DM

Data Analysis of Trail Running
Runner Passage Prediction

MASTER DISSERTATION

Ricardo Daniel Pinheiro
MASTER IN INFORMATICS ENGINEERING

UNIVERSIDADE da MADEIRA
A Nossa Universidade
www.uma.pt

January | 2024

Data Analysis of Trail Running
Runner Passage Prediction

MASTER DISSERTATION

Ricardo Daniel Pinheiro
MASTER IN INFORMATICS ENGINEERING

SUPERVISION
Eduardo Miguel Dias Marques

Faculty of Exact Sciences and Engineering

Data Analysis of Trail Running: Runner Passage Prediction

Author:

Ricardo Daniel Pinheiro

Supervisor:

Prof. Dr. Eduardo Miguel Dias Marques

Constitution of the public examination committee:

Prof. Dr. Filipe Magno Gouveia Quintal, President

Prof. Dr. Fábio Rúben Silva Mendonça, Member

Prof. Dr. Eduardo Miguel Dias Marques, Member

On February 2, 2024, at 11 o' clock, this dissertation was defended at
`Room 2.108`, `Campus da Penteada`, `Funchal`
to obtain a Master's degree in **Informatics Engineering**

February 2024

Acknowledgements

First and foremost, I want to thank my supervisor, Professor Doctor Eduardo Marques. He has always been available, patient, and supportive in the development of this work. The extensive talks and his share of scientific knowledge led to the consolidation of this work with more confidence and motivation.

I acknowledge my friends and colleagues who helped me during my 7 years of academic pursuits, whether directly or indirectly. Those to whom this thank you is addressed will know :-)

I am thankful towards those that I met in 3 Erasmus+ Studies mobility semesters during my bachelor's and master's degrees. I also express my gratitude to the University of Madeira for providing these international opportunities.

I thank my friend Sandra for the endless hours of discussions and constant companionship and encouragement. Her cat Pingus was also a delightful companion throughout this academic year.

And finally, I want to express my gratitude to my brother Michael, for his support.

Thank you to everyone.

May education forever have no barriers for a more just society.

Abstract

Trail running is a running sport that spans across outdoor and mountainous terrain, often with hilly ascents and descents. This created a competitive scene that has garnered the interest of trail running enthusiasts into competing in 100km+ grueling races that can last for over 24 hours. Machine learning is a research field that focuses on developing algorithms capable of learning and making predictions based on data. In essence, it revolves around creating algorithms that can analyze data, identify patterns, and make informed predictions.

The organizations that setup trail running events operate a system that keeps track of the competitors arrival times across the designated points in a trail running course. The data analysis of trail running can provide valuable insights in respect to the planning of the event, namely in regard to the guarantee of the runner's safety, the scheduling of transport for runners, the allocation of human resources, and the allocation of food and beverages. A first initial research finds that the existing literature does not focus on the prediction in regards to both trail running and machine learning topics, the latter being a solution found in research related to marathon running.

At first, an initial approach was conducted in order to comprehend the prediction of arrival times based on an existing implementation. This approach analyzed the velocities of the runners, and such approach revealed that it would be optimal to utilize the Mean Absolute Error (MAE) as a metric for the next approaches. The second approach evolved into the use of the models LASSO and Random Forests with features related to both the checkpoints of a race and it's runners in competition. This approach netted a 24.45% MAE reduction when compared to the first approach. In the third approach, the LASSO model was excluded as the Random Forest model had overcome with least MAE. With the inclusion of more race time data in the last approach, we were able to reduce the MAE of the first approach by 28.71% with the employment of the Random Forest model.

Keywords: Trail running, Machine learning, ITRA, MIUT

Resumo

O *trail running* é um desporto de corrida que se estende por terrenos montanhosos e ao ar livre, muitas vezes com subidas e descidas acidentadas. Este desporto criou um cenário competitivo que despertou o interesse dos entusiastas do *trail running* para competirem em corridas extensas de mais de 100 km que podem durar mais de 24 horas. O *machine learning* é uma área de investigação focada no desenvolvimento de algoritmos capazes de aprenderem e de realizarem previsões baseadas em dados. Essencialmente, este centra-se na criação de algoritmos que podem analisar dados, identificar padrões e realizar previsões informadas.

As organizações que organizam eventos de *trail running* utilizam um sistema que regista os tempos de chegada dos concorrentes aos pontos designados num percurso de *trail running*. A análise de dados do *trail running* pode fornecer informações valiosas relativamente ao planeamento do evento, nomeadamente no que diz respeito à garantia da segurança dos atletas, ao agendamento do transporte para os atletas, à alocação de recursos humanos e à alocação de alimentos e bebidas. Uma primeira investigação inicial constata que a literatura atual demonstra poucos trabalhos relacionados com a previsão e associados aos tópicos de *trail running* e o *machine learning*, sendo esta última uma solução que já surgiu em contextos ligados a corridas de maratonas.

Numa primeira fase, foi efectuada uma abordagem inicial para compreender a previsão dos tempos de chegada com base numa implementação existente. A primeira abordagem baseou-se na análise das velocidades dos atletas, o que revelou que seria ótimo utilizar o *Mean Absolute Error* (MAE) como métrica para as abordagens seguintes. A segunda abordagem evoluiu para a utilização dos modelos LASSO e *Random Forests* com *features* relacionadas com características associadas aos postos de controlo da corrida e com os atletas em competição. Esta abordagem obteve uma redução de 24.45% do MAE em comparação com a primeira abordagem. Na terceira abordagem, o modelo LASSO foi excluído, uma vez que o modelo *Random Forest* obteve o menor MAE. Com a inclusão de mais dados de tempo de corrida na última abordagem, conseguimos reduzir o MAE da primeira abordagem em 28.71% com o uso do modelo *Random Forest*.

Keywords: *Trail running*, *Machine learning*, ITRA, MIUT

Table of Contents

List of Figures

List of Tables

List of Acronyms

CP Checkpoint

ETL Extract-Transform-Load

ITRA International Trail Running Association

LASSO Least Absolute Shrinkage and Selection Operator

LOYO Leave-One-Year-Out

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

MIUT Madeira Island Ultra Trail

MSE Mean Squared Error

R^2 Coefficient of Determination

RF Random Forests

RMSE Root Mean Squared Error

UML Unified Modeling Language

XGBoost Extreme Gradient Boosting

1 Introduction

In recent decades, trail running has experienced significant growth as a sport [6], particularly in the early 2000s, although its origins date back to the late 1970s in the United States.

Trail running is a sport that combines running with mountainous terrain, typically on hiking trails without pavement. According to the International Trail Running Association (ITRA), trail running is defined as such [7]:

> *A trail race is a pedestrian competition open to everyone, which takes place in a natural environment, with the minimum possible of paved roads (20% maximum). The course can range from a few kilometers for short distances all the way to 80 kilometers and beyond for ultra-trail races. Mountains or forests, countryside or desert, this endurance race takes place on naturally variable terrain, including very often significant climbs and descents, which result in elevation gain and loss between the start and finish line. The distance isn't the only thing that matters! Together, the unique features of the terrain and the relationship between distance and elevation changes all work together to create the overall level of difficulty for a given race.*

With this definition, we can see that Figure 1 is an example of a typical trail running race with steep ascents in a hilly environment, along with many runners competing in the trail.

Fig. 1: SkyRunner USA Series [1]

One trail running event to be studied is the Madeira Island Ultra Trail[1] (MIUT). Since 2008, Madeira Island (Portugal) has hosted this trail running event, spanning fourteen editions[2]. As of 2023, the event includes five races, each with different courses and a specific set of mandatory equipment requirements.

It may be of help to visualize a race profile of a race, in order to comprehend how distant and high a trail running race can be. In Figure 2, we can observe the race profile of the MIUT

[1] `https://www.miutmadeira.com/en/`
[2] The 2010 and 2020 editions were canceled

115km race, which includes 10 checkpoints, including the finish line. The elevation gain/loss is +7290m and -7320m, with several uphill and downhill sections. In the 2023 edition, the race began at midnight on the 22nd of April, with the finishing line closed at 08:00 on the day after. A runner can compete in the race course for 32 hours, the maximum time limit imposed in the race.

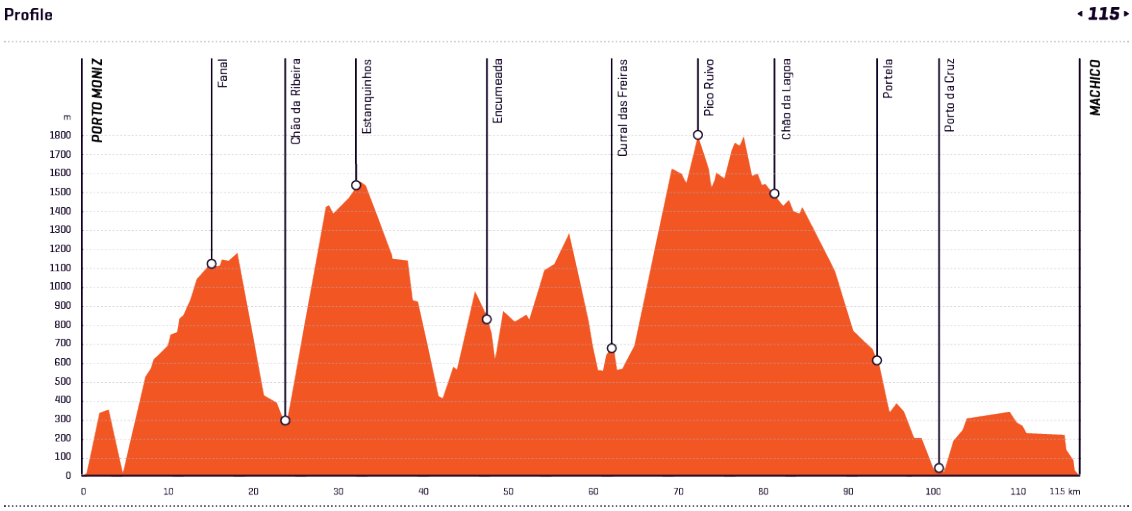


Fig. 2: Profile of the 115km race in Madeira Island Ultra Trail (2023 edition) [2]

In Figure 2, we can see Pico Ruivo, a checkpoint at the highest peak of Madeira Island (1862m) that cannot be reached by transport, as the volunteers must carry food and beverages by foot to the checkpoint for 1 hour on a trail. This introduces complexities to the logistics planning of the event, as this requires the volunteers to be on time in order to welcome the runners and register their time passage in the checkpoint.

Trail running races typically have checkpoints along the race course for the purposes of time control, in order to assure the safety of the runners and as well to avoid any kind of cheating on behalf of a runner. The checkpoints contain several features, providing food and beverages to runners, medical assistance, and the assistance of a personal assistant, if the runner has one. Sometimes these runners, more so the elite runners, have teams that also accompany them throughout the race. These checkpoints are usually managed by a group of individuals, often volunteers. The allocation of these volunteers require pre-planning in regards to their schedule, what will be their function, and in which checkpoints they will provide assistance to the runners, whether it's verifying if hot foods at the checkpoint are warm for adequate consumption, the transport scheduling of a runner that gave up on the race or the supervision of the timing system.

A variety of logistics processes are occurring in real-time during the event, and the surge of a large amount of runners arriving at a checkpoint in a trail running race [8] can be overwhelming for the volunteering team as this can create overcrowding and can complicate the workload of the volunteers in order to guarentee the runners well-being and everyone's safety, including the volunteers themselves and any race spectators nearby.

Despite a brief explanation of trail running as a sport, it is often used [9] as an umbrella term for other running-related sports as it follows.

- Mountain running - A major component of mountain running is elevation gain or loss, along with the chance of having steep ascents or descents as there are criteria for the amount of elevation change on the race course.
- Fell running - A fell race is a race on a hill or mountainous terrain, although it may seem similar to trail running, but fell running happens exclusively on mountainous trails and nowhere else.
- Cross-country - Cross-country races typically range between 3 to 12km in distance and the course width is at least 6 meters. It's focused on team-oriented competitions and although it is also done on non-paved trails, there are very specific rules and guidelines defined by the USATF (USA Track & Field).
- Skyrunning - Skyrunning races are mostly focused on altitude. These races occur above 2000 meters of altitude and within Skyrunning, there's SkyMarathon whose objective is to reach 4200 meters of altitude within the usual marathon distance (42.195km) and SkyRace whose objective is to reach 3000 meters of altitude.
- Ultrarunning - This type of running goes beyond the marathon's distance (42.195km) and they're also defined by mountain trails or paved surfaces.

1.1 Motivation

The purpose of this dissertation is to improve the logistical planning of trail running events, prioritize runner safety, and provide data insights to race spectators. This will be accomplished by exploring algorithms that can predict a runner's passage time at the next checkpoint. The evaluation of the algorithms will involve comparing metrics and performing error analysis.

The prediction of a trail runner's arrival at the next checkpoint can be beneficial for personal assistants and organizers of trail running events. Knowing the expected arrival time can help personal assistants provide necessary support to the runner, while organizations can improve logistics with this information. As a result, accurate time-based predictions are necessary in order to gain the trust of human beings. As such, metrics must be used to compare the accuracy of different algorithms and models.

Providing context for the current study is crucial after stating the purpose. This research aims to address challenges in predicting runner passage times in trail running races. Developing accurate predictive models and identifying optimal algorithms are critical for enhancing race management and logistics. Determining the optimal algorithm evaluation method will enhance the dependability and replicability of prognostic outcomes in this field. These goals will benefit not only race organizers and participants, but also have wider implications for the implementation of predictive modeling and algorithm evaluation in other logistical and sporting contexts.

The problem under investigation concerns predictive analysis in trail running races, specifically the prediction of a runner's passage time at the next checkpoint and the accuracy assessment of the method. This can enhance logistics handling for trail running events, such as determining when to assemble and disassemble aid stations, optimizing the number of shuttle buses for runners who quit halfway through the race, ensuring runner safety, and informally updating race spectators or personal assistants[3] on a runner's progress throughout a race.

[3]In MIUT, a designated personal assistant can aid a runner throughout the race

Moreover, this work will take an organizational point of view, meaning that predicting a runner's passage time at a checkpoint for a trail running race will be approached within the logistics of the race. This means that predicting a runner's checkpoint passage time during a trail running race falls within the realm of logistics of a trail running race. Therefore, there are three research questions in order to approach this problem:

1. How accurately can we predict a runner's passage time in a given checkpoint within a trail running race?
2. What is the most accurate performance metric, for assessing the precision of arrival time predictions in trail running races?
3. What is the most effective method for assessing the accuracy and performance of models?

The analysis of the results will include performance metrics for the algorithms used, which will be compared to determine the most optimal algorithm. In a real-world scenario, these algorithms can be utilized in a platform that receives real-time passage time data from a trail running event, allowing the race spectators and organizers to have more informed interactions with the event.

1.2 Organization of the dissertation

This dissertation is divided into several chapters, and the following is a summary of each chapter that contributes to the exploration of the research theme.

In Chapter 1, we introduced trail running and the research theme, along with its significance and relevance to the field. The motivation and scope are outlined in respect to the qualities of being able to know when a runner will reach the next checkpoint, and how this can be auxiliary information for both the race spectators and organizers.

In Chapter 2, presents a review of the existing literature related to the research topic, with a broad overview of machine learning and its definitions. The chapter explores trail running to characterize the sport and identify topics of interest for researchers. It lays the groundwork for the subsequent chapters by presenting machine learning concepts and the foundation of trail running.

