 ± 0.141
10-second avg	14.63 ± 2.56	11.89 ± 2.22	0.015 ± 0.143
25-second avg	14.19 ± 2.65	11.55 ± 2.29	0.022 ± 0.149
50-second avg	13.63 ± 2.78	11.08 ± 2.41	0.029 ± 0.168

Table 2: Prediction Performance of Base Model Speed using Random Forest

The key finding from altitude profiles alone, are that they are insufficient for accurate speed predictions. For the Random Forest model, the R^2 values are slightly better than 0.0, whereas the R^2 values of XGBoost are all negative. A negative R^2 indicates that the model performs worse than a simple baseline that always predicts mean speed. This suggests that despite the slight improvements with increased smoothing, the model was constantly unable to capture the patterns necessary for accurate speed prediction with the current set of features and approach.

5.1.2 Enriched speed model performance

When the features of **Representation 2** are added to the feature set, the prediction accuracy improves significantly compared to the baseline, demonstrating a critical role of terrain-derived dynamics in modeling rider performance during a race.

Window Size	RMSE (km/h)	MAE (km/h)	R^2
Raw Speed (no avg)	6.37 ± 0.70	4.61 ± 0.54	0.795 ± 0.110
5-second avg	6.56 ± 0.72	4.76 ± 0.57	0.780 ± 0.117
10-second avg	6.75 ± 0.79	4.93 ± 0.62	0.762 ± 0.131
25-second avg	6.97 ± 0.87	5.20 ± 0.66	0.733 ± 0.138
50-second avg	7.28 ± 1.03	5.51 ± 0.81	0.686 ± 0.157

Table 3: Prediction Performance speed XGBoost

Window Size	RMSE (km/h)	MAE (km/h)	R^2
Raw Speed (no avg)	7.52 ± 0.57	5.59 ± 0.49	0.722 ± 0.119
5-second avg	7.64 ± 0.60	5.72 ± 0.50	0.709 ± 0.132
10-second avg	7.73 ± 0.65	5.81 ± 0.55	0.696 ± 0.135
25-second avg	7.82 ± 0.84	5.93 ± 0.69	0.671 ± 0.148
50-second avg	7.89 ± 1.01	6.01 ± 0.82	0.638 ± 0.159

Table 4: Prediction Performance speed RandomForest

The results of the speed performance model showed a clear relationship between performance and the level of smoothing applied to the speed variable. As presented in Tables 3 and 4, the best performance metrics of XGBoost and RandomForest models were achieved when no smoothing was applied. For XGBoost this included an RMSE of 6.37 km/h and an R^2 of 0.795. The Random Forest model scored lower with an RMSE of 7.52 km/h and an R^2 of 0.722.
We observed a decrease as the temporal window for rolling average calculations increased. When comparing the largest rolling average with the predictions of the raw speed data, the performance of XGBoost decreased from an R^2 of 0.795 to 0.686. The same happened to the Random Forest model, where the R^2 dropped from 0.722 to 0.638.
The loss of predictive performance is likely due to the importance of small decisions in brief moments in the races. These decisions differ from tactical accelerations to responses to gradient changes and cornering adjustments. The preservation of these characteristics appears essential for creating a better prediction model.

5.1.3 Evaluation of Representation 1 vs Representation 2

The performance metrics across the different model setups highlight a clear distinction between models trained on **Representation 1** and **Representation 2**.

There was a drastic difference between **Representation 1** and **Representation 2**. Both base models, regardless of the rolling average window or algorithm choice, as shown in Tables 2 and 1, demonstrated poor predictive performance with negative or near-zero (R^2 = -0.081 - 0.02) values. On the other hand, the enriched models predict rider speed with high accuracy, confirming the importance of using more detailed features derived from the race profile. The impact of which machine learning model was used was also visible. While both models showed the same performance trends, XGBoost performs better overall in terms of best RMSE, MAE, and R^2.

To summarize the findings, the best model was XGBoost and the best predicted target variable was raw speed with an R^2 score of 0.795. It also demonstrates that velocity prediction along altitude profiles is achievable, but only when terrain-derived features are included in the model. Purely spatial data does not provide enough information on its own.

5.1.4 Statistics of the best speed model

As explained in the experimental setup section, the best model in combination with the best target variable is selected for the next stage in the prediction pipeline. In this case, the raw speed variable combined with XGBoost resulted in the best metrics. This was further substantiated by the significance tests across the metrics all being significant, and the effect sizes being constantly large. This subsection focuses on an analysis of this specific configuration to understand which features were the most influential and how they contribute to the model's overall accuracy.

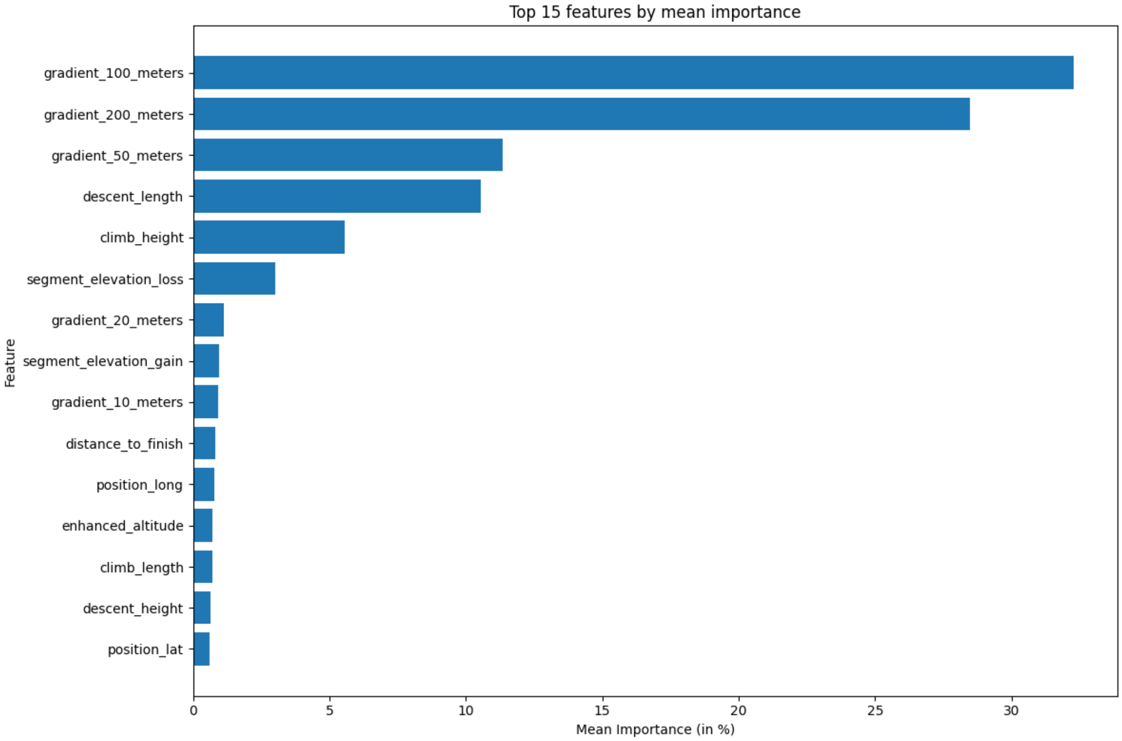

Figure 7: Importance of the features created

Figure 7 shows the feature importance of the best model. The most influential predictors are the gradient-based features, with `gradient_100_meters`, `gradient_200_meters`, and `gradient_50_meters`. These results align with the expectations, changes in gradient have a direct effect on the rider's velocity, with uphill sections reducing the speed of the rider due to gravitational resistance, and downhill sections increasing speed due to acceleration. temporal aggregation of gradient features has the highest importance in predicting the rider's speed. This result is logical, where changes in road gradient heavily influence the cyclist's velocity, due to the increased resistance when climbing and acceleration when descending.

There are also features with very limited importance, for example `longitude` and `latitude`, which almost have no effect in the predictive outcome. The latitude and longitude combined describe the rider's geographical position, hence it is a logical outcome since the model aims to predict speed. These results confirm that terrain-derived features, especially slope information, are essential for accurate speed prediction, while the geographical location carries little predictive value.

Last but not least, the two models were compared by evaluating the best-performing speed representations, which corresponded to the raw speed, rather than smoothed rolling averages.

Metric	**XGBoost**	**Random Forest**	**Difference**	**p-value**	**Cohen's d**
RMSE (km/h)	6.37 ± 0.70	7.52 ± 0.57	-1.15 ± 0.46	$2.11 \cdot 10^{-14}$	-2.49
MAE (km/h)	4.61 ± 0.54	5.59 ± 0.49	-0.99 ± 0.33	$1.90 \cdot 10^{-16}$	-2.98
R^2	0.795 ± 0.110	0.722 ± 0.119	+0.072 ± 0.035	$2.29 \cdot 10^{-12}$	2.07

Table 5: Statistical Comparison of Speed Prediction Models

Within Table 5, the paired t-test results demonstrate that XGBoost significantly outperforms Random Forest for speed prediction across all evaluation metrics, with $p < 0.05$ for all comparisons. The effect sizes are consistently large, indicating not only statistical significance but also practical importance of the performance differences.

5.2 Power Prediction

In this Section, we again experiment with the XGBoost and Random Forest algorithms by predicting the power output over increasing window sizes (raw, 5, 10, 25, 50 seconds) to test whether smoothing improves prediction performance. As mentioned in Section 4.3.3, the best-predicted speed variable was taken as an additional feature besides **Representation 2**. In the previous experiment of speed in Section 5.1, we found that the best predicted variable was the raw speed, predicted by the XGBoost model.

5.2.1 Base power prediction models

These base results indicate that predicting the power output only with the features of **Representation 1** is not possible. Both models score negative R^2 scores when trying to predict each individual rolling average, meaning that their predictions are worse than simply using the mean power value as a baseline. There are some small improvements as the window size increases, likely due to the smoothing effect filtering out short-term noise.

Target	Mean RMSE	Mean MAE	Mean R^2
Raw Power	180.38 ± 20.94	143.99 ± 20.10	-0.034 ± 0.055
Average Power 5	164.70 ± 18.85	130.53 ± 18.12	-0.079 ± 0.060
Average Power 10	150.25 ± 17.22	119.27 ± 16.45	-0.114 ± 0.065
Average Power 25	128.29 ± 15.01	102.34 ± 14.23	-0.126 ± 0.068
Average Power 50	112.52 ± 13.15	90.09 ± 12.47	-0.078 ± 0.066

Table 6: Random Forest Power Prediction Performance Across Different Averaging Windows

Target	Mean RMSE	Mean MAE	Mean R^2
Raw Power	183.39 ± 22.63	146.26 ± 21.19	-0.070 ± 0.108
Average Power 5	167.43 ± 24.22	132.63 ± 19.72	-0.118 ± 0.318
Average Power 10	153.90 ± 27.20	121.96 ± 19.45	-0.181 ± 0.558
Average Power 25	133.18 ± 25.71	105.61 ± 17.21	-0.190 ± 0.498
Average Power 50	117.94 ± 16.25	93.96 ± 13.07	-0.206 ± 0.420

Table 7: XGBoost Power Prediction Performance Across Different Averaging Windows

5.2.2 Enriched power model performance

Once **Representation 2** and the predicted speed variable were added, the model's performance improved significantly

Window Size	RMSE (W)	MAE (W)	R^2
Raw Power (no avg)	154.41 ± 23.23	114.57 ± 20.17	0.244 ± 0.093
5-second avg	133.29 ± 18.79	98.80 ± 16.07	0.300 ± 0.105
10-second avg	115.62 ± 15.79	86.96 ± 13.50	0.351 ± 0.119
25-second avg	94.24 ± 13.94	72.76 ± 11.88	0.394 ± 0.161
50-second avg	82.92 ± 10.93	65.02 ± 8.53	0.400 ± 0.167

Table 8: Prediction Performance power XGBoost

Window Size	RMSE (W)	MAE (W)	R^2
Raw Power (no avg)	153.63 ± 23.08	113.24 ± 19.91	0.253 ± 0.083
5-second avg	132.56 ± 17.88	97.60 ± 15.15	0.309 ± 0.088
10-second avg	115.38 ± 14.68	86.27 ± 12.61	0.355 ± 0.095
25-second avg	92.74 ± 11.73	71.06 ± 10.00	0.418 ± 0.114
50-second avg	82.19 ± 10.12	63.97 ± 8.02	0.418 ± 0.126

Table 9: Prediction Performance Power RandomForest

For the XGBoost model, the RMSE decreases from 154.41 ± 23.23 watts to 82.92 ± 10.93 watts. Similarly for the Random Forest model, the RMSE decreases from 153.63 ± 23.08 to 82.19 ± 10.12. The same trend happened with MAE for both models, where the values dropped substantially when the rolling window increases. The pattern seen in both models suggests that smoothing the power data through rolling averages helps the algorithm capture trends better. The standard deviation also decreases, which suggests better model stability and less sensitivity to noise. The decreases in both RMSE and MAE support the expectation, because opwer output data is noisy due to sudden bursts of power and various physiological and environmental factors during the race. Smoothing power reduces noise and random fluctuations that can obscure the underlying physiological patterns.

5.2.3 Evaluating of Representation 1 and Representation 2

There is a significant difference in the predictive performance when moving from **Representation 1** to **Representation 2**, which includes terrain-derived features and predicted speed. The results highlight the importance of enriched features in capturing the relationship between terrain context and predicted speed to predict power output.

The baseline model, trained on **Representation 1**, has no predictive capabilities, with R^2 values below 0 across all window sizes and both models (R^2 = -0.034 to -0.2). In contrast, the enriched models based on **Representation 2** and predicted speed demonstrated a meaningful relationship between input features and target value, resulting in improved prediction accuracy (R^2 = 0.24 to 0.42).

When comparing the two machine learning algorithms, both XGBoost and Random Forest (RF) performed similarly in terms of trend and overall structure. However, Random Forest achieved marginally better results across all window sizes. The optimal window length here was reached at 50-seconds for both models, where the RF reached a slightly lower RMSE and MAE (82.19 W vs 82.92 W), (63.97 W vs 65.02 W), and a higher R^2 (0.418 vs 0.4) compared to XGBoost. Interestingly, the R^2 of the RF for the rolling average power of 25 and 50 seconds stays the same, suggesting that the optimal level of smoothing was reached. This could mean that there was enough noise filtered out without losing too much meaningful variation in the power signal.

Lastly, the best-performing target variables of both models are compared to see if there is a significant difference between the models.

Table 10: Statistical Comparison of Power Prediction Models using 50-second Averaged Power Target

Metric	**XGBoost**	**Random Forest**	**Difference**	**p-value**	**Cohen's d**
RMSE (W)	82.92 ± 10.93	82.19 ± 10.12	0.73 ± 10.52	0.261	0.07
MAE (W)	65.02 ± 8.53	63.97 ± 8.02	1.05 ± 8.27	0.158	0.13
R^2	0.400 ± 0.167	0.418 ± 0.126	-0.018 ± 0.146	0.205	-0.12

RMSE is marginally lower with a reduction of 0.73 W, which does not result in a statistically significant difference (p = 0.261 and Cohen's d = 0.07). The negligible effect size also indicates no meaningful difference between the algorithms. The MAE has a slightly higher difference than the RMSE; however, this also lacks statistical significance (p = 0.158 and a Cohen's d = 0.13). The R^2 value of Random Forest has a 1.8% better score, but not enough for it to be statistically significant (p = 0.205 and Cohen's d = -0.12).

Unlike the speed prediction, where the XGBoost demonstrated to be the better model, power prediction reveals approximately even results between the best-performing target values of Random Forest and XGBoost. All effect sizes fall below 0.2, classifying as negligible, and the p-values are all significantly above 0.05.

5.2.4 Best power model

While the paired t-test analysis revealed no significant differences between XGBoost and Random Forest for a 50-second average power prediction, the small advantages demonstrated by Random Forest across all evaluation metrics determined the model decision.

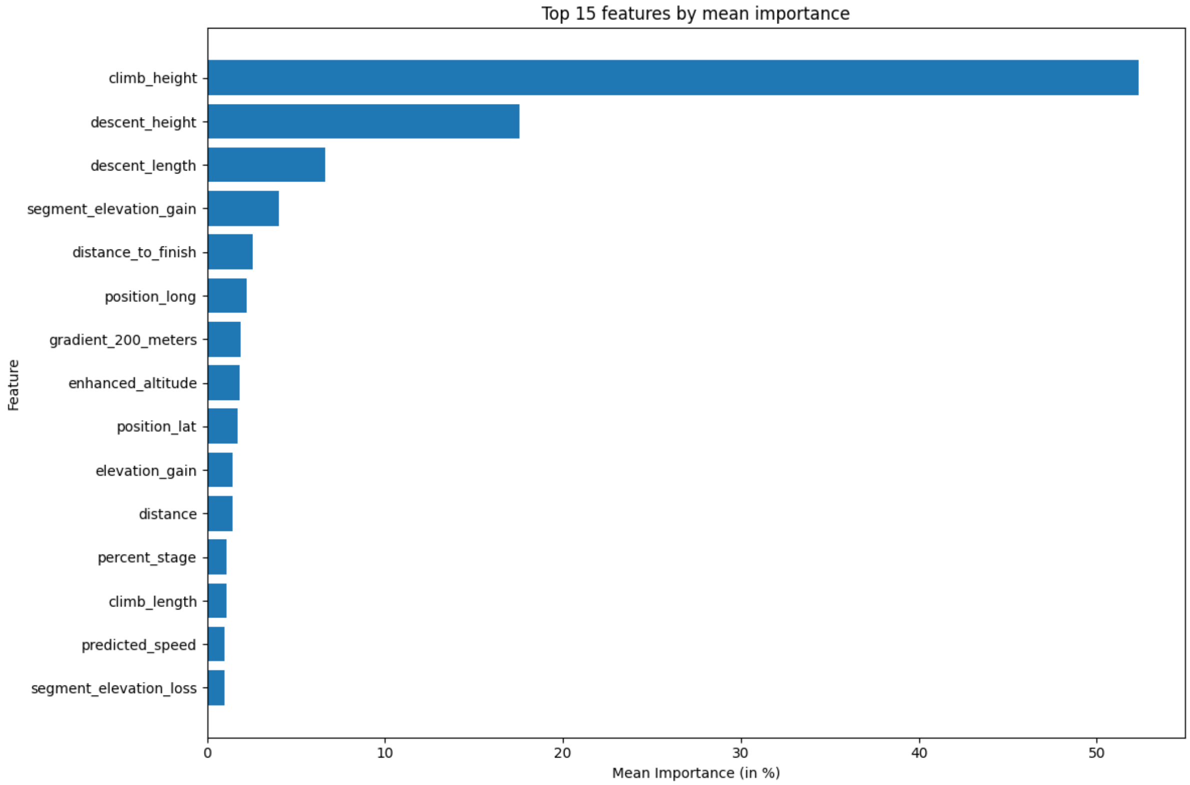

Figure 8: Power feature importance

As visualized in Figure 8, the climb height and descent height variables contribute the most to the prediction of power, while the predicted speed does not contribute as much. The importance of the height of climb and descent effectively captured the workings of the human body. As mentioned in Section 2.1, the gravitational force becomes the dominant force to overcome during ascents. The higher the gravitational force, the more power the rider has to output to maintain speed. The opposite happens when descending, where the gravitational force helps the riders gain speed without pedaling.