In Chapter 3, we examine the process of preparing the collected data for the case studies ahead. It explains the techniques used to collect, clean, and organize the necessary data. Additionally, it provides an overview of exploratory data analysis to highlight the particularities of the data.

In Chapter 4, we present the first case study, which aims to create algorithms using basic statistical tools based on the average velocity of competing runners to predict a runner's passage time for the next checkpoint, and such races selected for this case study were the MIUT 115km races between 2016 to 2022, as a starting point to elaborate this case study based on long-distance races. The chapter details the methodology and findings of the case study, and analyzes the results to determine which approach yielded the least amount of error.

In Chapter 5, we build on the foundation laid in Chapter 4 by exploring machine learning techniques to improve the accuracy and reduce the error in the prediction of passage times, which will allow for a comparison of the results between the case studies. Our focus is to compare the outcomes of different approaches for the MIUT 115km races held between 2016 and 2023. This will provide a benchmark for determining the most successful approach in terms of accuracy and error yield.

In Chapter 6, the last case study is unveiled as a step up from Chapter 5 in terms of the inclusion of other long-distance races but also shorter distance races in order to verify if similar results occur, as a means to assimilate which approach may be more adequate in terms of accuracy and what exceptional results may arise as a consequence of the inclusion of more races into the last case study.

In Chapter 7, final remarks and future work are presented as guidelines for the improvement of what this dissertation has elaborated for the related fields of study that were addressed.

2 State of the art

This chapter will introduce a collection of works that establish the groundwork for this dissertation's implementation. The focus will rest heavily on analyses that employ statistical methods and/or machine learning algorithms to predict when trail runners will arrive at the next checkpoint during a trail running race. Since this is associated with the logistics management of trail running events, logistics is also a topic of interest that can assist us in implementing predictions of the runner's arrival at the next checkpoint.

In the initial section of this chapter, we will examine fundamental machine learning concepts to gain insight into how to plan, execute, and assess prediction algorithms. Therefore, we will delve into defining data and its suitability for machine learning objectives. We will explore three primary problem paradigms in machine learning, beginning with **supervised learning**, intended for problems where the data includes labeled examples. Contrary to this, **unsupervised learning** deals with problems where the data is unlabeled. The third paradigm, **reinforcement learning**, focuses on how agents can take actions in an environment to maximize rewards over time. After assessing these three paradigms, we review methods for evaluating supervised learning models in order to analyze the model's accuracy and gain insight into decision-making regarding the model's optimal parameters. To conclude the subsection, we compare several supervised models based on their strengths and weaknesses.

The second section will cover the literature review conducted on the subject of trail running as a sport, in order to comprehend the research topics of interest. Although this is not the primary focus of our project, acknowledging other areas of research is crucial.

The literature review is repeated in the third subsection, this time covering machine learning and trail running. This was necessary as the previous review only yielded one relevant paper concerning our research questions. Thus, this third subsection functions as a filter to ensure that the works done are relevant to our primary project focus, which is predictive analysis for predicting when a trail runner will reach the next checkpoint during a trail running race. Machine learning is also explored in this subsection with respect to marathon races, and although marathon races do not resemble trail running, it is still a running sport, which therefore may be used as inspiration to further enrich our project.

2.1 Machine learning

Machine learning is a technology that involves developing computer algorithms capable of emulating human intelligence. The field draws on concepts from various disciplines, including artificial intelligence, probability and statistics, computer science, information theory, psychology, control theory, and philosophy. This technology has been applied in diverse fields such as pattern recognition, computer vision, spacecraft engineering, finance, entertainment, ecology, computational biology, and biomedical and medical applications. The most important property of these algorithms is their distinctive ability to learn the surrounding environment from input data with or without a teacher [10].

The field of machine learning has received several formal definitions in the literature. Arthur Samuel defined machine learning as 'a field of study that gives computers the ability to learn without being explicitly programmed.' Tom Mitchell defined machine learning as 'A computer program learns from experience (E) with respect to some class of tasks (T) and performance measure (P),

if its performance at tasks in T, as measured by P, improves with experience E, using a computer science lexicon. Similarly, Ethem Alpaydin defined machine learning as the field of 'Programming computers to optimize a performance criterion using example data or past experience' in his textbook. These definitions share the idea of teaching computers to perform tasks beyond traditional number crunching by learning from repeated examples of their environment [10]. Although there has been increased crosstalk with other fields in the past decade, we are only beginning to tap into the potential synergies and diversity of formalisms and experimental methods used across these multiple fields to study systems that improve with experience [11].

This section examines the topics related to machine learning in detail by discussing core concepts such as data points, features, and the primary categories of machine learning techniques: supervised learning, unsupervised learning, and reinforcement learning. Finally, the last section presents performance metrics that evaluate the accuracy of the employed techniques.

The following section explores the definition of a dataset and its features, as well as the significance of features, data quality, and biased data.

2.1.1 Data and Features

The content covered in this section is based on the authors' definition of datasets in [5]. They define datasets as the following.

> *In ML, we deal with data and datasets. A dataset is composed of multiple data points (sometimes also called samples), where each data point represents an entity we want to analyze. Therefore, a data point can represent anything like a patient or a sample taken from a cancer tissue. Many of the issues related to data are universal and affect not only ML approaches but any quantitative discipline, including pharmacometrics. To compile the dataset, one has measured and collected a number of features (i.e., data that describe properties of the data points).*

The collected features in respect to a dataset can be categorized as either categorical (predefined values with no particular order such as male or female), ordinal (predefined values with an intrinsic order, such as disease stages), or numerical (real values).

Each feature represents a dimension of the feature space, and the specific value of a feature for a given data point positions the point in a defined location in that dimension of the space. Collectively, all the values of all features of a data point make up a feature vector. The greater the number of features collected for the dataset, the higher the resulting dimensionality of the feature vector and feature space. As dimensionality increases, visualizing the feature space dimensions becomes more complex, and computers must be utilized to identify relevant patterns or to apply dimensionality reduction methods (explained later in section 2.1.3). Most machine learning algorithms are designed to handle high-dimensional datasets, allowing for a large number of features within the existing data. As a preprocessing step, data transformation can significantly impact model performance. Thus, is it advisable to generate relevant features based on pre-existing features, a process commonly referred to as feature engineering [12].

Data quality is a crucial factor in machine learning as outliers and missing data can affect the accuracy of the results. Not all machine learning techniques are capable of handling data missingness, which may require data transformation as a preprocessing step. There are various methods for imputing data, but the approach used depends on the specific dataset and method

employed. One of the more common ways of imputing data involves the replacement of missing values with the mean of the feature mean across all samples where the value is defined [5].

Bias is also another important factor in data quality, which in itself can affect the ability of the model to generalize beyond the training dataset (and even the test dataset if both share a similar bias). By inspecting feature importance, this provides information about the magnitude and the effect of the bias, which is recommended to be used for checking the trustworthiness of machine learning models [5].

The following section presents techniques utilized in supervised learning - a paradigm for problems where the data is labelled.

2.1.2 Supervised learning

In supervised learning, the computer is provided with training data consisting of observations and their corresponding known output values. The objective is to learn general rules, also known as a 'model', that can map inputs to outputs. This enables the prediction of output values for new, unseen data, where input values are observed but their associated output is not [5].

In Figure 3, we can observe a typical architecture for supervised learning. This involves training a model on labeled data and testing it on unlabeled data. The fundamental architecture begins with collecting the dataset, partitioning it into testing and training data, and then preprocessing the data. Extracted features are fed into an algorithm, and the model is trained to learn the features associated with each label. Finally, the model is provided with the test data, and it makes predictions on the test data by providing the expected labels [3].

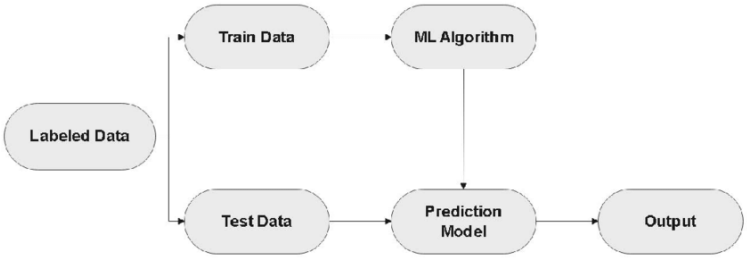


Fig. 3: A basic architecture of supervised learning [3]

There are two types of supervised learning: **classification** and **regression**. In classification, a model predicts unknown values based on a set of known values. When the output is in categorical form, the problem is referred to as a classification problem. Classification is also known by different terms:

- Classification algorithm (or classifier) - it learns from the training dataset and assigns each new data point to a certain class
- Classification model - it uses a mapping function, which is concluded by the model from the training dataset, to predict the class label for the test data
- Feature - it is associated with the dataset, which helps in building a precise predictive model

In regression problems, predictions are made for continuous values based on input variables. Examples of such predictions include a person's weight, age, or salary, weather forecasts, or housing

price predictions. Regression can be divided into two primary categories: simple linear and multiple. Simple linear regression involves a straight line representing the relationship between two variables. In multiple regression, this can involve more than two variables and is divided into linear and non-linear models.

In respect to other existing models, we present several models commonly used for classification and regression in Figure 4. K-nearest neighbors, decision trees, support vector, ensemble bagging/boosting methods, and ANNs (including deep neural networks) are models that can be used for both classification and regression with small modifications. However, linear regression and logistic regression are models that cannot (or cannot easily) be used for both problem types [4].

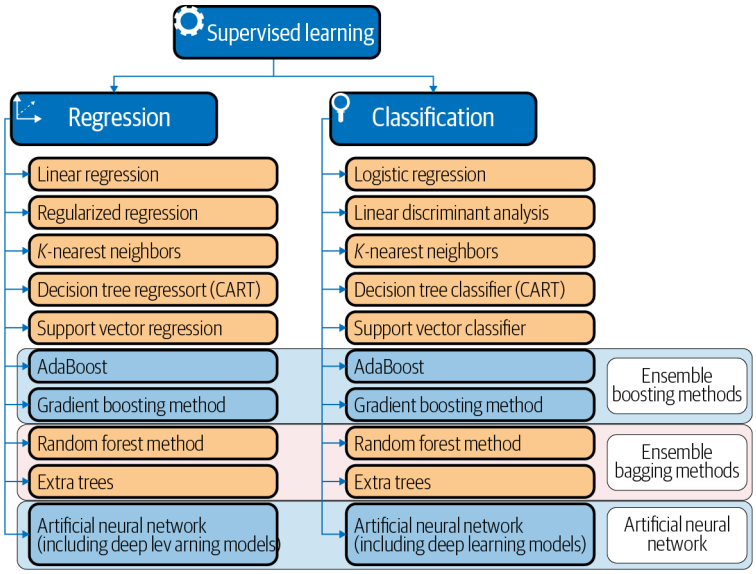


Fig. 4: Models for regression and classification-based problems [4]

This section will review certain models observed in Figure 4 in order to provide an explanation of these common models used in supervised learning.

Linear Regression is a model for regression-based problems that focus on finding relationships and dependencies between variables. It represents a modeling relationship between a continuous scalar dependent variable y (also called label or target in machine learning terminology) and one or more explanatory variables (also called independent variables, features, data points, etc.) denoted by X using a linear function. This takes the form of Equation 1, where β_0 is called intercept and $\beta_1 \dots \beta_i$ are the coefficients of the regression.

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_i x_i \tag{1}$$

Logistic Regression is a model used for classification and regression-based problems. It predicts the probability of an event occurring by fitting data to a logistic function. The model utilizes numerical or categorical predictor variables. To distinguish it from the linear regression model, the logistic regression model modifies its output domain to $[0, 1]$. This is because without this modification, the model could produce outputs less than zero or greater than one, which would not make sense when predicting probabilities. To avoid this, a sigmoid function in Equation 2 is applied [4]:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

Equation 3 presents an example of a logistic regression:

$$y = \frac{exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_i x_1)}{1 + exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_i x_1)} \tag{3}$$

Where y is the predicted output, β_0 is the bias or intercept term, and β_1 is the coefficient for the single input value (x). Each column in the input data has an associated β coefficient (a constant real value) that must be learned from the training data.

K-nearest neighbors (KNN) is a "lazy learner", in the sense that there is no learning required in the model. Predictions are made by searching through the entire training set for the K most similar instances (the neighbors) and summarizing the output variable for those K instances.

To determine which of the K instances in the training dataset are most similar to a new input, a distance measure is used. *Euclidean distance* is generally used for this, which is calculated as the square root of the sum of the squared differences between a point a and a point b across all input attributes i, which is represented by Equation 4.

$$d(a, b) = \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2} \tag{4}$$

Classification and regression trees (or CART or decision tree classifiers) are models that can be represented by a binary tree (or decision tree), where each node is an input variable x with a split point and each leaf contains an output variable y for prediction. Each node is tested according to a specific feature and it will be evaluated with a chosen criteria. If this criteria is passed, then the feature will pass to the next node. Otherwise, we follow up on another node successively until we have a selected classification. For regression predictive modeling problems, the cost function is different than of classification problems - while for classification all possible split points are evaluated and chosen in a greedy manner, in regression the cost function to minimize is the sum of squared errors across all training samples that fall within the rectangle. This cost function is represented in Equation 5.

$$\sum_{i=1}^{n} (y_i - p_i)^2 \tag{5}$$

where y_i is the output for the training sample and p_i is the prediction output for the rectangle [4].

Ensemble models combine different classifiers into a meta-classifier that has better generalization performance than each individual classifier alone. The two most popular ensemble methods are **bagging** and **boosting**. *Bagging* (or *boot-strap aggregation*) is an ensemble technique of training several individual models in a parallel way. Each model is trained by a random subset of the data. *Boosting* is an ensemble technique of training several individual models in a sequential way. This is done by building a model from the training data and then creating a second model that attempts to correct the errors of the first model. By combining individual models, the ensemble

model tends to have less bias and variance. Ensemble methods combine multiple, simpler algorithms to obtain better performance. Just as decision trees, bagging, and boosting can be applied in both classification and regression tasks.

Random forest is a tweaked version of bagged decision trees. A bagging algorithm works in the following way (with an example of one thousand instances):

1. Create many (e.g., one hundred) random subsamples of our dataset
2. Train a CART model on each sample
3. Given a new dataset, calculate the average prediction from each model and aggregate the prediction by each tree to assign the final label by majority vote.

A problem with decision trees like CART is that they are greedy. They choose the variable to split by using a greedy algorithm that minimizes error. Even after bagging, the decision trees can have a lot of structural similarities and result in high correlation in their predictions. Combining predictions from multiple models in ensembles works better if the predictions from the submodels are uncorrelated, or at best are weakly correlated. Random Forest changes the learning algorithm in such a way that the resulting predictions from all of the subtrees have less correlation [4].

Adaptive boosting (or **AdaBoost**) is a boosting technique to try predictors sequentially, and each subsequent model attempts to fix the errors of its predecessor. Each iteration change the sample distribution by modifying the weights attached to each of the instances. It increases the weights of the wrongly predicted instances and decreases the ones of the correctly predicted instances [4].

Gradient boosting method (GBM) is another boosting technique similar to AdaBoost, where the predictors are tried sequentially. Gradient boosting works by sequentially adding the previous underfitted predictions to the ensemble, ensuring the errors made previously are corrected. Contrary to AdaBoost, which tweaks the instance weights at every interaction, this method tries to fit the new predictor to the residual errors made by the previous predictor [4].

Extreme gradient boosting (XGBoost) is an implementation of gradient boosted decision trees designed to improve speed and performance. The implementation is a decision tree-based ensemble machine learning algorithm that uses a gradient boosting freamwork. Suppose we have a training data x_i and their labels y_i, XGBoost utilizes a classifier to predict the final prediction ${\hat{y_l}}^t$ in Equation 6.

$$ {\hat{y_l}}^t = \sum_{k=1}^{t} f_k(x_i) = {\hat{y_l}}^{t-1} + f_t(x_i) \tag{6} $$

Where ${\hat{y_l}}^{t-1}$ is the previous prediction and $f_t(x_i)$ is the new prediction [13].

Artificial neural networks (ANNs) are systems based on a set of connected units or nodes called artificial neurons, which model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal from one artificial neuron to another. An artificial neuron that receives a signal can process it and then signal additional artificial neurons connected to it. From the perspective of supervised learning, neural network are reducible to a classification or regression model with the activation function of the node in the output layer. In the case of a regression problem, the output node has linear activation function (or no activation

function). A linear function produces a continuous output ranging from $-\infty$ to ∞. Hence, the output layer will be the linear function of the nodes in the layer before the output layer, and it will be a regression-based model. In the case of a classification problem, the output node has a sigmoid or softmax activation function. Both functions produce an output ranging from zero to one to represent the probability of target value. Softmax function can also be used for multiple groups for classification [4].

The next section is related to another type of machine learning techniques - unsupervised learning.

2.1.3 Unsupervised learning

Contrary to supervised learning, unsupervised learning algorithms attempt to infer patterns from the data without any knowledge of the output the data is meant to yield. Without requiring labeled data, which can be time-consuming and impractical to create or acquire, this family of models allows for easy use of larger datasets for analysis and model development. There are two key techniques within unsupervised learning - **dimensionality reduction** and **clustering**.

Dimensionality reduction compresses the data by finding a smaller, different set of variables that capture what matters most in the original features, while minimizing the loss of information. This helps to mitigate problems associated with high dimensionality and permits the visualization of salient aspects of higher-dimensional data that is otherwise difficult to explore. Example of one technique that is frequently used for dimensionality reduction is **Principal component analysis** (PCA). PCA reduces the dimensionality of a dataset with a large number of variables, while retaining as much variance in the data as possible. PCA allows us to understand whether there is a different representation of the data that can explain a majority of the original data points [4].

PCA finds a set of new variables that, through a linear combination, yield the original variables. The new variables are called *principal components* (PCs). These principal components are orthogonal (or independent) and can represent the original data. The number of components is a hyperparameter[4] of the PCA algorithm that sets the target dimensionality. The PCA algorithm works by projecting the original data onto the principal component space. It then identifies a sequence of principal components, each of which aligns with the direction of maximum variance in the data (after accounting for variation captured by previously computed components). The sequential optimization also ensures that new components are not correlated with existing components. Thus the resulting set constitutes an orthogonal basis for a vector space. The decline in the amount of variance of the original data explained by each principal component reflects the extent of correlation among the original features. The number of components that capture, for example, 95% of the original variation relative to the total number of features provides an insight into the linearly independent information of the original data.

Clustering is a category of unsupervised learning techniques that allows us to discover hidden structures in data. Both clustering and dimensionality reduction summarize the data, however clustering categorizes the original data instead of creating new variables. Clustering algorithms assign observations to subgroups that consist of similar data points. The goal of clustering is to find a natural grouping in data so that items in a given cluster are more similar to each other than to those of different clusters. Clustering serves to better understand the data through the lens of

[4]Hyperparameters are variables that are set before the training process, and they cannot be learned during training

several categories or groups created. **K-means** is the most well-known clustering technique. The algorithm of k-means aims to find and group data points into classes that have high similarity between them. This similarity is understood as the opposite of the distance between data points. The closer the data points are, the more likely they are to belong to the same cluster [4].

The algorithm finds k centroids[5] and assigns each data point to exactly one cluster with the goal of minimizing the within-cluster variance (called *inertia*). It typically uses the Euclidean distance, but other distance metrics can be used. The k-means algorithm delivers a local optimum for a given k through randomly moving around the specified number of centroids in each iteration of the algorithm, assigning each data point to the closest centroid. Once we have done that, we calculate the mean distance of all points in each centroid. Then, once we can no longer reduce the minimum distance from data points to their respective centroids, we have found our clusters. In Figure 5, we can observe an example of k-means clustering with k set to 2.

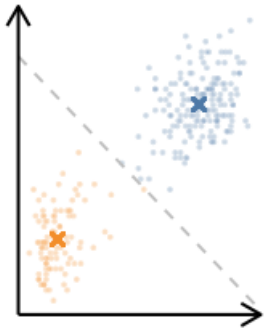


Fig. 5: Outcome of clustering through k-means with $k = 2$. The X represents the closest center point to each cluster [5]

The last section of machine learning techniques - reinforcement learning - is discussed below, along with its core concepts and a few of the techniques are also displayed.

2.1.4 Reinforcement learning

Reinforcement learning (RL) is an approach towards training a learner (or machine) to learn what and how to do, and how to create solutions to a situation through the means of the feedback that the learner receives, either through reward or punishment. This feedback is enabled by the learner according to his discovery of actions that yield the most reward by trying them. Two of the distinguishing features of reinforcement learning are the trial-and-error search approach that it does and delayed reward, which means that by the learner's subsequent actions, he may (or not) receive rewards later in his discovery process of the solution [15].

Authors in [4] describe three main concepts of RL: **1.** Components, **2.** Modeling framework, and **3.** Model types.

1. Components
   - Agent - the entity that performs actions
   - Actions - the things an agent can do within its environment

[5] A centroid is a data point (imaginary or real) at the center of a cluster [14]

- Environment - the world in which the agent resides
- State - the current situation
- Reward - the immediate return sent by the environment to evaluate the last action by the agent

The goal is to learn an optimal strategy through experimental trials and feedback loops. An optimal strategy will make the agent capable of actively adapting to the environment to maximize the rewards. An agent's actions are usually conditioned on what the agent perceives from the environment. What the agent perceives is referred to as the observation or the state of the environment.

2. Modeling framework

   Almost all RL problems can be framed as **Markov decision processes** (MDPs). MDPs formally describe an environment for RL. A MDP consists of five elements: $M = S, A, P, R, \gamma$, described as the following.

   - S - a set of states
   - A - a set of actions
   - P - transition probability
   - R - reward function
   - γ - discounting factor for future rewards

   MDPs frame the agent-environment interaction as a sequential decision problem over a series of time steps $t = 1, \ldots, T$. The agent and the environment interact continually, with the agent selecting actions and the environment responding to these actions and presenting new situations to the agent, to come up with an optimal policy or strategy. Bellman equations[6] form the basis for the overall algorithm.

3. Model types

   RL can be categorized into *model-based* and *model-free* algorithms, based on whether the rewards and probabilities for each step are readily accessible.

   Model-based algorithms try to understand the environment and create a model to represent it. When the RL problem includes well-defined transition probabilities and a limited number of states and actions, it can be framed as a finite MDP for which dynamic programming can compute an exact solution.

   Model-free algorithms try to maximize the expected reward only from real experience, without a model or prior knowledge. Model-free algorithms are used when we have incomplete information about the model. The agent's policy[7] $\pi(s)$ provides the guideline on what is the optimal action to take in a certain state to maximize the total rewards. Each state is associated with a value function $V(s)$ predicting the expected amount of future rewards we can receive in this state by acting on the corresponding policy. In other words, the value function quantifies how good a state is.

[6] Bellman equations refer to a set of equations that decompose the value function and action-value function (or Q-value) into the immediate reward plus the discounted future values

[7] Policy is a set of rules that describe how an agent makes its decisions

RL comes with some challenges, such as being computationally expensive and data intensive and lacking interpretability, but it can align perfectly with areas that are suited for policy frameworks based on reward maximization. The next subsection will present the idea of performance metrics, metrics that evaluate whether or not the algorithm/model being executed is accurate within the scope of its purpose.

2.1.5 Evaluation metrics

The metrics used to evaluate the machine learning algorithms are very important. The choice of metrics to use influences how the performance of machine learning algorithms is measured and compared. The metrics influence both how you weight the importance of different characteristics in the results and your ultimate choice of algorithm. The main evaluation metrics for regression and classification problems are in Table 1.

Table 1: Evaluation metrics for regression and classification problems [4]

Regression	**Classification**
Mean Absolute Error (MAE)	Accuracy
Mean Squared Error (MSE)	Precision
R Squared (R^2)	Recall
Adjusted R Squared (Adj-R^2)	Area Under Curve (AUC)
	Confusion Matrix

Mean Absolute Error (MAE) measures the sum of absolute differences between predicted and actual values. It reflects a linear score where each individual difference is weighed equally in the average. This score offers an indication of the degree of inaccuracy in the predictions made. The measure provides an estimation of the error's magnitude, yet it does not indicate its direction, such as over-predicting or under-predicting. The formula for the MAE is presented in Equation 7.

$$\mathrm{MAE}(y, \hat{y}) = \frac{1}{n_{\mathrm{samples}}} \sum_{i=0}^{n_{\mathrm{samples}}-1} |y_i - \hat{y}_i| \tag{7}$$

Mean Squared Error (MSE) is a metric that measures the average of the squared differences between predicted values generated by a model and actual observed values in a given dataset. To calculate the MSE, we take the average of the squared differences between each predicted value and its corresponding actual value. **Root Mean Squared Error** (RMSE) is calculated based on the MSE and offers a more understandable way of measuring error. It is calculated by taking the square root of the MSE, which scales the error back to the original units of the output variable. RMSE is a useful metric to express the model's error in the same units as the target variable, making it a preferred option. The formula for the MSE and RMSE is presented in Equation 8.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{\mathrm{true},i} - y_{\mathrm{pred},i})^2 \qquad \mathrm{RMSE} = \sqrt{\mathrm{MSE}} \tag{8}$$

R Squared (R^2) provides an indication of the "goodness of fit" of the predictions to actual value. In statistical literature this measure is called the coefficient of determination. This is a value

between zero and one, for no-fit and perfect fit, respectively. The formula for R^2 is presented in Equation 9.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_{\text{true},i} - y_{\text{pred},i})^2}{\sum_{i=1}^{n}(y_{\text{true},i} - \bar{y}_{\text{true}})^2} \tag{9}$$

Adjusted R Squared (Adj-R^2) shows how well terms fit a curve or line but adjusts for the number of terms in a model. It is given in Equation 10.

$$\text{Adj-R}^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - k - 1} \tag{10}$$

where n is the total number of observations and k is the number of predictors. Adjusted R^2 will always be less than or equal to R^2.

Mean Absolute Percentage Error (MAPE) calculates the average of absolute percentage errors [16]. This metric remains independent of scale and produces results that are easier to interpret, making it a popular choice across various industries. The formula for the MAPE is presented in Equation 11.

$$\text{MAPE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \tag{11}$$

Selecting an evaluation metric for supervised regression depends on the objective. If the main objective is predictive accuracy, then RMSE is recommended. It is computationally simple and is easily differentiable. The loss is symmetric, but larger errors weigh more in the calcuation. The MAEs are symmetric but do not weigh larger errors more. R^2 and adjusted R^2 are often used for explanatory purposes by indicating how well the selected independent variable(s) explains the variability in the dependent variable(s) [4].

After displaying several evaluation metrics for regression problems, we will approach the evaluation metrics for classification problems. Binary classification problems will be presented in this section as a baseline to elaborate on the common evaluation metrics. In binary classification problems, these involve only two outcomes, such as true or false. Some frequently used terms within this context include:

- True Positives (TP): Predicted positive and are actually positive
- False Positives (FP): Predicted positive and are actually negative
- True Negatives (TN): Predicted negative and are actually negative
- False Negatives (FN): Predicted negative and are actually positive

Three commonly used evaluation metrics for classfication are: precision, recall and accuracy. Figure 6 displays the formulas:

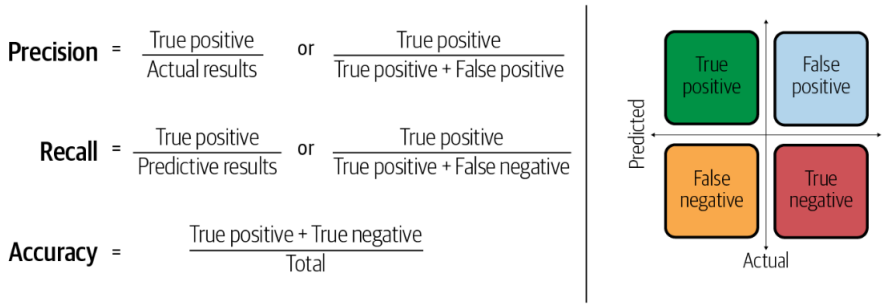


Fig. 6: Accuracy, precision, and recall [4]

Accuracy is the number of correct predictions made as a ratio of all predictions made. This is the most common evaluation metric for classification problems and is also the most misused. It is most suitable when there are an equal number of observations in each class (which is rarely the case) and when all predictions and the related prediction errors are equally important, which is often not the case.

Precision is defined as the percentage of true positive predictions out of the total predicted positive instances. The denominator is the total number of instances identified as positive by the model. Precision is particularly useful in cases where the cost of false positives is high, for instance, in email spam detection.

Recall (or sensitivity or true positive rate (TPR)) is the percentage of positive instances out of the total actual positive instances. Therefore, the denominator (TP + FN) is the actual number of positive instances present in the dataset. Recall is a good measure when there is a high cost associated with false negatives (e.g., fraud detection).

Area under ROC curve (AUC) is an evaluation metric for binary classification problems. ROC is a probability curve, and AUC represents degree or measure of separability. It tells how much the model is capable of distinguishing between classes. The higher the AUC, the better the model is at predicting zeros as zeros and ones as ones. An AUC of 0.5 means that the model has no class separation capacity whatsoever. The probabilistic interpretation of the AUC score is that if you randomly choose a positive case and a negative case, the probability that the positive case outranks the negative case according to the classifier is given by the AUC.

Confusion Matrix lays out the performance of a learning algorithm. The confusion matrix is simply a square matrix that reports the counts of the true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions of a classifier, as shown in Figure 6.

The confusion matrix provides a clear and concise presentation of a model's accuracy when dealing with multiple classes. The table maps predictions on the x-axis and accuracy outcomes on the y-axis. Each cell represents the number of predictions made by the model. For instance, a model could predict either zero or one, and these predictions could actually be either zero or one. Predictions for zero that accurately matched actual values of zero appear in the cell where prediction = 0 and actual = 0, while predictions for zero that actually turned out to be one are displayed in the cell where prediction = 0 and actual = 1.

Selecting an evaluation metric for supervised classification depends on the task at hand. For instance, recall is a useful measure when false negatives carry a high cost, as is often the case in fraud detection.

2.1.6 Model Selection

The selection of the ideal machine learning model can be a challenging task [4], as no single solution or approach is a one-size-fits-all solution. There are several factors that can influence the choice of a machine learning model. Typically, the main criterion is the model's performance, which has been discussed previously. However, many other factors should be considered during the model selection process. Such factors considered for the model selection process include the following.

Simplicity The degree of simplicity of the model. Simplicity usually results in quicker, more scalable, and easier to understand models and results.

Training time Speed, performance, memory usage, and overall time taken for model training.