Interestingly, the predicted speed has minimal contribution to the prediction of the power, suggesting a non-linear relationship between the two variables. Environmental factors such as gradient capture the relationship more accurately, also confirmed in the already mentioned Martin et al. (1998) paper [MMC+98].

5.3 Feature Engineering and Model Enhancement Results

5.3.1 Baseline Classification

This section evaluates the performance of four classification algorithms trained using only features from **Representation 1**. The features were chosen to establish a simple baseline, from which the impact of the more advanced feature representation can be later measured.

Baseline Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
LightGBM	0.7954	0.5764	0.5211	0.5473	0.7921
Logistic Regression	0.8249	0.6987	0.4611	0.5555	0.8025
Random Forest	0.8194	0.7087	0.4058	0.5161	0.7675
XGBoost	0.7954	0.5812	0.4937	0.5339	0.7992

Table 11: Baseline Drop-off Prediction Model Performance

The results in Table 11 show that all models achieve similar overall accuracy scores, ranging from approximately 82% to 84%. As stated in 4.2.1, the classifiers should all score higher than 73.97%, meaning that they all would outperform a majority class classifier with accuracy alone and have learned some real patterns. However, accuracy alone is not a reliable metric for this task due to the class imbalance of certain races in the dataset. Therefore metrics such as precision, recall F1 score and ROC AUC are also considered.

A noticeable trend across all models is the trade-off between precision and recall. Most models are favoring precision, meaning they correctly predict drop-off events only when highly confident, but at the cost of missing actual drop-off events. The highest recall is achieved by XGBoost with 48.95%, while the highest precision is obtained by RF 75.63%. The F1 scores represent the balance between precision and recall and range from 53% to 58%, indicating an average overall performance in detecting drop-off points. Last but not least, the ROC-AUC values, which reflect the model's ability to rank predictions correctly, range from 77.5% to 80.0%. This confirms that the models perform better than random chance and can distinguish between two classes to a reasonable degree. All in all, the models can find some patterns with only features from **Representation 1**, but there is room for improvement.

5.3.2 Feature Importance Baseline Model

To understand which **Representation 1** features contribute the most to the baseline model performance, feature importance analysis was conducted across all four classification algorithms. Figure 9 presents the heatmap showing the relative importance of each feature across different models.

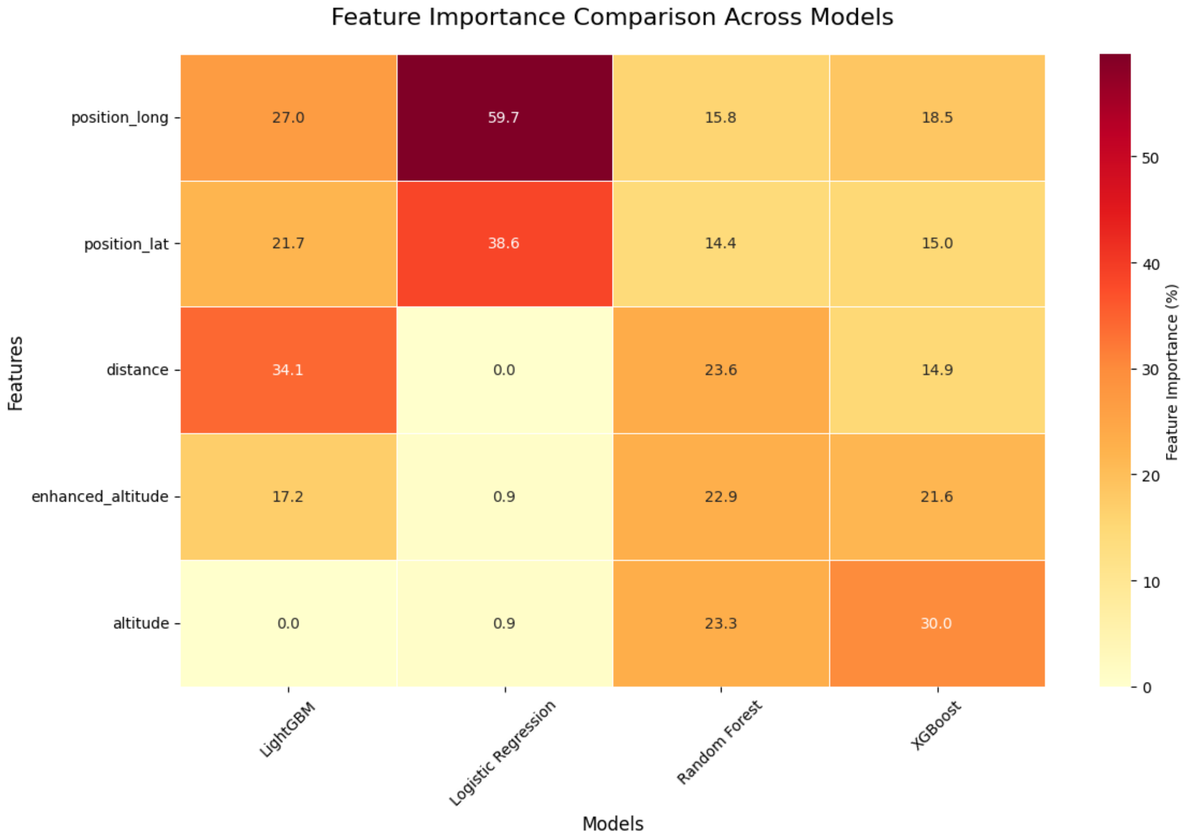

Figure 9: Base classification heatmap

The analysis reveals patterns in feature utilization across the different models. The geographical features emerge as the most important feature across the models, with Logistic Regression being almost solely dependent on the geographical position. These findings show that the models, especially Logistic Regression (59.7% latitude and 38.6% longitude), are too dependent on the geographical location and reveal a critical point in the dataset. The dataset contains multiple editions of identical race routes from different years, creating a scenario where models learn to memorize specific geographical locations rather than understand the underlying mechanisms of cyclist drop-off events.

Positive signs emerge when analyzing Random Forest and XGBoost, where distance and enhanced altitude influence the decision for drop-off points the most. This aligns more closely with cycling performance compared to geographically dependent models. For Random Forest, the enhanced altitude (22.9%) and distance (23.6%) serve as primary predictors, while having some influence because of geographical features. Having a bit of geographical influence is not a problem, because race routes are often repeated annually or with minor modifications.

The current five-feature baseline demonstrates that limited feature diversity forces models towards geographical memorization. However, both XGBoost and Random Forest indicate that generalizable prediction models can be made. Expanding the next three representations with terrain-derived and physiologically derived features should theoretically shift the decisions made more towards a terrain-based feature importance system.

5.3.3 Terrain Feature Enhancement

Building upon the baseline performance established in Section 5.3.1, this section evaluates the impact of adding **Representation 2** features, an enriched feature set that includes derived terrain features from **Representation 1**.

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
LightGBM	0.8269	0.6356	0.6351	0.6353	0.8574
Logistic Regression	0.8683	0.7587	0.6530	0.7019	0.8999
Random Forest	0.8574	0.7775	0.5594	0.6507	0.8594
XGBoost	0.8267	0.6375	0.6263	0.6318	0.8515

Table 12: Enhanced Model Performance with Terrain-Derived Features

The enriched feature models improve drastically across all evaluation metrics compared to the baseline results in Table 5.3.1. The most substantial improvements observed are recall and F1 score, indicating better detection of actual drop points.

The Logistic Regressor has improved the most with each metric improving by 4% up to 21%. The 21% improvement in recall is particularly significant as it indicates the model's ability to identify actual drop-off events that were previously missed. This suggests that terrain-derived features provide crucial information about the moment when the rider could drop off.

Other models also consistently improved, with all the models outscoring the baseline model. Recall was also the feature that improved the most for these models, with improvements ranging from 17% (LightGBM) to 10% (Random Forest). This strengthens the basis that adding terrain-derived features improves the prediction capabilities of predicting real drop points.