Handle nonlinearity in the data The ability of the model to handle the nonlinear relationship between the variables.

Robustness to overfitting The ability of the model to handle overfitting, which can occur by a model fitting training data too closely, capturing noise, and will perform poorly on unseen test data due to lack of generalization.

Size of the dataset The ability of the model to handle a large number of training examples in the dataset.

Number of features The ability of the model to handle the high dimensionality of the feature space.

Model interpretation How explainable is the model? Model interpretability is important because it allows us to take concrete actions to solve the underlying problem.

Feature scaling Does the model require variables to be scaled or normally distributed?

Figure 7 illustrates a comparison between supervised learning models on the factors mentioned and outlines a general rule-of-thumb to narrow down the search for the best machine learning algorithm[8] for a given problem.

Relatively simple models, such as linear and logistic regression, are less complex than ensemble and ANN models. Linear models and CART have shorter training times than ensemble methods and ANN. Linear and logistic regression cannot handle nonlinear relationships, unlike all other models. Support Vector Machines (SVM) can handle the nonlinear relationship between dependent and independent variables with nonlinear kernels[9].

The following section will provide a literature review of the trail running sport practice to gather insights on research interests within the field of trail running.

[8]The table does not include AdaBoost as the overall behavior across all the parameters is similar to Gradient Boosting and Random Forest

[9]In machine learning, a "kernel" is a mathematical function that allows SVMs to transform data into a higher-dimensional space, facilitating the separation of data points in cases where a linear boundary is insufficient to capture complex relationships.

	Linear regression	Logistic regression	SVM	CART	Gradient boosting	Random forest	Artificial neural network	KNN	LDA
Simplicity	✔	✔	✔	✔	✘	✘	✘	✔	✔
Training Time	✔	✔	✘	✔	✘	✘	✘	✔	✔
Handle non-linearity	✘	✘	✔	✔	✔	✔	✔	✔	✔
Robust to overfitting	✘	✘	✔	✘	✘	✔	✘	✔	✘
Large datasets	✘	✘	✘	✔	✔	✔	✔	✘	✔
Many features	✘	✘	✔	✔	✔	✔	✔	✘	✔
Model interpretation	✔	✔	✘	✔	✔	✔	✘	✔	✔
Feature scaling needed	✘	✘	✔	✘	✘	✘	✘	✘	✘

Fig. 7: Model comparison across supervised learning models [4]

2.2 Trail Running

Initially, a search was conducted to assess the existence of literature related to predictive analysis and trail running. However, the search yielded no relevant results. Hence, the purpose of this section is to gain a deeper understanding of the research areas of interest in trail running. While our project primarily focuses on predictive analysis, it is essential to acknowledge other relevant topics being studied.

The literature review is separated into 3 phases: **Identification**, to collect and identify studies from search engines/academic databases, **Assessment**, to appraise and synthesize studies through a set of filters, and **Included**, to gather studies that comply with the established criteria. In Figure 8, we can observe the flow of search up until the articles that were included in the end of the search.

In the first phase, Identification, a search was conducted on Google Scholar[10] using the term `"trail running"`, which yielded 14,600 results. As our work's objectives do not require an exhaustive review of all 14,600 records, we deemed it sufficient to extract the first 100 records based on their relevance ranking on the search engine. This could introduce bias, but the goal is solely to obtain a general representation of the efforts.

In the Assessment phase, the 100 records were categorized by screening their titles and/or abstracts according to their area of discipline. As a result, we obtained 21 different categories. However, we observed that some of these categories have subjects in common, prompting us to consider creating a new level of categorization, designated as **Class**. The identified classes are: **Health** (61), **Sports** (21), **Social Sciences** (16), and **Others** (2). In Table 2, the classes and categories, along with the total number of articles, are displayed.

The analysis of the articles showed that the topics of interest for research on trail running are mostly related to health, with the remaining associated with sports or social sciences. The two articles in the **Others** class were excluded as they were not related to trail running. To further understand what topics are often sought after, 9 articles were selected based on their amount of

[10] `https://scholar.google.com/`

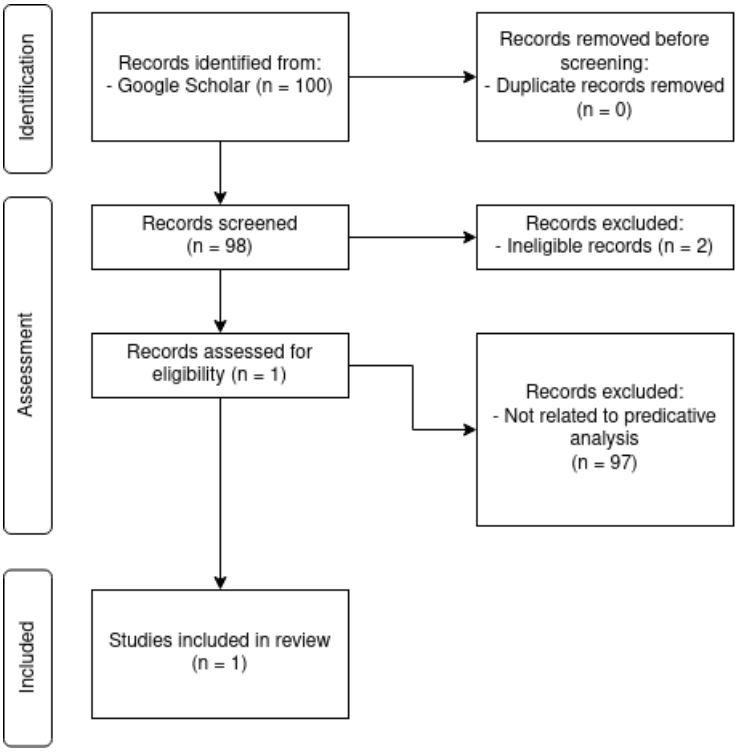


Fig. 8: Identification of relevant items for review flow diagram

Table 2: Classes and categories of collected articles

Class	Category	Articles	Sum of Articles
Health	Physiology	32	61
	Injuries	11	
	Biomechanics	6	
	Psychology	6	
	Cardiology	2	
	Diabetes	1	
	Endocrinology	1	
	Hemorheology	1	
	Immunology	1	
Sports	Ergonomics	8	21
	Sports	5	
	Wearables	4	
	Pacing Strategy	2	
	Performance	1	
	Physical Education	1	
Social Sciences	Management	9	16
	Tourism	4	
	Geography	1	
	Law	1	
	Topography	1	
Others	Out of scope	2	2

citations. The articles selected will be succinctly presented in the remainder of this section and 5 are related to the **Health** class, 2 to **Sports**, and 2 to **Social Sciences**.

The records under the **Health** class explore topics regarding the influence of hydration [17] on a runner's body functions and performance during trail running in the heat. Four 12-km runs were done in the heat with hydrated and dehydrated conditions, with the outcome that a reduction in hydration status will cause the runner's body functions and performance to be impaired while trail running in the heat. A similar study [18] was done to analyze runner's body functions and running speed. Three 4-km laps were done with 4-minute rests between laps, along with moderate running speed. Under hypohydration conditions, the runners in these laps had slower run times, greater perceived effort and elevated gastrointestinal temperature. Researchers conducted [19] a study about pacing under heat conditions with hydration and dehydration statuses in three 4 km loops with a 4-minute rest between loops. The findings reveal that dehydration is associated with a decrease in a runner's ability to evenly pace themselves during a competitive situation.

Young and master runners were studied [20] to understand the effects of a trail running competition on the runner's muscles, with findings that trail running is detrimental to muscle and that training may not halt muscle deterioration through aging, but can help maintain performance level. Muscle and kidney injuries [21] were found to be possible to be predicted based on exercise data retrieved from a 35-km trial run by 20 well-trained runners.

Records under the **Social Sciences** class explore topics regarding trail degradation [22] in protected areas that host trail running events or organized sporting events. An urban protected area of Hong Kong was empirically examined and evaluated for impacts generated by a running competition, the MSIG HK50 – 2015. It was found that a running competition can cause significant initial degradation on the trail and although some degradation features can recover in 6 months, some negative effects can persist or worsen. Vegetation cover impact [23] was studied under the cause of mountain bicycling, trail running, and hiking in a shortgrass prairie environment. Vegetation cover measurements were taken at multiple intervals after recreational use. This resulted in increases in bare ground cover between the first measurement and post-treatment sampling one year later. Short-term effects were more variable, walking and bicycling caused increases in bare ground cover, but running did not. The authors suggest that the dynamics of trail formation from running deserve further attention and likely differ from hiking or mountain biking impacts.

Records under the **Sports** class explore the influence of wearing compression socks [24] on body responses and performance responses post-exercise from trail running. Road runners have been wearing compression socks, in particular in an attempt to enhance performance. Eleven trained runners were tested under a 15.6km trail run at a competition while wearing or not wearing compression socks. The comparison of data with or without compression socks concluded that there is no difference between the run conditions and that the findings suggest that competitive runners do not gain any practical or physiological benefits from wearing compression socks during prolonged off-road running. The influence of calf compression sleeves [25] has also been studied to determine whether they affect physiological, biomechanical, and performance parameters. Fourteen runners took part in two identical 24-km trail running sessions with different running speeds and different compression garments. It was found that calf compression did not affect performance and physiological responses, but it did affect biomechanics and lower limb capabilities.

In the Included phase, we obtained 1 record, but it will be analyzed in section 2.3.

In most cases, the literature is concerned with the runner's bodily functions under trail running conditions, which may hinder a runner's efficiency in competing in trail running events. Methods to improve runner's efficiency on the trail are also of interest, with emphasis on athletic equipment, pacing strategies, and wearable technology. While this is not directly related to runners, the management of trail running events, its potential for tourism/leisure purposes, and the consequences towards protected areas have been studied. Under our main objectives, no literature was found regarding the performance of runners in order to use it as a means of predictive analysis, nor the logistics handling of event planning. Only 1 article was found to be very in common with our objectives, which will be analyzed in the next section.

The following section will explore the use of predictive analysis and machine learning in trail running, which is closely aligned with the main focus of our project. We will also investigate other sports to determine if their methods can be applied to our work.

2.3 Machine learning applied to trail running & other sports

In this section a new literature review was performed to focus the results in efforts of more interest to our project. The reason to not to focus from the beginning was to have a clear understanding of the trail running field as a whole.

Since our area of interest is machine learning, we will include it in the trail running terminology while avoiding any papers that fall outside the scope of our research.

159 results were found in Google Scholar, 9 in ScienceDirect[11], and 5 in ACM Digital Library[12], with many of them not being relevant to either trail running and/or machine learning. In Google Scholar, the search query: "trail running" AND "machine learning" -campus -deception -wearable -satellite -earthquake -query -connectionist -robot -hockey -security -photo-sharing -classroom -"off-policy evaluation" -"long skip connection" -patients was written to exclude irrelevant keywords to our project focus that would appear in the search results. This search query resulted in 19 results, which couldn't further be reduced due to the search query character limit. Analogous to the previous section, we will go over a second review to further understand if there are articles in common with our project focus. In Figure 9, we can observe the filtering process up until the articles that were included in the end of the search.

In the Identification phase, 33 records were collected, with 1 duplicate record being eliminated. During the Assessment phase, 24 records were eliminated for not being related to trail running or machine learning, and 7 more were eliminated at the end of the Assessment phase for not aligning with our project focus mission. In the Included phase, 1 record was saved for review. This is the identical record saved from the previous subsection, suggesting that research on our project focus is limited to both trail running and machine learning.

Following up with the work found in [26], the Trail Running Assessment of Performance (TRAP) assesses the runner's performance before and during the race. Through statistical machine learning tools, three indicators are calculated regarding the runner's:

- expected passage time
- probability of giving up

[11] https://www.sciencedirect.com/
[12] https://dl.acm.org/

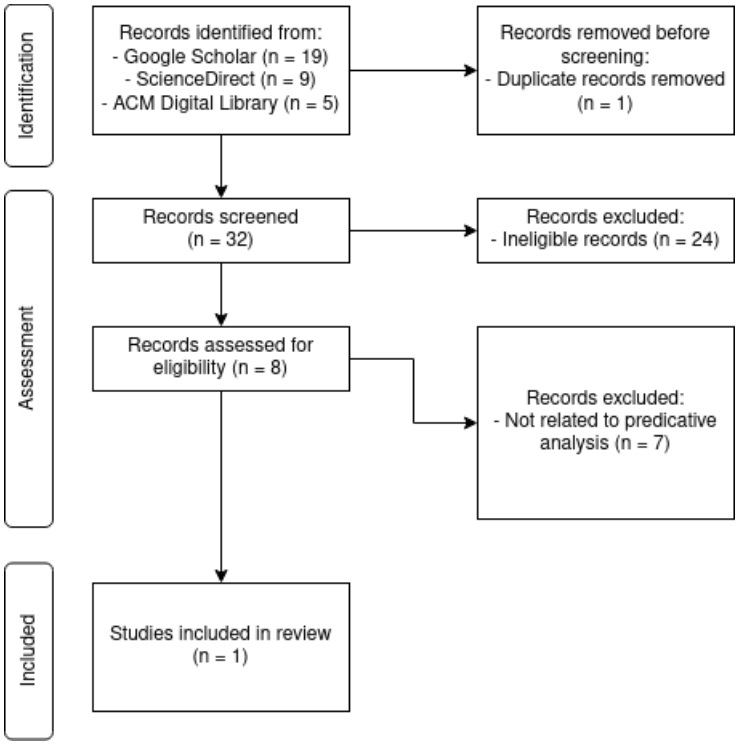


Fig. 9: Identification of relevant items for review flow diagram

– prediction interval for passage time

These indicators are calculated for the next checkpoint based on the runner's position, with the possibility of extending this over to multiple checkpoints ahead at arbitrary positions.

The features that are used to calculate these indicators belong to three different sets: checkpoint-level features, runner-level features, and lag information. Checkpoint-level features contain geographical data and competition features such as food, liquids, or medical assistance. Runner-level features contain demographical data and information on past race history through ITRA. Lag information is based on delayed versions of the passage time when predicting the passage time (and probability of giving up) at the next checkpoint. Beyond that, the checkpoint-level features can be combined with lagged passage time, resulting in improved predictions.

The models considered in this framework are the LASSO regression model [27], XGBoost [28] and Random Forests [29]. The intercept-only model (baseline) is also considered but only meant to serve as a reference point for the model validation.

The LASSO regression model is used for the expected passage time model with the default mean-squared error (MSE) measure to choose the regularization parameter with cross-validation (CV), while for the probability of giving up model, the logistic LASSO regression with the area under the curve (AUC) measure was used. Both models use the one standard error regularization penalty.

XGBoost and Random Forests are tree-based models trained with default MSE loss for the expected passage time model. For the probability of giving up, both the LASSO and XGBoost were trained to maximize the AUC. Random Forests were trained as described in [30].

To ensure that the selected models are performing well in more than one race, leave-one-year-out (LOYO) CV (e.g., train on all runner and race checkpoints in years 2015 to 2017 and test on 2018) was used. The root mean squared error (RMSE) and area under the curve (AUC) were used to evaluate the models for the expected passage time and probability of giving up, respectively. The models were evaluated based on two criteria: **1.** Overall performance measures and **2.** Level of confidence verification of all checkpoints.

The different sets of features were also evaluated, with four separate groups:

1. Checkpoint
2. Checkpoint + Runner
3. Checkpoint + Runner + Lag 1
4. Checkpoint + Runner + Lag 1 & Lag 2

The expected passage time model obtained the most favorable results with tree-based models, it suggested that these perform better than the intercept-only model and the LASSO model. The addition of both sets of lag variables also contributed to an increase in model performance. The model in regards to the probability of giving up yielded comparable results with LASSO, XGBoost, and Random Forests, with XGBoost being the best-performed model out of these three.

An analysis of the features was carried out to provide the best performance for the selected models. It was found that the most important variable for predicting the expected passage time at the checkpoint is the gain in elevation, with the second being related to the distance from the last checkpoint. For predictions of the probability of giving up, the most important variable is the increase in altitude and a relative 1-lagged response, and the second variable is the cumulative distance traveled.

The study reviewed in this section provided us with a starting point for deciding what models/metrics should be used under the context of trail running, in specific to the arrival prediction of a runner to the next checkpoint. Additional importance was given to this study due to the amount of studies found within the same objectives as this work. This also gives us a streamlined route to compare our future results with the results from [26] that we can opt in.

The previous literature searches in 2.2 and in this section resulted in the same one paper, and because we did not find any other papers in the same project focus as ours, this part of the state of the art will provide additional insight into machine learning techniques, but within marathon running since it is also a running sport that spans exactly 42km, thus this has similarities with trail running, although this search related to marathon running is not in-depth as we only wanted a small amount of works for this part of the section.

A deep learning approach was employed [31] to predict athletes' marathon finishing times with their race results and public training workout data. This data originates from the 2017 Boston Marathon, within a five-month window before the race. The deep learning models used are Recurrent Neural Networks (RNN) [32], Long Short-Term Memory (LSTM) [33] and Gated Recurrent Units (GRU) [34]. All of these three are used and evaluated against several deep learning performance metrics, such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), to assess which model has the highest prediction accuracy. Through the MAPE evaluation method, the GRU model was the most accurate out of the three

with 95% accuracy, RNN at 94%, and LSTM at 89%. Moreover, the three deep learning models also outperformed the standard linear regression model.

A proposal was investigated [35] to build in-race recommendations for runners, based on existing techniques that are currently employed for in-race finish-time prediction. Data was collected from 7931 marathon finishers, with its source in the Strava[13] app, and contains information on a runner's pace, heart rate, and cadence during the marathon. The features used to build the model to predict the finish time were generated at every 500m interval in the race. With these, a separate model was built at every interval using XGBoost to predict a runner's finish time from that point. To evaluate the model, it was compared against the prediction of a baseline prediction. The baseline used is often used by marathon race organizers to calculate a runner's finish time and assumes a runner will run the race at the same pace for the duration of the race. To predict race splits (pacing slowdown) and its magnitude at every point in the race, another model was built with XGBoost, however an extended set of features were generated to improve the model's performance. As a performance metric, Mean Absolute Error (MAE) was used and the pacing recommendations *versus* the actual pace were calculated for error. As a result, if a runner follows the recommendations, they will allow themselves to adapt their pace to achieve a minimal slowdown.

Researchers in [36] also investigated the topic of marathon pacing to understand if machine learning can be used to identify and assist recreational runners in adjusting their pace depending on their fitness level and the temperature of the day. The dataset consists of 423496 entries from runners who completed the Gothenburg Half Marathon between the years 2010 - 2019. Three methods were used to predict the finish time from passing intermediate split times: the baseline model, a model that uses multiple linear regression, and a neural network consisting of one hidden layer with 40 nodes. For evaluation, the Mean Absolute Error (MAE) was calculated, using 5-fold cross-validation with an 80-20% random split averaging over all splits. Both the linear regression model and the neural network outperform the baseline at all intermediate splits with the neural network slightly outperforming the linear regression model at the 10 and 15km splits. Additional features were experimented with, proving to improve model performance, especially for predictions early in the race. It was also found that the Balanced Random Forest model had the best performance to predict which runners will suffer a slowdown of over 25% on a 5 km segment, compared to a previous base-pace. The base pace is defined as the average pace over the first 0-5km and 5-10km segments. The model identified 76% of the runners who suffered a slowdown (Recall 0.76) but at the cost of also labelling many who did not suffer a slowdown as such (Precision 0.19).

Running events can have image documentation of the events themselves, and through image annotation, the work in [37] proposed an automatic image annotation system using the YOLOv3 [38] algorithm based racing bib number recognition method. The dataset consisted of photos from the BNI ITB Ultra Marathon 2019. The photos gathered have at least one visible racing bib that is mostly not obstructed by any other obstruction. To evaluate the model performance towards test data, an object detection metric called mAP (mean value of average precision (AP)) is used for each object class in the model. AP is an object detector evaluation metric used at the PASCAL VOC Challenge [39]. Precision is the ratio between the true object prediction and the total number of object prediction by the model. Recall is the ratio between the true object prediction and the total number object in the ground truth label. F1 score is used to describe how well the balance between precision and recall from the racing bib number prediction was made, thus these three are

[13] `https://www.strava.com`

used as performance metrics. As a result, 83.0% Precision, 81.5% Recall and 82.2% F1 Score were achieved on the racing bib number recognition with the proposed method.

The literature on machine learning in the context of trail running focuses on evaluating trail runners' performance before and during races. Technical abbreviations are explained upon their initial usage. Three indicators are used to calculate the runner's progress towards the next checkpoint: expected passage time, probability of giving up, and prediction interval. These indicators rely on the runner's position. The authors used regression-based models, including LASSO, XGBoost, and Random Forest, and evaluated them using metrics such as root mean squared error (RMSE) and area under the curve (AUC). They also assessed the importance of features to determine which variables are most important for prediction. In the literature on marathon running, the focus is on predicting athletes' finishing times, providing in-race recommendations, and recognizing racing bibs through image analysis. Various authors use different models and evaluators, including deep learning models such as RNN, LSTM, and GRU, as well as XGBoost, Random Forest, and the YOLOv3 machine learning algorithm. They also use the standard linear regression model as a benchmark.

2.4 Conclusion

To conclude this chapter, we will give a summary of the papers that were synthezied.

The machine learning solutions presented a range of three main paradigms: supervised learning, unsupervised learning, and reinforcement learning. Selecting one of these will depend on what type of data is available, that is, whether or not the data is labelled or unlabelled. To assess the evaluation of the models related to supervised learning, a wide array of metrics - MAE, MSE, R^2, Adj-R^2, Accuracy, Precision, Recall, AUC, and Confusion Matrix, with their own purpose in either regression or classification-based problems.

Concerning the literature search, it was revealed that only one paper related to our objectives was found to be of interest. However, this resulted in gaining knowledge in respect to the topics of interest for research on trail running. We learned that for the most part, the literature on trail running refers to 4 classes, due to the initial analysis resulting in 21 different categories. After grouping these 21 categories, this resulted in 4 classes: Health, Sports, Social Sciences, and Out of Scope (2 papers were found to be not related to trail running). The first 3 classes' approach matters in regards to the runner's bodily functions, and methods to improve the runner's efficiency in trail running races with particular attention to athletic equipment, pacing strategies, and wearable technology. The handling of management of trail running events was also explored, as a result of its potential for tourism/leisure purposes, and the consequences against protected areas also have been studied. To further improve our literature search under our interests and objectives, another literature search was made to filter our search results in-depth.

The second literature search in 2.3 was attained by searching for both trail running and machine learning as keywords. The same paper obtained in the first literature search was found again in the end of the synthesis. This may suggest to us that this paper can be used as inspiration for later stages of the implementation of the project. Based on these two literature searches, we can see that both literature searches issued [26] as a result - in these works the authors predicted the expected passage time of runners in each checkpoint on trail running races through the use of supervised learning methods. The models: LASSO regression model, XGBoost, and Random

Forest were used, with more favorable results in the use of XGBoost and Random Forest. Linear regression was also used as a benchmark to compare with the previous three mentioned models. To assess the evaluation, the mean squared error (MSE), and root mean squared error (RMSE) will be used. The importance of features is also important as they can help us understand what variables are most important, therefore giving us more insight into optimizing our analysis.

As a consequence of only attaining one paper relevant to our main project focus, we searched for works related to predictive analysis but applied to road running. What was found were works within marathon running that encompass different methods, but three out of four that were analyzed were indirectly related to assessing the runner's finish times or pacing strategization. Regarding what models were used, these were XGBoost, neural networks, and Random Forest, which were previously discussed, but we found new algorithms such as YOLOv3 - an object detection model, and the use of deep learning techniques were also identified. The evaluation metrics in use were RMSE, MAE, MAPE, F1 Score, Precision, and Recall, with most studies out of the four having used the Mean Absolute Error (MAE) in their methodology.

In conclusion, this chapter resulted in a synthesis of the base theory of machine learning, what topics of interest surround research in trail running, what methods have been used that align with our main project focus, and what approaches were used towards predictive analysis in the context of marathon running.

The following chapter will detail the data collection and processing procedures utilized on the collected dataset to address the research questions outlined in Chapter 1. A comprehensive overview of these procedures will be provided to ensure a thorough understanding of the analyzed dataset.

3 ETL for Data Preparation

This chapter will present the stage of implementation in order to answer our research questions posed in Section 1.1. We will introduce what data was collected and processed for analysis, in order to have a clear understanding of the type of data we have. Afterwards, a set of use-cases will take place with the same data process and each use-case will approach different methods to explore solutions that can address our research questions.

The collected data is stored in a MySQL[14] relational database, which is accessed through a web application[15]. The database stores data from trail running events that span through one or more races per event. Each race in turn contains one or more checkpoints, which are locations along the race course where runners are timed and recorded at checkpoint arrival. Runners are registered in one or more races, according to their corresponding inscription. Every runner can have one or more inscriptions, with each inscription containing the runner's category (constituted by their sex and age[16]). We can represent these entities in an adapted UML[17] class diagram:

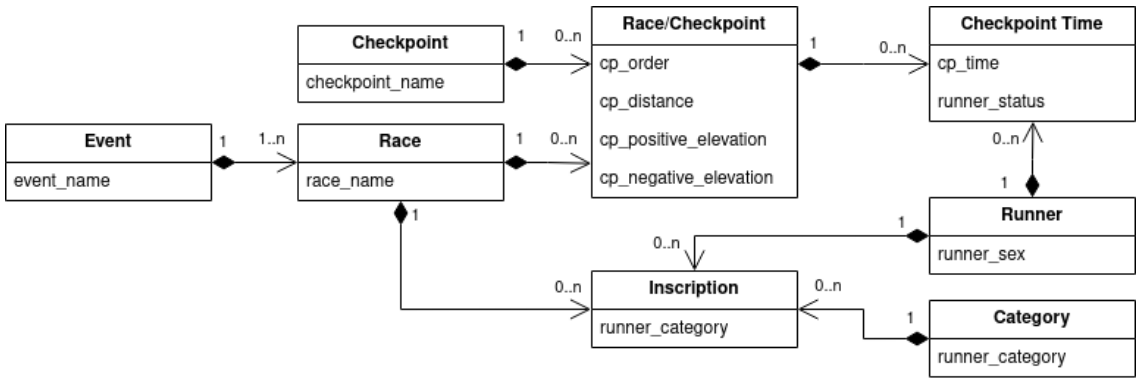


Fig. 10: UML diagram of the entities

Initially Figure 10 had a different entity scheme than the actual database. The database design contains conceptual errors, which were discussed with the database authors. However, Figure 10 will now follow the logical structure of the real database, since database design is not relevant to our work of interest.

The adopted framework for data manipulation is the **extract-transform-load** (ETL) framework, used for data integration from multiple sources [40]. The ETL framework is separated in three phases:

1. **Extract**: involves data extraction from appropriate data sources. Data is usually available in flat file formats such as CSV files, XLSX/ODS files and TXT files. In other cases, data extraction can be done from a database, XML, or JSON files;

2. **Transform**: involves cleansing data, sometimes invoking quality checks, to comply with the target schema. Typical transformation activities involve normalizing data, removing duplicates,

[14] `https://www.mysql.com/`

[15] `https://ts.uma.pt/` - Timing System Information from Trail Running events

[16] The used data in this project did not include any personal information, besides the runner's sex and age

[17] `https://www.uml.org/`

checking for integrity constraint violations, filtering data based on defined regular expressions, sorting and grouping data, and applying built-in functions where necessary, and;

3. **Load**: involves propagating the transformed data, usually into an operational database for production or other uses.

The next subsection will present the initial data exploration, to better adapt the trail running information to the case studies. This process followed the ETL framework to clarify the data manipulation performed over the initial data.

3.1 Extract

The extracted data contains all necessary information to answer the research questions. Table 3 presents the columns of interest identified in the database, including the details necessary for case study analysis.

Table 3: Identified data of interest in the database

#	Name	Description
1	runner_id	Unique identifier of a runner
2	runner_sex	Sex of a runner
3	runner_category	Category of a runner
4	runner_status	Race status of a runner
5	event_id	Unique identifier of an event
6	race_id	Unique identifier of a race
7	cp_id	Unique identifier of a checkpoint
8	cp_order	Checkpoint number of the race
9	cp_time	Date/Time of the recorded checkpoint time
10	cp_distance	Checkpoint distance
11	cp_positive_elevation	Checkpoint positive elevation
12	cp_negative_elevation	Checkpoint negative elevation

The columns represent the described model of information, and the data exists in several tables from the database. The final matrix of data extracted, for the purpose of simplification of the analysis, only one table is kept with all the required information for data processing. There is some redundancy, but this organization simplifies the analysis of the information. An extract of the information is presented in Table 4, where ten random rows were selected, from only one race (`race_id` 280).

Table 4: Data rows examples from one race (ordered by `cp_time`)

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
136466	F	V45 F	EP	111	280	9	7	2022-04-23 16:11:11	5100	395	500
136172	F	SEN F	DNF	111	280	9	7	2022-04-23 16:11:31	5100	395	500
133234	M	SEN M	EP	111	280	9	7	2022-04-23 16:11:39	5100	395	500
137108	M	V45 M	DSQ	111	280	15	5	2022-04-23 16:11:56	14800	935	1130
7847	M	SEN M	EP	111	280	9	7	2022-04-23 16:12:04	5100	395	500
3979	M	SEN M	EP	111	280	8	6	2022-04-23 16:12:11	11000	1475	305
136568	M	SEN M	EP	111	280	8	6	2022-04-23 16:12:15	11000	1475	305
136387	M	V45 M	EP	111	280	223	8	2022-04-23 16:12:26	4500	20	245
136857	M	SEN M	DSQ	111	280	15	5	2022-04-23 16:12:27	14800	935	1130
136257	M	V40 M	EP	111	280	223	8	2022-04-23 16:12:30	4500	20	245

The data represents many different situations of a race, such as, the start of the race, in column 8 with value `0`, the quitting of a runner, in column 4 with value `DNF`, or the end of race, in column 8 with value `99`. In column 3, the runner's category is described with the runner's sex and age category. The data needed to be filtered and sanitized for further analysis and the next subsection will describe the second phase of the ETL framework for this project data.

3.2 Transform

During the **Transform** phase, the data is cleaned, filtered and consolidated. The primary objective of this phase is to make sure that the data is consistent, accurate, and relevant to the objectives of this project.