5.3.4 Feature importance terrain-enriched

To understand which terrain-derived features contribute most to the drop-off points performance, an enlarged set of features was analyzed across the same four models to compare the results.
Position-based features demonstrate the highest predictive power, with the latitude and longitude remaining the most influential variables across the model. However, their importance has decreased compared to **Representation 1**, suggesting that the additional terrain features provide alternative insights for prediction. The LightGBM and Logistic Regression still show geographical dependence, with latitude contributing 18% (LightGBM) and 32% (Logistic Regression), and longitude 18% (LightGBM) and 23%. This in combination with the feature set of **Representation 1**, shows these

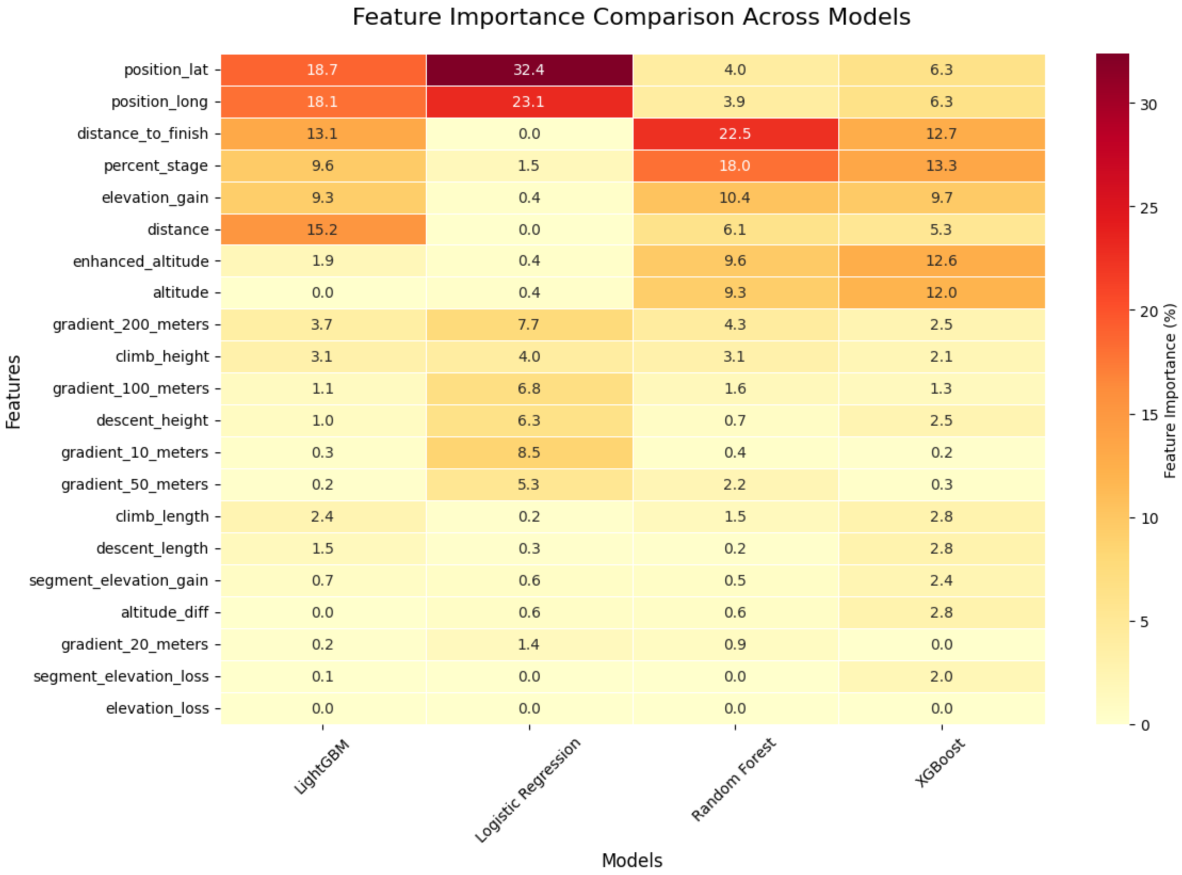

Figure 10: Enriched feature comparison with **Representation 2**

models rely on memorized spatial patterns from repeated race routes.

Distance-related metrics also show significant importance, particularly distance to finish (13% in LightGMB, 22.5% in Random Forest and 12% in XGBoost). The percentage of the stage is also influential, contributing 18% to Random Forest and 13% to XGBoost. This indicates that fatigue accumulation, without knowing any physiological factors yet, determines when the performance of the SR decreases. Interestingly, the logistic regressor is not influenced at all by the distance-related metrics, potentially indicating that it heavily relies on learned spatial patterns from the training data rather than generalizable fatigue dynamics. This suggests that the logistic regression approach may be overfitting to specific geographical locations and stage characteristics seen during the races, making it less capable of predicting the outcome of new races.

Other features include elevation ones, with elevation gain and enhanced altitude contributing meaningfully to predictions. Gradient-based features demonstrate model-specific preferences, with Logistic Regression relying the most on the gradients. This suggests that different models capture distinct aspects of terrain difficulties, with linear models focusing on immediate slope changes while ensemble models (XGBoost and Random Forest) include more complex terrain interactions.

5.3.5 Predictive Feature Integration

The integration of **Representation 3** alongside the existing terrain-derived features offers only marginal improvements compared to **Representation 2**. The performance gains are minimal across all algorithms, with most metrics showing negligible changes or even slight decreases in some cases.

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
LightGBM	0.8254	0.6334	0.6280	0.6307	0.8553
Logistic Regression	0.8722	0.7712	0.6562	0.7090	0.8994
Random Forest	0.8590	0.7923	0.5500	0.6493	0.8622
XGBoost	0.8215	0.6251	0.6193	0.6222	0.8482

Table 13: Predictive feature integration

Compared to Table 12, the performance comparison shows that the predicted power and speed characteristics provide a limited additional predictive value beyond what is already captured by the terrain characteristics.

All the models show marginal improvement or a slight decrease in performance. The gains made by the models are within typical statistical noise margins and may not give the models a significant improvement. In fact, for some models, the predicted features may introduce noise rather than a meaningful signal.

The limited impact of predicted features suggests that terrain-derived characteristics already have the majority of relevant information for drop-off prediction. GPS-derived features appear to effectively simulate the physical demands the rider would go through, which would otherwise be represented through the power and speed measurements.

This finding could have important practical implications if the convoluted data also has no impact on the performance of the model for real-world scenarios. The results demonstrate that effective drop-off can be achieved using available GPS and elevation data without requiring the building of sophisticated power estimation. This reduces the system's complexity while maintaining the same predictive performance.

5.3.6 Predictive Feature Performance and Model Selection

To show the impact the predicted speed and power have on the model's performance, a new feature importance plot has been made.