We will tackle several items in Table 3 that are necessary to undergo data transformation:

- `runner_status`: A runner can have four of the following status: EP - *In race*; DNS - *Did not start the race*; DNF - *Did not finish the race*; DSQ - *Disqualified*. The last three status are to be removed from the dataset given that we are not seeking to analyze runners that did not complete races - in doing so this could add noise to the dataset.
- `cp_order`: The first checkpoint (`0`) is the starting point of the race, but the rows associated to it are zeroed with the exception of `cp_time`, and this can add outliers to the dataset. So, any row associated to the starting point of the race is removed. The final checkpoint (`99`), is renamed to the last natural checkpoint order number as this is how spectators view the order of checkpoints in trail running races.
- `cp_time`: This column is converted into seconds, in the shape of integer numbers.
- `cp_distance`, `cp_positive_elevation` and `cp_negative_elevation` are to be interpreted in meters.

The data is totalled in 279434 rows pre-transformation, and post-transformation 258048 rows, less 21386 rows than before. This ensures that we are not relying on data that will only add noise to the analysis.

3.3 Load

The final phase of the ETL framework is concerned with the output of the transformed data. The transformed data is stored in comma-separated value files, organized by event. Each event contains the transformed data and any generated predictions for each race.

3.4 Case Study Organization

This section will present an introduction to the case studies developed in this dissertation.

In Section 4, the first case study will analyze the behaviour of runners based only in the time of passage in the checkpoints (in respect to the MIUT 115km race data of 2016 to 2022), and it is only possible to derivate the velocity of the runners. Since the number of data points is variable during a race, we will evaluate the impact of three different velocities: global average, last checkpoint average velocity, and, the last 2 checkpoint average velocity. It is intended to determine which velocity algorithm provides better predictions of runner behavior at the checkpoints. This

case study will evaluate the comparison process of each algorithm to serve as a basis for later case studies.

In Section 5, we present machine learning techniques to delve further into the prediction of the runners passage times in order to obtain improved results from the previous case study. The MIUT 115km race data between 2016 to 2022 are present in an effort to verify for any differences between the results of each race edition. Their respective features are also presented, among with which models and evaluation will be employed in the case study.

In Section 6, the use of machine learning techniques continue, but we extend the inclusion of all races available from the provided database by the organizers. A larger amount of races led us to classify the races according to their total distance and total positive elevation. This allows us to have clusters of races that are related with each other in regards to their difficulty, but for comparison we are also utilizing the race data without any clusters for prediction of the runners time passages.

The use of clustered races according to their difficulty and without the use of clusters for prediction both have two approaches, one with MIUT races exclusively and the second with MIUT races and other races outside of the MIUT event.

The next chapter will present the first case study.

4 Case Study: Velocity-based Algorithms

A prediction is a statement of when a future event will occur, in this case the context is time-based predictions based on historical data. Predicting can be useful for planning and decision-making, in trail running races this means that predicting checkpoint crossings of runners can help the race organizers (and personal assistant[18] if a runner has one) to know when a runner will reach his next checkpoint. This enables the organizers to adjust the logistics of each checkpoint depending on how many runners will be crossing a checkpoint at a time. Security is another factor that can be improved since the predictions can indicate us in the case that a runner is taking too long to reach the checkpoint, which can lead the race organizers to assume that the runner is injured, comitting cheating, or abandoned the race with no previous notification to the race organizers.

This chapter will present a case study with the challenge of obtaining predictions of checkpoint crossings in trail running races. In alternative, the challenge could be predicting when a runner will reach the end of the race but this is more complex, so predictions are done individually per checkpoint. The research questions described in Section 1.1 are questions related to the objective of this case study that is comprised of a subset of a dataset containing checkpoint crossings over time from a trail running race with the goal of predicting when a runner will cross his next checkpoint and to ensure that the methods are trustworthy of their predictions. The estimation of the arrival of a runner's next checkpoint is based on approximation of the runner's estimated velocity. The velocity is estimated at each checkpoint based on the runner's run time and travelled distance in order to be able to extrapolate an approximation of the runner's velocity at a given checkpoint. We will carry on this analysis within the 115km race of Madeira Island Ultra Trail 2022 as a baseline in this case study.

The following sections will: **1.** describe what methods were conceptualized, **2.** characterize the implementation of the case study, **3.** what results were achieved with these methods, and **4.** what can be learned from these results that can answer our research questions.

4.1 Definition of Velocity Algorithms

A collection of sets were written in order to present the formulas developed and used for time-based prediction based on historical data. The collection of sets are related to distance and time as these are the only measurements required for deriving the average velocity of the runners. The average velocity will be used a field site to understand if average velocity can be accurate in predicting when a runner will cross his/her next checkpoint.

For a particular trail running race TR with k total checkpoints, let the collection of sets be:

- $X = x_c$ be the set of distances between consecutive checkpoints, for $c = 1, 2, ..., k$
- $D = d_c$ be the set of total distance covered from the start of the race to each checkpoint, where $d_c = d_{c-1} + x_c$ for $c = 1, 2, ..., k$
- $T = t_c$ be the set of time crossings between consecutive checkpoints, for $c = 1, 2, ..., k$
- $E = e_c$ be the set of total elapsed time in each checkpoint, where $e_c = e_{c-1} + t_c$ for $c = 1, 2, ..., k$

The dependent collections of sets (D, E) are used as input to derive the average velocity of a runner at every checkpoint:

[18] A personal assistant is a person that accompanies the runner throughout his race

$$v_{A_1c} = \frac{d_c}{e_c} \text{ where } c = 1, 2, ..., k \tag{12}$$

eq. (12) contains the average velocity of a runner at checkpoint c based on the total distance covered and total elapsed time. The next two average velocities have a different criteria:

$$v_{A_2c} = \begin{cases} \frac{d_c}{e_c} & \text{c} = 1 \\ \frac{d_c - d_{c-1}}{e_c - e_{c-1}} & \text{c} > 1 \end{cases} \tag{13}$$

$$v_{A_3c} = \begin{cases} \frac{d_c}{e_c} & \text{c} = 2 \\ \frac{d_c - d_{c-2}}{e_c - e_{c-2}} & \text{c} > 2 \end{cases} \tag{14}$$

eq. (13) holds the average velocity of a runner at checkpoint c based on the distance covered and elapsed time between checkpoint c and the previous checkpoint $c-1$. This restricts the average velocity exclusively to a track segment instead of the entire track profile. $\frac{d_c}{e_c}$ is still present but only for the first checkpoint as there is no previous data to base on.

eq. (14) is analogous to eq. (13) but it comprises the distance covered and elapsed time between checkpoint c and the previous two checkpoints $c-2$.

The algorithms A_n for $n = 1, 2, 3$ generate a forecast for e_{A_nc+1} for every runner competing in the race (with $c = 1, 2, ..., k-1$):

$$e_{A_1c+1} = \frac{d_{c+1} - d_c}{v_{A_1c}} + e_c \tag{15}$$

$$e_{A_2c+1} = \frac{d_{c+1} - d_c}{v_{A_2c}} + e_c \tag{16}$$

$$e_{A_3c+1} = \frac{d_{c+1} - d_c}{v_{A_3c}} + e_c \tag{17}$$

These formulas (eqs. (15) to (17)) result in a prediction of the time of a runner's arrival at a checkpoint based on the distance they have travelled, the elapsed time, and the average velocity. Each formula incorporates a different average velocity, which is calculated based on distance and time, as demonstrated in eqs. (12) to (14).

4.2 Implementation

A sub-sample of the dataset presented in 3.1 will be used to evaluate the algorithms. This sub-sample refers to the runner's time crossings of Madeira Island Ultra Trail 2022 edition, more specifically in the 115km race as this contains more varied data compared to the other four races (85km, 60km, 42km, and 16km).

The resulting dataset contains 6160 samples, corresponding to the same number of time crossings, from 560 runners, setting 11 samples for each runner. The Python programming language and the Pandas[19] library in conjunction will be the tools for manipulating the data and its results.

The columns used in this case study are a specific subset of columns outlined in Table 3 (page 29) and detailed below.

[19]https://pandas.pydata.org/

inscription_id Represents a runner's unique identifier.

cp_order Indicates the sequence of the race. The checkpoints start at 1 to 11, with 11 being the finishing point.

cp_time Represents time measurements in seconds since the starting point of the race.

cp_distance Representation of distance in meters. This variable is related to the distance between the checkpoint and the previous checkpoint.

To better understand the dataset, a simple analysis is presented next. Out of the 560 runners, 493 are male and 67 are female. Also, every runner is placed in a race category that corresponds to their age and sex. There are 11 categories, represented in Figure 11, having SEN M as the most frequent category with 189 runners and V55 F is the least frequent category with 3 runners.

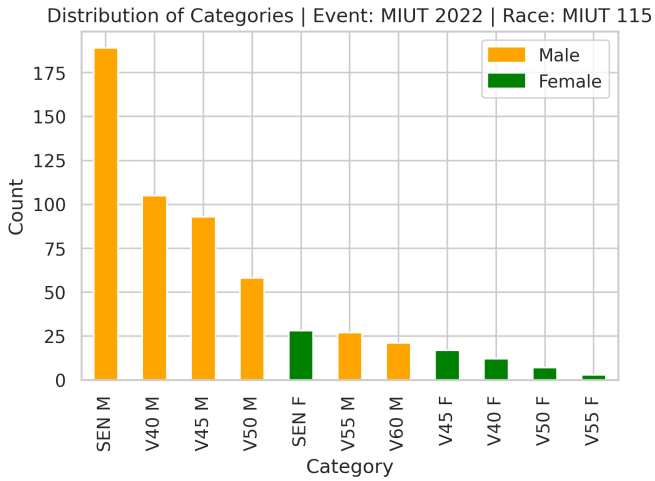


Fig. 11: Distribution of categories

Table 5 presents a global overview of the race across all checkpoints in regards to how much time the runners took to cross each checkpoint. The minimum and maximum values can hint at how fast or slow runners can be in each race segment. The maximum and minimum difference, can reveal which sections of the trail are most demanding, and the mean and median values can provide insights about the skewness of the checkpoint time data. Also, the underlying data in Table 5 is manipulated in eqs. (15) to (17), and a description of this data can improve our findings.

As an example, we can consider checkpoint 6. The fastest runner, from checkpoint 5 reached checkpoint 6 in 5897 seconds (1 hour, 38 minutes and 17 seconds), and the slowest runner in 18903 seconds (5 hours, 15 minutes and 3 seconds). The maximum-minimum difference is at 13006 seconds (3 hours, 36 minutes and 46 seconds), which can imply that the race segment is difficult, and with the mean value at 12641 seconds, and the median value at 13016 seconds, this can help to understand how demanding each race segment can be.

To further understand the distribution of the runner's time crossings in each checkpoint, a boxplot in Figure 12 is employed. As a point of reference, we can use the mean of the checkpoint time, and observe that there are 6 checkpoints with boxes above the red dashed line. In the case of checkpoint 3 to 6, they have a considerable amount of outliers (with the exception of checkpoint

Table 5: Descriptive statistics of checkpoint time

CP #	Location	Min.	Max.	Max.-Min.	Mean	Median
1	Fanal	6071	11806	5735	9052	9132
2	Chão da Ribeira	2571	7029	4458	4666	4724
3	Estanquinhos	5138	14269	9131	9071	9181
4	Encumeada	5983	14349	8366	10366	10388
5	Curral das Freiras	6453	17099	10646	11957	12074
6	**Pico Ruivo**	**5897**	**18903**	**13006**	**12641**	**13016**
7	Pico do Areeiro	2790	10278	7488	5622	5701
8	Chão da Lagoa	1264	3968	2704	2609	2577
9	Portela	3644	13931	10287	7917	7609
10	Porto da Cruz	1803	9618	7815	4666	4531
11	Machico	4924	18390	13466	10669	10635

5) and the whiskers length is progressively wider until checkpoint 7. The checkpoints 7 to 11 have a more varied pattern of the box sizes, but checkpoints 9 and 11 have a very wide upper whisker, similar to checkpoints 5 and 6.

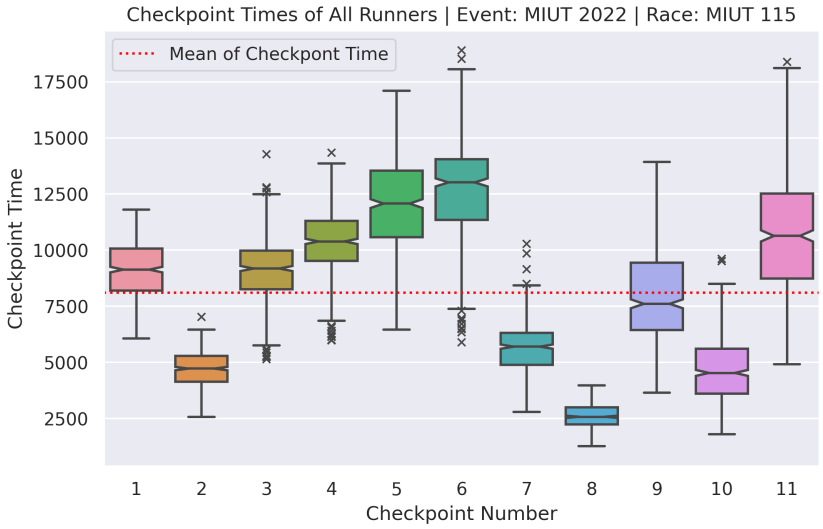


Fig. 12: Checkpoint times of all runners

Of the 11 checkpoints evaluated, 7 show outliers of varying amounts. Checkpoints 3, 4, 6, and 7 contain a notable number of outliers, with most being below the lower whisker. The plot style used for visualizing the checkpoint times gives an idea of the difficulty level each checkpoint poses for a runner. It also offers a preview of the impact that the checkpoint time may have on the analysis of the case study's results.

This section presented the sub-sample of the dataset that will be used throughout this case study. The purpose of this selected sub-sample is to match the type of data that we require in order to obtain predictions of the runner's arrivals to the checkpoints, and to evaluate the algorithm's performance. A descriptive overview of the runner's attributes is also presented, to further elaborate the dataset's aspects along with an aggregation of all of the runner's checkpoint times in each checkpoint that can explain the distribution of checkpoint times and the possibility of

heavy outliers in the dataset that can affect the data quality. The subsequent subsection presents a comprehensive overview of the results obtained after evaluating the algorithms.

4.3 Results & Analysis

In this section, we aim to present the results of the estimation of runners' future arrival time at the next checkpoint, conducted by data from one race (MIUT 2022 - 115km) into our methodology. This integration allowed us to calculate an approximate velocity for each runner. By extrapolating this estimated velocity, it's possible to predict the arrival time of the runners at the subsequent checkpoint.

In Table 6, after obtaining results, there are three columns to append to this dataset as an output of the methods that were presented in the subsection 4.1.

Table 6: Dataset of Velocity-based Algorithms case study

Name	Datatype	Observations
inscription_id	Integer	-
cp_order	Integer	1..11
cp_time_segment	Integer	Unit is seconds
cp_time	Integer	Unit is seconds
cp_distance	Integer	Unit is meters
pred_a1	Integer	Result of eq. (15)
pred_a2	Integer	Result of eq. (16)
pred_a3	Integer	Result of eq. (17)

The fields **pred_a1**, **pred_a2**, **pred_a3** contain the estimate of when a runner will reach to the next checkpoint, extrapolations of a runner's velocity based on the algorithms defined in section 4.1. The next section will evaluate the algorithms used for the predictions to understand how accurate they are through a global lens of the race.

4.3.1 Global Error Values

At this stage, the results require an evaluation of their accuracy, which aligns with one of our research questions: "How accurate can we predict a runner's passage time in a given checkpoint within a trail running race?". For computing the metrics, we selected scikit-learn[20] for Python as a tool among various approaches to evaluate metrics. A group of metrics is assembled to evaluate the accuracy of the algorithms in order to obtain a first-hand overview of what results the metrics return.

1. Mean Squared Error (MSE)
2. Root Mean Squared Error (RMSE)
3. Mean Absolute Error (MAE)
4. R Squared (R^2)
5. Mean Absolute Percentage Error (MAPE)

[20] https://scikit-learn.org/

Table 7 contains the global error values of the runners for each algorithm A_n. In respect to the research question related to evaluating which algorithm is most accurate, each algorithm's error values will be compared. The algorithm's error values generally point to algorithm A_2 being better than algorithms A_1 and A_3, with A_1 being best in MAE and MAPE and A_2 in MSE, RMSE and R^2.

Table 7: Global error values of algorithms A_n

Metric	A_1	A_2	A_3
MSE	4550862	4278569	5097861
RMSE	2133	2068	2257
MAE	1580	1586	1751
R^2	0.993	0.994	0.991
MAPE	3.41	3.62	3.60

Upon analyzing the error values, it can be observed that the MSE values are too large in comparison to RMSE and MAE values, as a consequence of MSE squaring every number. On the contrary, RMSE takes the square root, and MAE takes the absolute value of the subtraction between real and predicted values. The R^2 values, with their proximity to 1, imply that the data possesses a strong capability to find and capture patterns within it. For MAPE values, these indicate the average percentage difference between the actual values and the prediction values, which may be the cause of the MAPE value of A_3 being 3.608, which is lower than A_2's value at 3.628, but this does not align with the same pattern noticed with RMSE and MAE values.

Based on the analysis of various metrics, it can be concluded that the performance of the algorithms can be evaluated as follows.

RMSE and MAE The algorithms with the lowest RMSE and MAE values are considered to have performed the best. Note that RMSE values will always be larger than MAE values as it is more sensitive to outliers, as a consequence of it's squaring operation. MAE is based on absolute operation, so errors are interpreted as equal, regardless of the magnitude of outliers. If the RMSE and MAE values do not have a large difference among different algorithms, it indicates that their performance in terms of accuracy is comparable.

MSE In contrast to RMSE and MAE, if a algorithm has a significantly larger MSE value compared to other algorithms, it suggests that the algorithm's predictions have larger errors on average. Higher MSE values indicate a higher overall error magnitude, which may imply less accurate predictions. Intuitively speaking, RMSE is easier to interpret as it provides the average prediction error in the same units as the dependent variable.

R^2 The similarity of R^2 values across different algorithms suggests that the algorithms explain the variability in the data to a similar extent. This similarity also implies that the algorithms ability to accurately predict the dependent variable (the target variable) using the independent variables (the features). However, the similar R^2 values alone may not provide conclusive evidence about the superiority of one algorithm over another. It is necessary to complement R^2 with other metrics to assess the overall performance and reliability of the algorithms's predictions.

MAPE MAPE values, on the other hand, are not directly comparable to RMSE and MAE. MAPE is a percentage-based metric that measures the average percentage difference between the predicted and actual values. It is not as commonly used as RMSE and MAE for evaluating algorithm performance. Comparing MAPE values to RMSE and MAE may not provide a meaningful comparison due to the different scales and interpretations.

Considering the analysis of the metrics, three metrics will be omitted from here on forward and these are: MSE, R^2, and MAPE. Next, the reasons of this choice are presented.

MSE will be omitted as it exhibits a larger value compared to other metrics. This indicates that the algoritms' predictions have a higher overall error magnitude. By focusing on RMSE, which is derived from MSE but provides a more interpretable measure, we can still consider the magnitude of errors while maintaining a clearer understanding.

R^2 Although the algorithms exhibit similar R^2 values, which suggest that they explain the variability in the data to a comparable extent, R^2 alone does not provide conclusive evidence about the superiority of one algorithm over another. Therefore, R^2 will be omitted as well as it does not provide meaningful insights by itself for decision-making in this case study.

MAPE is a percentage-based metric that measures the average percentage difference between predicted and actual values. However, MAPE is not directly comparable to RMSE and MAE due to its different scale and interpretation. Therefore, MAPE is also omitted to maintain consistency and facilitate meaningful comparisons between algorithms.

Having explained the omissions, now, we explain why RMSE and MAE are chosen as the primary performance metrics for this case study. RMSE and MAE provide valuable information about the average magnitude of errors, regardless of the direction of those errors. Lower values of RMSE and MAE indicate higher accuracy and better-performing algorithms.

By focusing on RMSE and MAE, the evaluation process will prioritize accuracy and provide clearer insights into the performance of different algorithms. Omitting MSE, R^2, and MAPE allows for a more streamlined and effective decision-making process, keeping the focus on the most relevant and informative metrics.

The next section will evaluate the algorithms in respect to their accuracy in each checkpoint of the race.

4.3.2 Checkpoint Error Values

In order to gain a deeper understanding of the race dynamics and performance at various checkpoints along the trail, we applied the RMSE and MAE metrics. Instead of relying solely on global error values, analyzing the error values at each individual checkpoint provides valuable insights into the specific patterns and challenges encountered throughout the race.

By computing the RMSE and MAE for each checkpoint, we were able to assess the deviations between the actual recorded times and the predicted times. This granular analysis allowed us to identify checkpoints where the predictions were accurate, indicating a smoother and more predictable segment of the race. In contrast, checkpoints with higher RMSE and MAE values revealed areas where the predictions were less precise, highlighting potential race complexities or unpredictable conditions.

We begin by presenting the error values at each checkpoint in Table 8. The predictions done to checkpoint 2 (CP2) contain the minimum error value, below 400 seconds out of all checkpoints as it is the first time the predictions occur, and the checkpoint distance, referred in Table 3, between CP1 to CP2 is 6.35km. On the other hand, the maximum error value is held by CP6, between 3882 to 4963 seconds. The distance from CP5 to CP6 is 9.45km and the distance difference of the segments CP1 to CP2 and CP5 to CP6 is 3.10km. However the magnitude difference of the error values is approximately 10 times the size from the minimum values. This leads us to believe that there are other factors that can influence the error values of the predictions. These factors can be associated to elevation change, terrain difficulty, weather conditions or the runner's strategy and tactics throughout the race.

Table 8: Error values of the checkpoints for algorithms A_n

Checkpoint	**RMSE**			**MAE**		
	A_1	A_2	A_3	A_1	A_2	A_3
2	351	347	N/A	273	269	N/A
3	2322	2265	2318	2216	2164	2212
4	819	2462	1343	632	2286	1139
5	1864	1995	1178	1623	1768	935
6	**4962**	**4060**	**4707**	**4774**	**3882**	**4528**
7	1686	766	889	1541	607	697
8	888	2090	2287	829	2012	2234
9	1833	1096	2602	1624	822	2425
10	820	832	919	651	614	690
11	1882	2037	1225	1635	1438	896

As an example, we can consider checkpoint 6. The result of the error values is a calculation of either RMSE or MSE of the runner's real and predicted time values, but only the values pertaining to checkpoint 6. In this approach, we can visualize which algorithms are better, enabling us to analyze the algorithms A_n in a different way to further understand which algorithm is better performing. In the case of checkpoint 6, we can see that the algorithm A_2 had the least error out of the three in both RMSE and MAE metrics.

It's possible to gain a more substantial visual understanding of the patterns within the error values, in which we employed heatmaps as a tool to further elucidate the nuances within error values. The heatmap represents the RMSE and MAE values across all checkpoints. Darker colors indicate higher error values, while brighter colors indicate lower error values. Through heatmap data visualization, we can identify clusters of checkpoints with consistently high or low errors, highlighting areas where runners faced significant challenges or performed exceptionally well.

At first glance, both heatmaps have a similar color pattern, which is hinted towards the proportionality between RMSE and MAE. In regards to which algorithm A_n is best, in consideration with Figure 13 we can take into account the checkpoints where algorithms tend to be most light, in this case algorithm A_2 had the least error out of the three algorithms in checkpoints 2, 3, 6, 7, 9 and 10, A_1 in checkpoints 4 and 8, and A_3 holds the least error in checkpoints 5 and 11.

We can also observe that the error values in each checkpoint can vary by over 1000 seconds in certain algorithms, this is in relation to what context the algorithms hold. A_1 holds the context of

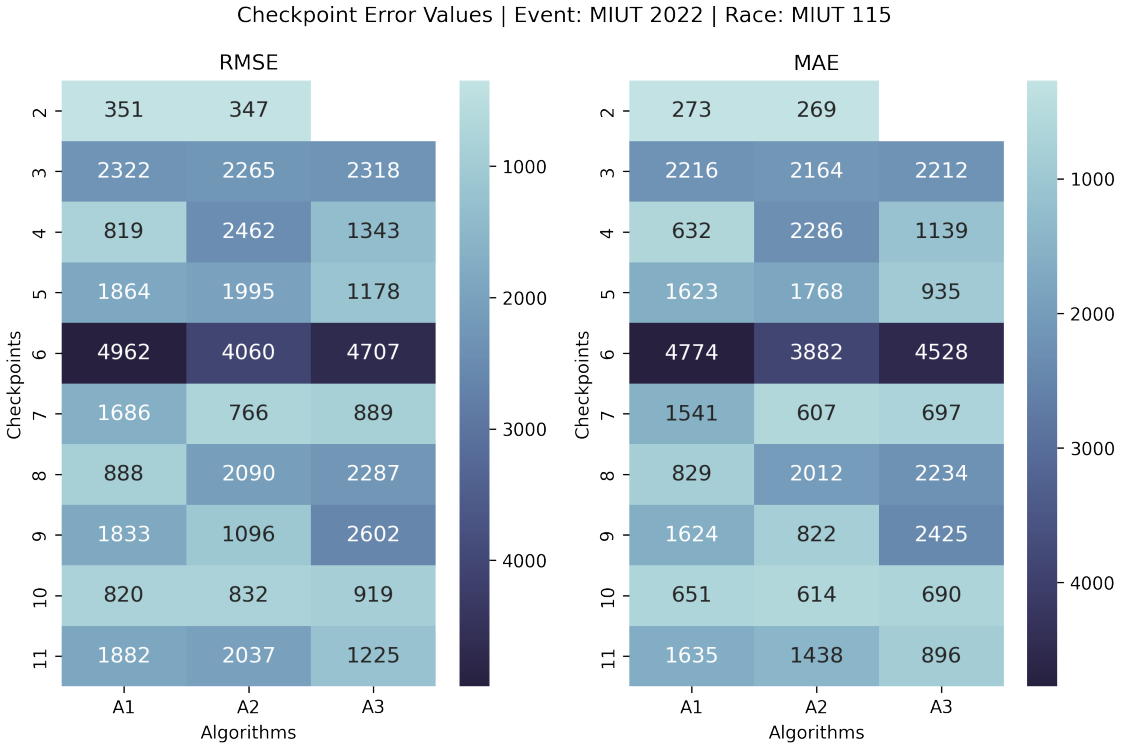


Fig. 13: Checkpoint error values for RMSE and MAE for each algorithm A_n

the checkpoints that the runner has passed, A_2 only holds the context of the previous checkpoint, and A_3 holds the context of the last two checkpoints.

A possible method to optimize the error values would be to purposefully select the algorithm that has the minimum value in each checkpoint in respect to RMSE and MAE. In this regard, at best we obtain an error value of 1346 for RMSE and 1165 for MAE. Comparing with Table 7, if we consider A_2's error value in RMSE (2068), the new minimum value has a 35% ratio difference. For A_1's MSE error value (1580), there is a 25% ratio difference.

The presented error values in Figure 13 between RMSE and MAE do not have major differences in relation to which algorithm is best in each checkpoint, however the RMSE values are always larger than MAE values due to its propensity to penalize heavy outliers. Heavy outliers can be caused by unexpected events and conditions in a trail running race. Considering that at the moment our predictions on our dataset, in it's base level, are based of time and distance, outliers can occur in the case of bad performance from runners, and this in consequence can hinder the representation of whether RMSE or MAE are better at determining an algorithm's error. RMSE will not be considered and this case study will proceeed with the MAE metric.

In the subsequent section, we will examine the accuracy of the algorithms specifically for each runner in the race.

4.3.3 Runner Error Values

An in-depth view of each checkpoint in resource to heatmaps allowed us to obtain new insights and patterns of the error values, and we can employ a similar approach to explore the error values in the context of each runner. In this way, we can look into each individual runner to understand if

there's any noticeable patterns among them. Table 9 is comprised of the first ten runners's MAE error values as the table is too large to fit here.

Table 9 contains the error values of the algorithms associated to each runner in the race. The minimum error value is 716 seconds (00:11:56), from algorithm A_2, and the maximum error value is 1069 seconds (00:17:49), from algorithm A_1. Presenting these min/max values can indicate us that the algorithm A_2 is the best out of the three given that it holds the minimum value from Table 9.

Table 9: Error values of each runner (first ten) for algorithms A_n

Runner	A_1	A_2	A_3
136117	814	609	613
136119	**884**	**746**	**786**
136125	857	737	754
136124	897	716	749
136118	774	890	825
136136	837	734	770
136138	854	640	719
136142	1048	846	942
136133	1069	865	862
136130	835	831	874

As an example, let's consider runner 136119. Considering every prediction in the race of runner 136119's time crossings, and the evaluation of MAE to algorithms A_n, the algorithm A_1 obtained 884 seconds of error (00:14:44), algorithm A_2 obtained 746 seconds of error (00:12:26) and algorithm A_3 obtained 786 seconds of error (00:13:06). Simply by comparing these three algorithms in respect to runner 136119, we can observe that algorithm A_2 was better than A_1 and A_3 at predicting the future checkpoint arrivals of runner 136133.

Table 9 does not have enough samples, and so drawing definitive conclusions becomes exceedingly challenging. The utilization of heatmaps becomes imperative to effectively visualize the error values corresponding to each runner. By employing these error values, new insights and patterns can be discovered, allowing for a more comprehensive analysis. In Figure 14, we can see the averaged error values for each runner in a heatmap.

The top of the heatmap has the lightest shade of values, indicating that the runners that finished first have the most minimum of errors. Across the algorithms A_n, the more we observe the heatmap downwards, the darker the heatmap becomes. This points out to the last finishers having larger error values, a consequence of this is that the longer a runner takes to finish the race (in respect to the first finisher of the race), the algorithms will have more difficulty in predicting when a runner will reach the next checkpoint. This comes in contrast to the difficulties such as fatigue that trail runners face when they do not perform as well as first place finishers, and this leads to an inconsistent performance throughout the race. Consequently, the predictions will also become inconsistent and with significant error values.

Figure 14 suggests that the algorithm A_2 performs the best out of the three algorithms, as the pattern is the last to get darkest over the runners. A_1 is second to A_2 with progressively more darker shades down in the heatmap, but A_3 is even more prone to them.