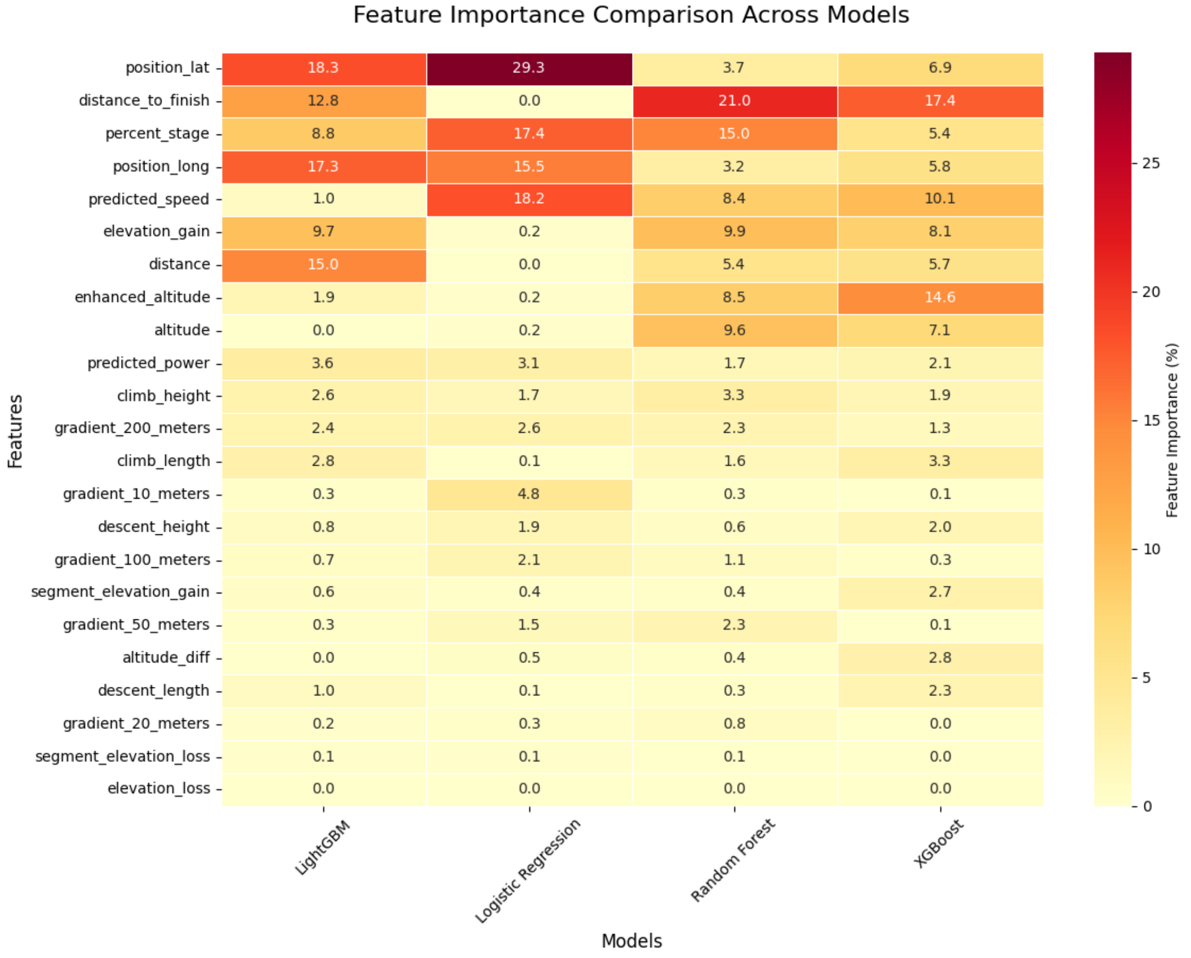

Figure 11: Feature importance of **Representation 3**

Contrary to the marginal improvements and slight decreases, the predicted speed and power features demonstrate meaningful influences over terrain-derived features alone, with performance gains varying across the different models. As shown in the feature heatmap, the predicted speed shows a considerable importance ranging from 1% to 18%, while the predicted power feature contributes

1% to 4%. Although terrain-derived characteristics such as position latitude, distance to finish, and percentage stage continue to dominate feature importance rankings, predicted speed and power characteristics provide physiological information that improves the model by adding more than geographic characteristics as input.

5.3.7 Workload Feature Integration

Random Forest is chosen to decide which model will be used for the final comparison and grid search optimization of convolution parameters (alpha and window size) due to its geographical independence and great handling of feature interaction. Unlike the Logistic Regressor, which achieves higher scores overall, the Random Forest demonstrates a more distributed feature utilization pattern, making it less susceptible to geographical biases that could compromise generalizability across different race locations and terrain types. The ensemble nature of Random Forest allows it to effectively capture non-linear relationships between both predicted physiological metrics and drop-off patterns. The geographical robustness and focusing mostly on terrain-derived features makes Random Forest the ideal candidate for exploring optimal convolution parameters, as the model will be more likely to generalize to new race environments and terrain conditions. These characteristics make the Random Forest future-proof for newly added stages.

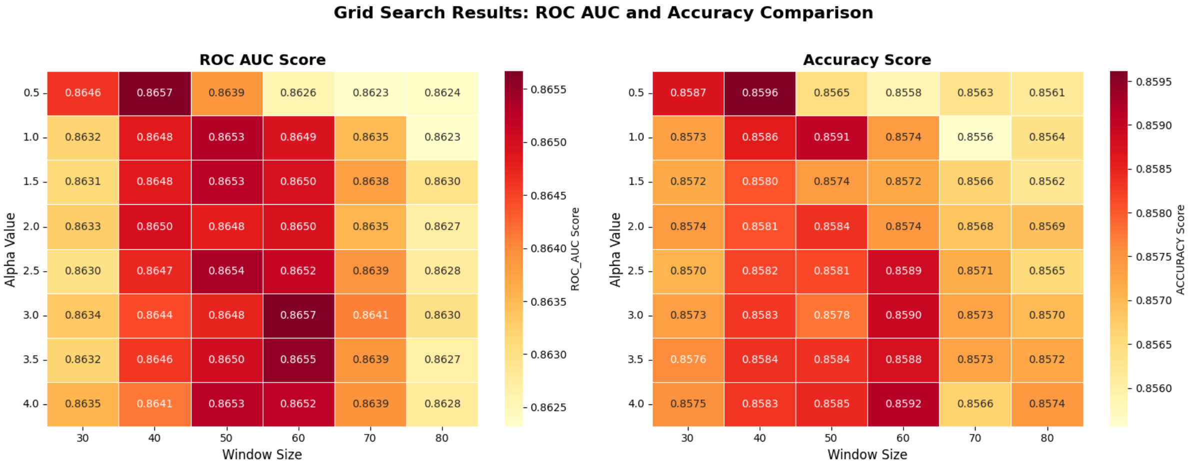

Figure 12: Grid search optimization alpha and window size

The grid search optimization, as shown in Figure 12, reveals that a window size of 40 minutes with an alpha of 0.5 achieves the optimal performance, delivering the highest accuracy of 85.96% and ROC AUC of 86.57% among all tested parameter combinations. The heatmap also makes it clearly visible by making both the grids dark red, indicating the peak performance.

The results show a distinct pattern across the parameter grid. Window sizes of 40-50 minutes outperform smaller or larger windows, suggesting that a moderate window size captures the most relevant historical information for drop-off prediction without introducing noise from distant past events. The alpha parameter remains stable across its range, with variations typically within 0.1-0.2 percentage points.

5.4 Final Drop-off Model

To conclude the feature engineering and model evaluation pipeline, the final Random Forest model was trained using the complete set of engineered features. This configuration was made to achieve the optimal dropout prediction capability. The final model performance metrics are as follows:

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Random Forest	0.8596	0.7904	0.5560	0.6528	0.8657

Table 14: Drop-off Model Performance Results

Based on the Table 14, the Random Forest model demonstrates strong predictive capabilities, achieving an accuracy of 85.96% and an ROC-AUC of 86.57%. The model has a precision of 79.04%, meaning that when it predicts a drop-off point, it is correct approximately 79% of the time. However, the recall of 55.6% states that the model identifies just over half of all actual drop-off points, which is a moderate score, resulting in a balanced F1-score of 65.28%.

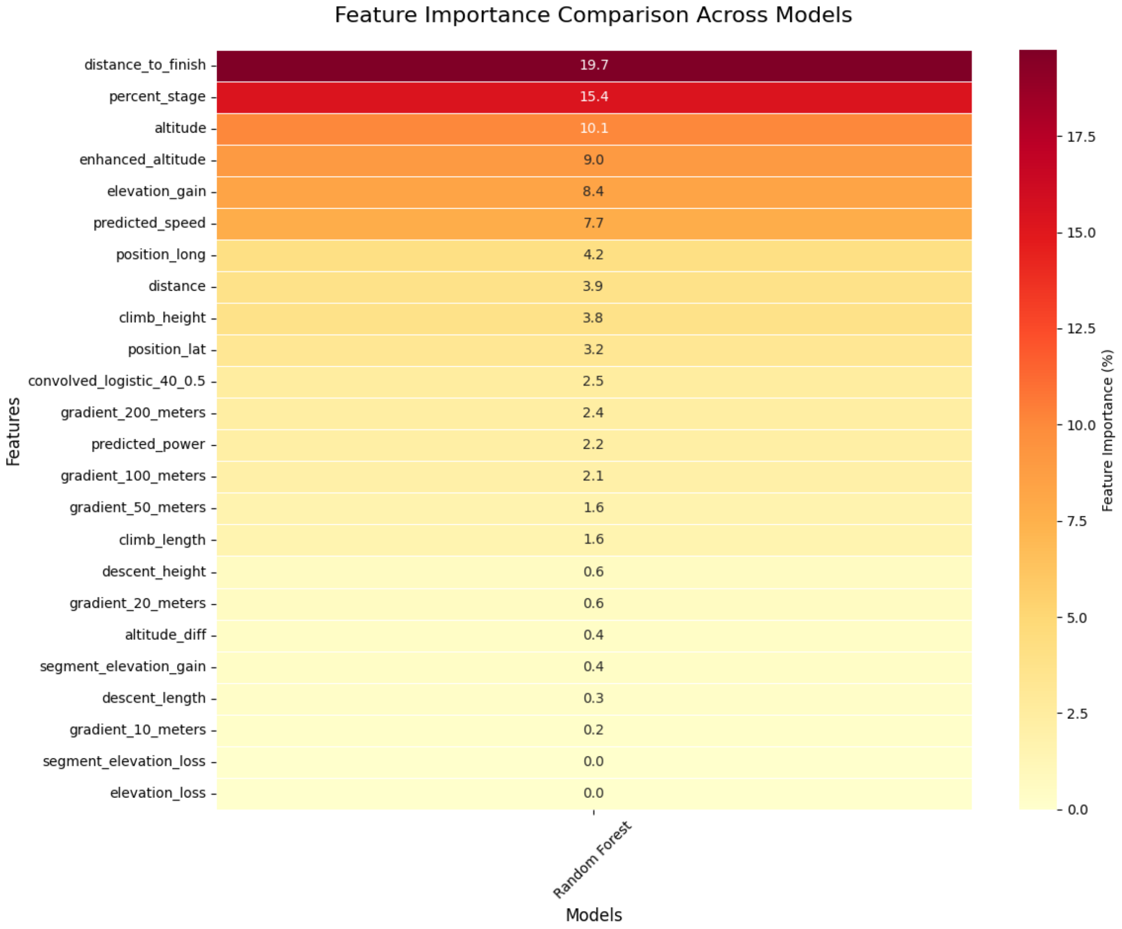

Figure 13: Final drop-off model

The feature importance analysis shows the most important predictors, with distance to finish (19.7%) and percent stage (15.4%) being the most dominant ones. The findings align with cycling knowledge, where the closer the rider is to the finish, the higher the chances are of him dropping off. Terrain-derived features such as altitude (10.1%), enhanced altitude (9%), and elevation gain (8.4%) also contribute to the model's power. These features are also logical for contributing to the drop-off points, because the amount of elevation gain and the number of climbs in the race are fundamental factors in the cyclist's fatigue.

The final results demonstrate that predicting the drop-off points is indeed feasible using GPS data. The model's high accuracy and ROC-AUC scores indicate a strong ability to filter out the drop-off and non-drop-off points. While the moderate recall shows that some drop-off points remain undetected, the performance of the model is sufficient for practical applications, where the coaches only need to know if the rider will drop off or not. The application could be used to support race planning, where the rider could be used as a domestique or as the main man for the race.

5.5 Has enrichment of features any impact on model performance?

The paired t-test results reveal key findings regarding the research subquestion about how the original multivariate time-series can be enriched with features to improve the final model.

Comparison	**p-value**	**Cohen's d**	**Significant**
Representation 1 vs **Representation 2**	0.009	-0.51	Yes
Representation 2 vs **Representation 3**	0.371	0.17	No
Representation 3 vs **Representation 4**	0.623	-0.09	No
Representation 1 vs **Representation 4**	0.021	-0.44	Yes

Table 15: Feature Set Comparison Results - Accuracy (Random Forest)

Comparison	**p-value**	**Cohen's d**	**Significant**
Representation 1 vs **Representation 2**	0.009	-0.60	Yes
Representation 2 vs **Representation 3**	0.082	0.38	No
Representation 3 vs **Representation 4**	0.722	0.08	No
Representation 1 vs **Representation 4**	0.006	-0.64	Yes

Table 16: Feature Set Comparison Results - ROC AUC (Random Forest)

The most substantial improvements occur from **Representation 1** to **Representation 2**. This enrichment shows statistically significant improvements in accuracy (p = 0.009, Cohen's d = -0.51) and ROC (p = 0.009 and Cohen's d = -0.60), confirming that terrain-derived features improve the model. The negative Cohen's d values indicate that **Representation 2** consistently outperforms **Representation 1**, suggesting meaningful improvements beyond statistical noise. Similarly, **Representation 4** is also significantly better to **Representation 1** (p = 0.021, Cohen's d = -0.44) and ROC AUC (p = 0.006, Cohen's d = -0.64).

The comparison between **Representation 2** and **Representation 3** reveals no significant improvement in accuracy (p = 0.371, Cohen's d = 0.17) or ROC AUC (p = 0.082, Cohen's d = 0.38), despite the addition of the predicted speed and power variables. This suggests that the terrain-derived features contain enough essential information for dropout prediction. Furthermore, the addition of convoluted predicted power in the comparison between **Representation 3** and **Representation 4** also shows negligible improvements with non-significant p-values (0.623 and 0.722) and (Cohen's d = -0.09 for accuracy, 0.08 for ROC AUC), indicating that convolution features don't provide benefits.

The enriched dataset significantly impacts the model performance; however, the complex additions of predicted speed, power, and convoluted power have minimal improvements. **Representation 2**, terrain-derived features, represents the best balance between complexity and performance gains, providing significant improvements without adding more sophisticated approaches. These findings are in line with reality, where the profile of the race matters the most.

6 Predictions for a Feature Race

To better understand how the pipeline works, we tested the implementation in a real professional race, the Paris-Nice Tour. The selection of these races is based on a race where the SR has a high chance of dropping off and a race where the rider will most likely contest for victory.

First, we start by analyzing the Paris-Nice stage, an example of a hilly stage where it would be difficult to predict if the rider would drop off due to terrain difficulty. The altitude profile of this race is seen in Figure 14. The analysis followed the complete pipeline methodology outlined in the previous sections.

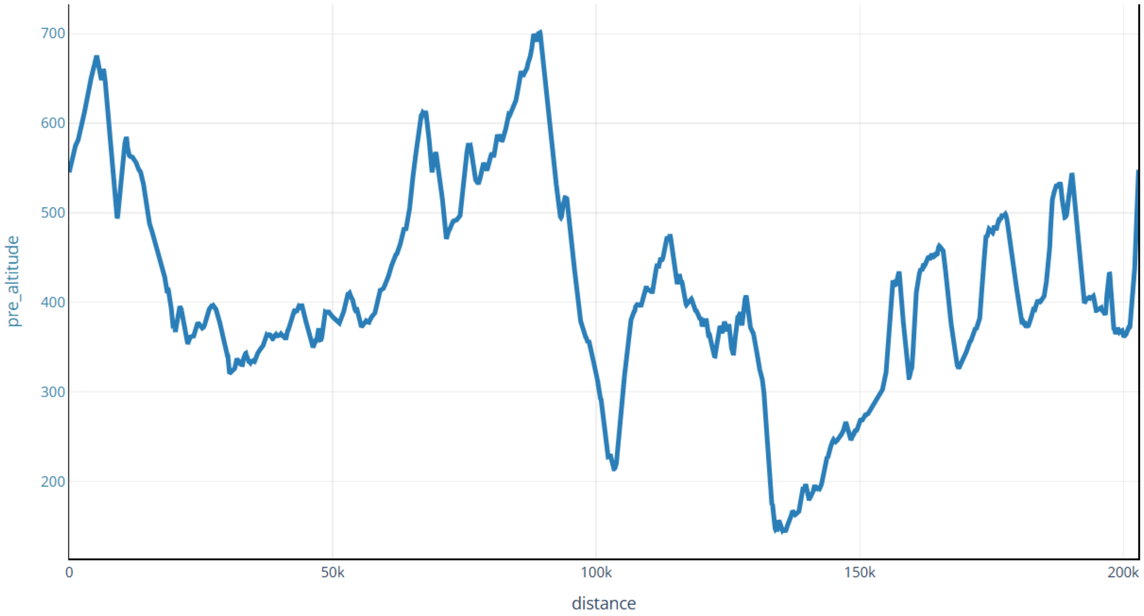

Figure 14: Altitude profile Paris Nice

The raw data underwent cleaning and feature engineering as described in Section 3.2.4. The cleaned dataset contained all the core features of **Representation 1**. The XGBoost model was chosen to do the speed predictions, generating them across the entire route as illustrated in Figure 15. What can be seen when comparing the altitude and speed profiles is notable speed reductions when the SR is climbing and increases during descent segments. The model aligns with expected cycling performance patterns, showing decreased velocities in steep segments and increased velocities on descents.

Subsequently, the power model estimated the SR's power output during the race, shown in Figure 16. The power profile also matched characteristic cycling patterns. During climbing, the power output was higher, and when descending, the rider was recovering, showing lower outputs.

The last added feature in the pipeline was the convolution, using the optimal parameters determined in Section 5.3.7. The convoluted power representation describes the fatigue effects and is shown in Figure 17. At the beginning of the graph, the window length is smaller than the actual data, meaning that the points are exaggerated. After the window length has been reached, the convolution

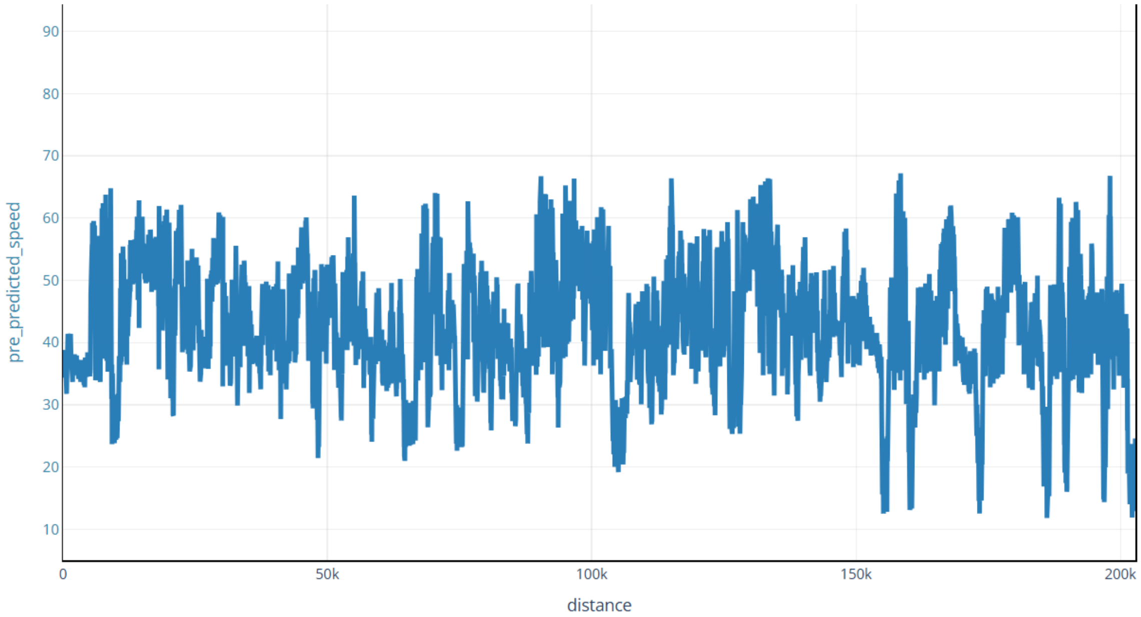

Figure 15: Speed prediction of stage in Paris Nice

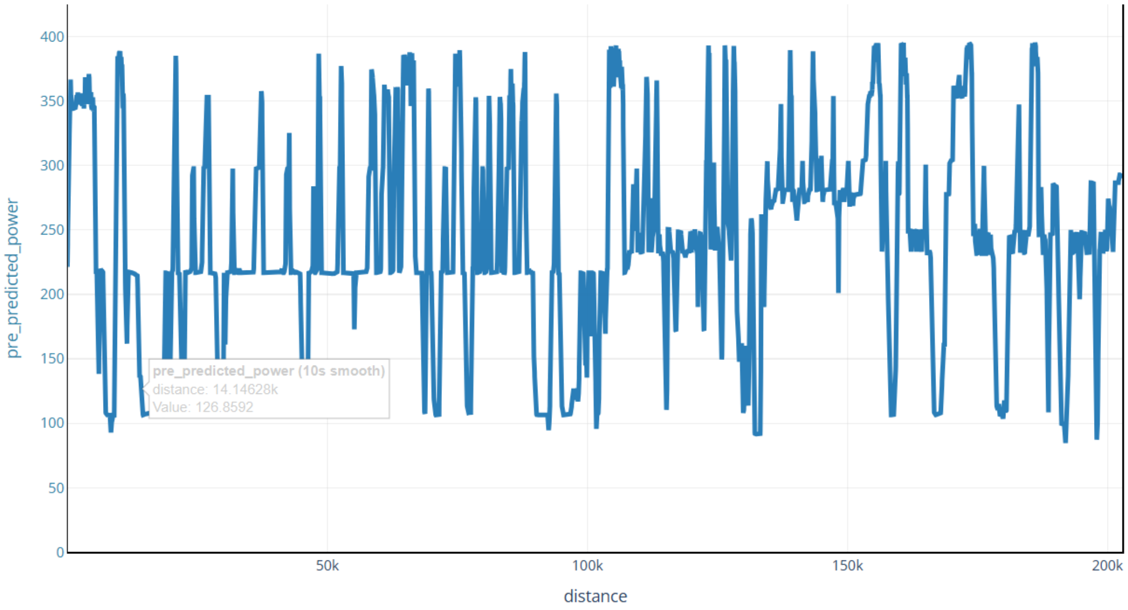

Figure 16: Predicted power of stage in Paris-Nice

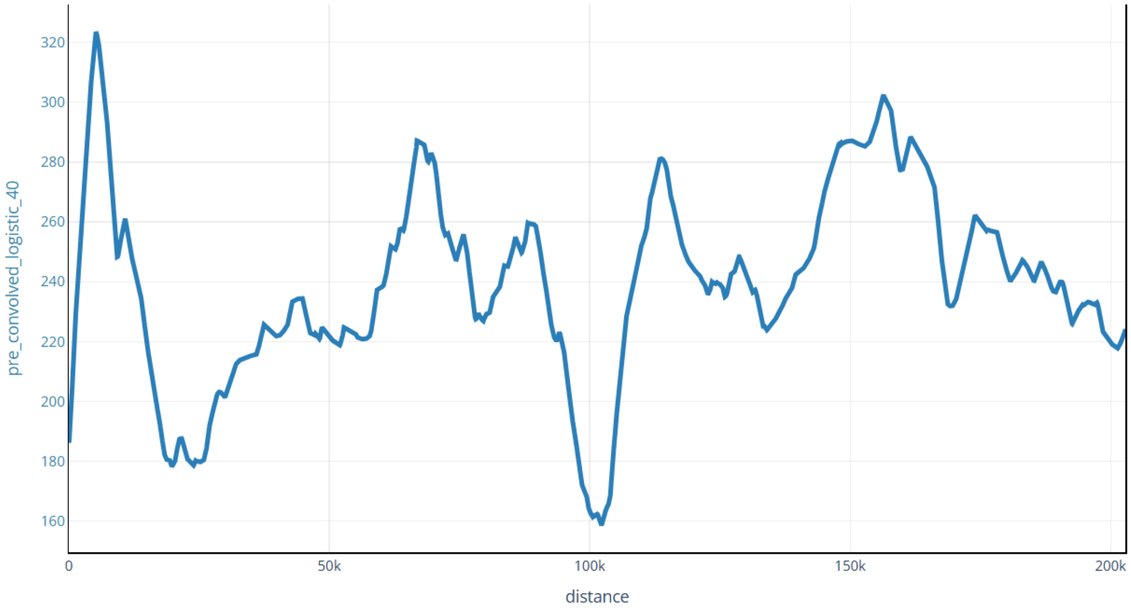

Figure 17: Convoluted data of the stage in Paris Nice

is more representative.

The final stage of the pipeline generates the drop-off probability with all available features, as shown in Figure 18. The visualization employs colors, with dark red indicating a high chance of dropping off and dark blue representing a low probability. In the analyzed graph, around kilometer 160, the algorithm senses that the SR is experiencing difficulties keeping the same pace as the other riders.

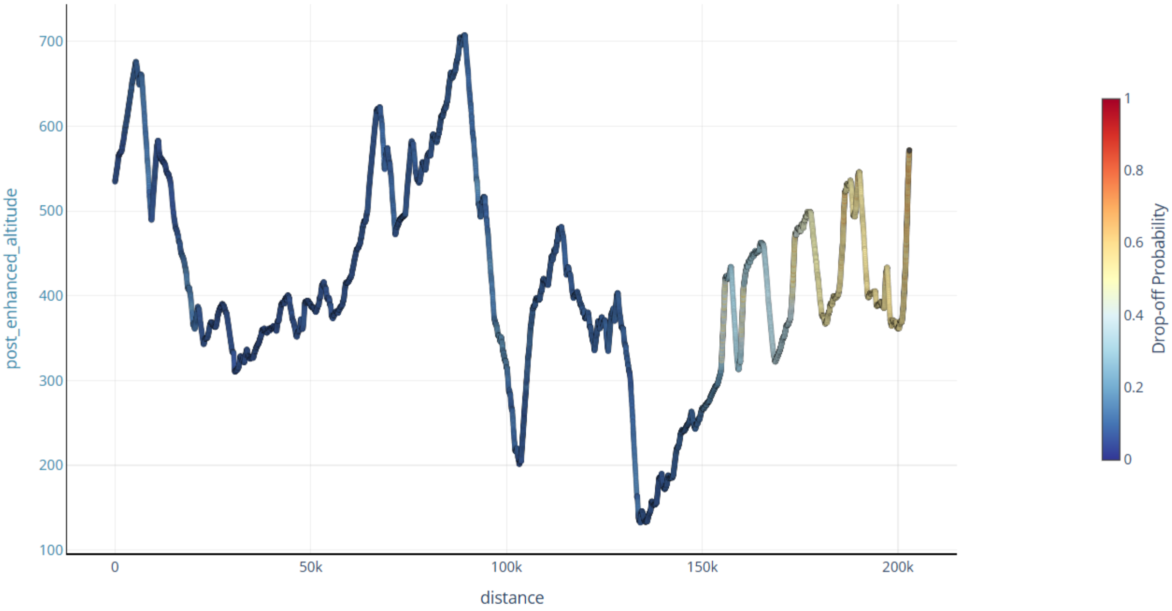

Figure 18: Drop-off prediction of the Paris Nice stage

To visualize the outcome of the race as validation, Figure 19 shows the final result. The SR could keep up with the FR until kilometer 185, which corresponded to a predicted drop-off probability of 50%.

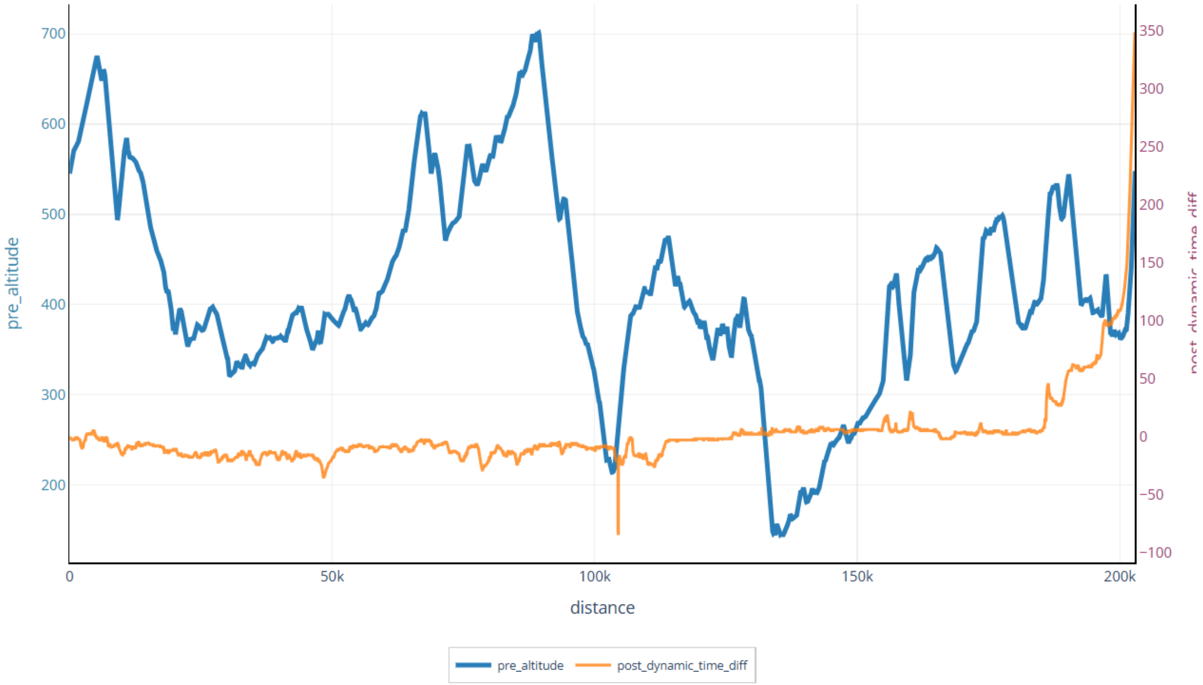

Figure 19: Post race result Paris Nice

7 Discussion and Conclusion

This research demonstrates the process of predicting drop-off points in professional cycling races using a machine learning pipeline. By combining the data of real professional cyclists, feature engineering, cross-validation, and a combination of regression and classification models, the developed model can identify moments of significant performance decline of our SR with reasonable accuracy.
The pipeline approach started with collecting all races that could give insights into when the SR could drop off. The data of these races were collected, and the noisy ones were filtered out. With synchronization and a dynamic time algorithm, the gap between the SR and FR was calculated to see the drop-off points in the collected races. The next step in the process was to featurize the stages, adding new features such as spatial, temporal, elevation, climb and descent, and gradient features. The features allowed us to predict the outcomes of the speed, power output, and drop-off points more accurately. After feature extraction, the speed and power outputs were predicted with regression models using a cross-validation method. The method maximized the information that could be retrieved from the small dataset. By using root mean square error (RMSE), mean square error (MAE), and R^2 we could determine the accuracy and performance of the models. The predicted speed and power output are integrated in all the featurized tables and used for the final drop-off model. This model used classification methods to predict the chance that the SR would drop at each point in the stage. With metrics such as precision, recall, F1 score, and ROC AUC,

the study provides a detailed understanding of the models' strengths and weaknesses. The pipeline could become important in practical applications, where knowing when the SR would or would not drop could be essential in the finalized plan of the team.
In conclusion, this research establishes a foundation for the automated prediction of drop-off points in cycling, demonstrating a machine learning pipeline that could provide interesting insights in upcoming cycling races. The approach has the potential to extend into other endurance sports and to support data-driven decision-making in a professional athletic setting.

7.1 Limitations

The framework created in this research for finding the drop-off points during a certain race also has its limitations.
First, environmental factors, such as wind, conditions and temperature, are not considered. These factors could determine the speed and power going through different segments of the race. For example, if it has been raining all day, riders are going to apply more caution on descents, decreasing their speed compared to a dry surface.
Second, the models assume that the rider has the same physiological state across all races (not including the fatigue accumulation build-up over a time period).
Third, the dataset is relatively small. With the Leave-one-out validation, the most information has been extracted. However, ideally, the dataset would at least include 50 files. Fourth, besides the data inside the race files, there was no data about events happening during the race. For example, the rider could be feeling sick, or there could have been a crash in front of him, and therefore, he slowed down. If these specific events were denoted, a more robust dataset could be created with only files that are interesting for the model.

To improve new methodologies, future research should address these limitations by expanding their datasets, including environmental variables, and noting down the happenings during the race to further enhance the practical utility of the prediction framework.

7.2 Further Research

For further research, there could be an integration of additional physiological data, such as the lactate threshold or real-time metabolic data, to improve the accuracy and interpretability of drop-off points. Lactate and real-time metabolic data can be crucial in analyzing when the cyclist is fatiguing and cannot keep up with the pace of the riders.
Another interesting topic for future research could be the integration of a real-time prediction based on live data. Developing a real-time system could provide the coaches with more details on whether the SR has any chance of winning the race, because during a professional cycling race, unexpected events can happen. In the best case, this could be integrated into the computer of the riders.
Last but not least, future studies could implement temporal models, such as recurrent neural networks or long short-term memory networks, which could maybe capture dependencies and relationships in cycling data better to enhance the performance of the predictions.
These future studies could help advance the field of professional cycling analytics with more insights for athletes, coaches, and researchers.

References

[AL08] Chris R. Abbiss and Paul B. Laursen. Describing and understanding pacing strategies during athletic competition. *Sports Medicine*, 38(3):239–252, March 2008.

[dHH$^+$20] Arie Willem de Leeuw, Mathieu Heijboer, Mathijs Hofmijster, Stephan van der Zwaard, and Arno Knobbe. Time series regression in professional road cycling. In *Discovery Science*, pages 689–703, Germany, 2020. 23rd International Conference on Discovery Science, DS 2020 ;.

[ELS21] Teun Erp, Robert Lamberts, and Dajo Sanders. Power profile of top 5 results in world tour cycling races. *International Journal of Sports Physiology and Performance*, Accepted for publication, 2021.

[FMB$^+$20] Pedro Forte, Daniel A. Marinho, Tiago M. Barbosa, Pedro Morouço, and Jorge E. Morais. Estimation of an elite road cyclist performance in different positions based on numerical simulations and analytical procedures. *Frontiers in Bioengineering and Biotechnology*, Volume 8 - 2020, 2020.

[JBM23] B. Janssens, M. Bogaert, and M. Maton. Predicting the next pogačar: a data analytical approach to detect young professional cycling talents. *Annals of Operations Research*, 325:557–588, 2023.

[KDSVL20a] Leonid Kholkine, Tom De Schepper, Tim Verdonck, and Steven Latré. A machine learning approach for road cycling race performance prediction. In *Machine Learning and Data Mining for Sports Analytics*, pages 103–112, Cham, 2020. Springer International Publishing.

[KDSVL20b] Leonid Kholkine, Tom De Schepper, Tim Verdonck, and Steven Latré. *A Machine Learning Approach for Road Cycling Race Performance Prediction*, pages 103–112. Kholkine, Leonid and De Schepper, Tom and Verdonck, Tim and Latré, Steven, 12 2020.

[KNH21] Aleksei Karetnikov, Wim Nuijten, and Marwan Hassani. Data-driven support of coaches in professional cycling using race performance prediction. In Pedro Pezarat-Correia, Joao Vilas-Boas, and Jan Cabri, editors, *icSPORTS 2021 - Proceedings of the 9th International Conference on Sport Sciences Research and Technology Support*, pages 43–53. SciTePress Digital Library, 2021.

[KSdL$^+$21] Leonid Kholkine, Thomas Servotte, Arie-Willem de Leeuw, Tom De Schepper, Peter Hellinckx, Tim Verdonck, and Steven Latré. A Learn-to-Rank approach for predicting road cycling race outcomes. *Front. Sports Act. Living*, 3:714107, October 2021.

[MMC$^+$98] James Martin, Douglas Milliken, John Cobb, Kevin McFadden, and Andrew Coggan. Validation of a mathematical model for road cycling power. *Journal of Applied Biomechanics*, 14:276–291, 08 1998.

[OZMR13] Bahadorreza Ofoghi, John Zeleznikow, Clare Macmahon, and Markus Raab. Data mining in elite sports: A review and a framework. *Measurement in Physical Education and Exercise Science*, 17:171–186, 07 2013.

[PBV+16] David C. Poole, Mark Burnley, Anni Vanhatalo, Harry B. Rossiter, and Andrew M. Jones. Critical power: An important fatigue threshold in exercise physiology. *Medicine & Science in Sports & Exercise*, 48(11):2320–2334, 11 2016.

[PLSH+11] Flávio Pires, Adriano Lima-Silva, J Hammond, Emerson Franchini, M Kiss, and Rômulo Bertuzzi. Aerobic profile of climbers during maximal arm test. *International journal of sports medicine*, 32:122–5, 02 2011.

[Ski14] Philip Friere Skiba. *The kinetics of the work capacity above critical power*. University of Exeter (United Kingdom), 2014.

[TBSS93] Hirofumi Tanaka, David Bassett, Tom Swensen, and Renan Sampedro. Aerobic and anaerobic power characteristics of competitive cyclists in the united states cycling federation. *International journal of sports medicine*, 14:334–8, 08 1993.

[vB23] Thijs van Druenen and Bert Blocken. Aerodynamic impact of cycling postures on drafting in single paceline configurations. *Computers Fluids*, 257:105863, 2023.

[VBVWG23] D. Van Bulck, A. Vande Weghe, and D. Goossens. Result-based talent identification in road cycling: discovering the next eddy merckx. *Annals of Operations Research*, 325:539–556, 2023.