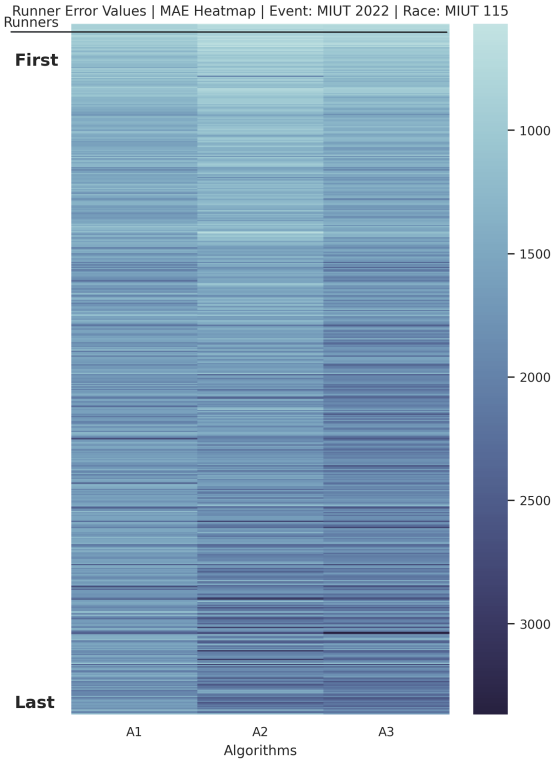


Fig. 14: Heatmap of the runner's error values

If we consider the global error values, we know that the algorithm A_1 has 8 less seconds of error from algorithm A_2. This indicates that A_1 is better than A_2, but in observation to Figure 14, A_2 may be more accurate in predicting the fastest runners of the race, however this is a task for future work.

Upon concluding this section, the algorithms A_n have been analyzed under the scope of only one race, and to build a foundation on these results implies utilizing other MIUT 115km races to further investigate if the same patterns can be seen in other MIUT 115km races. The subsequent section will approach the races MIUT 115km of 2016 to 2022 with the same analysis procedure executed on MIUT 2022 115km.

4.3.4 Long-Distance Races

This case study is about only one race, and we pretend to do the same procedures in this case study to other races in order to further reaffirm our findings.

We chose to analyze the number of runners and finishers to understand which races are meaningful for analysis. Figure 15 presents the registered runners and finishers in a race over the years. The time interval of the collected races span from 2008 to 2022, and the editions of 2010 and 2020 did not take place as they were cancelled. This sets the sample size to 13 races.

The number of registered runners in the race increased over the years with a minimum of 46 runners in MIUT 2011, and a maximum of 950 runners in MIUT 2019 as we can observe on Figure 15. The number of finishers increased in similar fashion to registered runners across the years, where 30 was the minimum number of finishers in MIUT 2009, and 625 was the maximum number of finishers in MIUT 2019.

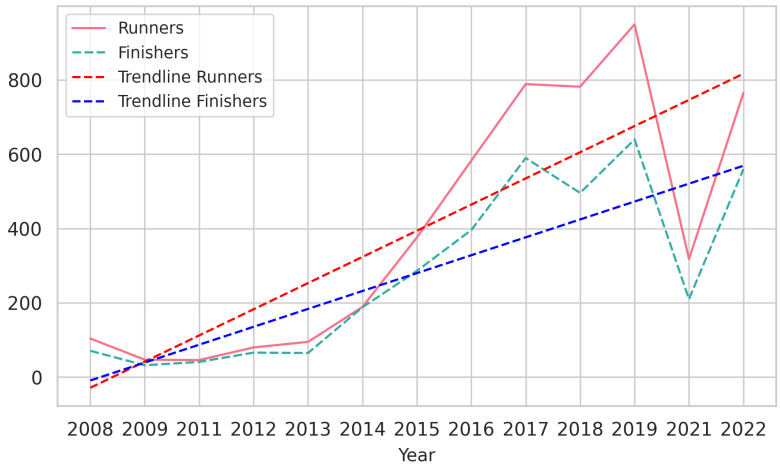


Fig. 15: Number of runners and finishers in MIUT2008-2022 races

We can observe that in the period 2008-2015, the pool of runners was smaller in comparison with the period 2015-2022. This indicates that the race grew in popularity over the years, with exception to 2021 as it was scheduled for November due to the COVID-19 pandemic, contrary to the race usually being scheduled for April. This information provides insights into the scale of the MIUT races over the years.

Figure 15 can indicate which set of races are suitable races for analysis. For that, we selected the last five races from 2008 to 2021.

The five races (MIUT2016-2021) selected will be analyzed following the analysis done to MIUT2022. We can present Table 10 that contains the global error values and seek any similar patterns in respect to MIUT 2022. The light orange color corresponds to the minimum error values in each race and metrics, and the darker orange color corresponds to the maximum values.

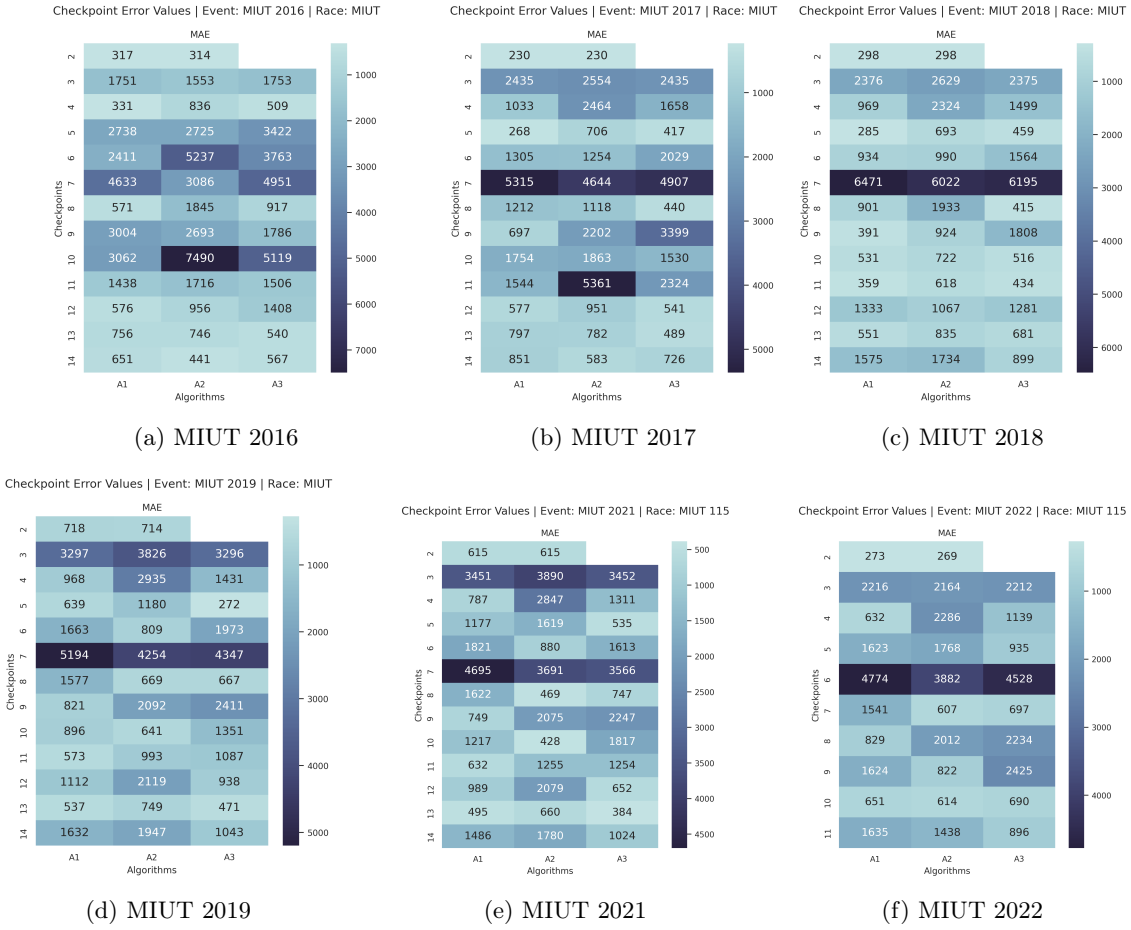
Table 10: Global MAE error values of MIUT 2016-2022

Year	A_1	A_2	A_3
2016	1711	2280	2187
2017	1387	1901	1742
2018	1306	1599	1511
2019	1510	1764	1608
2021	1519	1715	1551
2022	1580	1587	1751
Mean	**1501**	**1807**	**1724**

We can observe that the error values in Table 10 indicate variations in the magnitude of MAE error values within each column. A_2 tends to have the highest values overall, followed by A_3 and A_1. However, the observed patterns are all different from each other.

For further evidence to reaffirm similar patterns from MIUT 2022, we employed the same heatmap as in Figure 13 that contains the race's checkpoint error values. The produced heatmaps are presented in Figure 16.

The patterns of each race in Figure 16 are similar to the patterns seen in MIUT 2022's data, with highlight to checkpoint 7 across MIUT2016-2021 and 6 in MIUT 2022 as it is the checkpoint with most error out of all checkpoints.



(a) MIUT 2016 (b) MIUT 2017 (c) MIUT 2018

(d) MIUT 2019 (e) MIUT 2021 (f) MIUT 2022

Fig. 16: Checkpoint error values for algorithms A_n

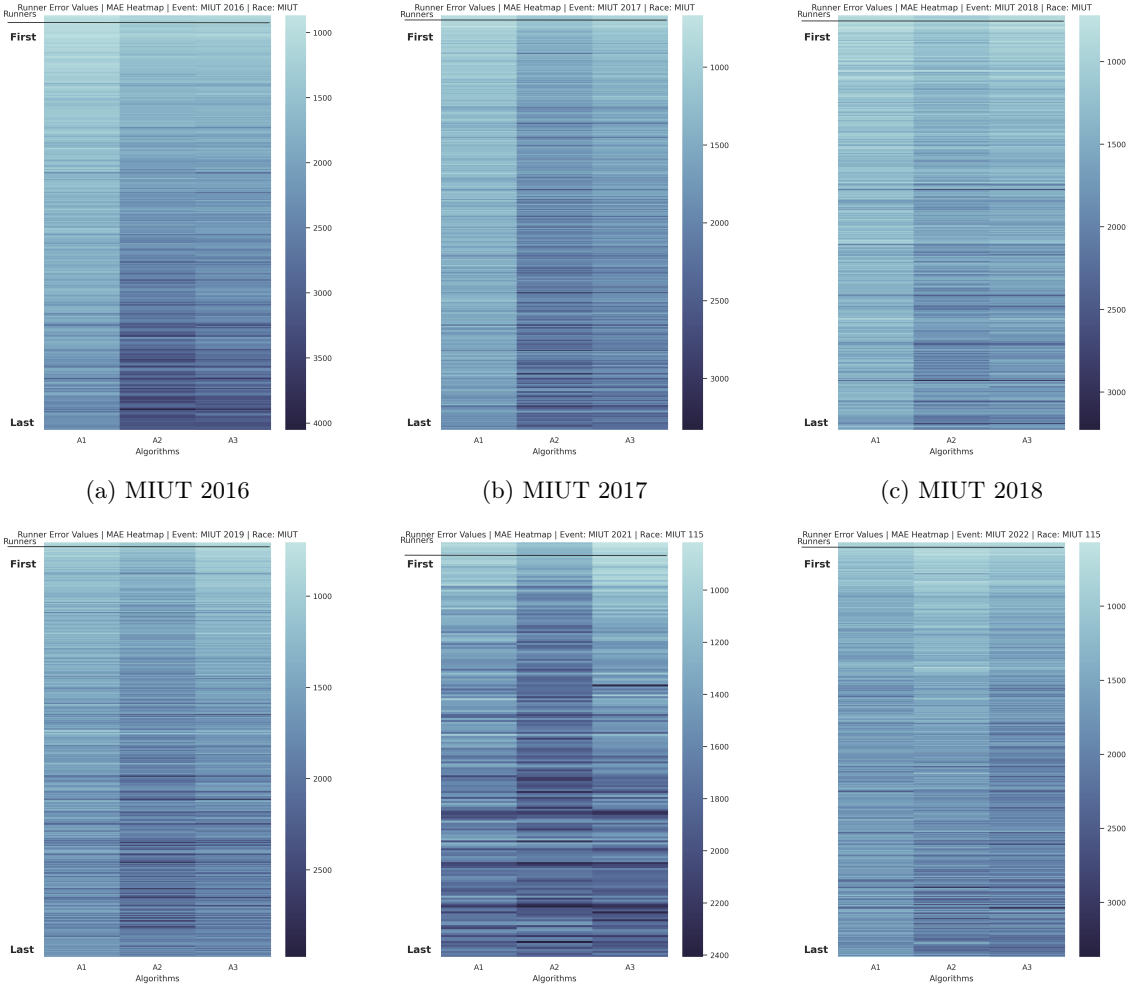
The patterns observed in Figure 16 mostly align with those depicted in Figure 16f, providing further evidence of consistent trends discovered in this analysis.

Having analyzed the checkpoint error values, now we present the analysis done to the runner's error values in Figure 17. We can see that the error values of A_2 tend to be darker than the values of A_1 and A_3. In contrast to the observations done in Figure 17f, we can notice that this is a distinct pattern and has different insights when taking into account other races. Algorithm A_2 is also progressively darker over time much faster than algorithms A_1 and A_3.

Considering that the algorithm A_1 in MAE values are the lightest in Figure 17, we can indicate that the algorithm A_1 carries the least error out of the three algorithms.

By observing that algorithm A_1 consistently exhibits the lowest MAE error values in the runners heatmap, it can be suggested that the algorithm A_1 performed well across the MIUT races, with exception to MIUT 2022, which was found to be the exception as it held algortihm A_2 as having least error out of the three algorithms, although it is noticeable across the races that the first

portion of runners where is most lightest can indicate that performing well in a race as a runner results in less error values.



(a) MIUT 2016 (b) MIUT 2017 (c) MIUT 2018

(d) MIUT 2019 (e) MIUT 2021 (f) MIUT 2022

Fig. 17: Runner error values of algorithms A_n

This analysis in regards to the MIUT2016-2022 races is to further find insights that can be seen in the analysis done to MIUT 2022, as a method to re-confirm our findings.

The error values of the races, as represented by the global MAE, deviate from the pattern observed in MIUT 2022. Specifically, algorithm A_2 performs worse than A_1 and A_3 in terms of error. Algorithm A_3 had a better performance than A_2 as it takes into account the context of the previous two checkpoints, in contrast to one checkpoint as A_2 operates in it's context. If we observe the difference between algorithm A_2 and A_3's error values in MIUT 2017 and 2018, we can see that A_3 contains the least error, although with a very marginal difference in MIUT 2018. The occurence of these differences between the algorithms A_n are often, more so between algorithms A_1 and A_3.

4.3.5 Multiple Distance Races

In this step of case study, the 115km races of MIUT 2016 to 2022 were analyzed, but now an analysis of the shorter MIUT races, namely 85km, 60km, 42km and 16km distances will be done, complementing our previous examination of the MIUT 115km trail running races. By exploring these various race lengths, we pretend to obtain insights about shorter races to understand if the same patterns occur as observed in this section.

Table 11 provides information on the number of participants and finishers for the MIUT 2016-2022 115km races, in order to understand the data distribution between each race edition.

Table 11: Number of runners and finishers in MIUT2016-2022

Total Distance	**Year**	**Runners**	**Finishers**	**Finishers (%)**
115	2016	583	375	-35.68%
	2017	789	584	-25.98%
	2018	782	482	-38.36%
	2019	950	625	-34.21%
	2021	318	210	-33.96%
	2022	765	560	-26.80%
	Mean	**697**	**472**	**-32.00%**
85	2016	280	237	-15.36%
	2017	408	352	-13.73%
	2018	402	341	-15.17%
	2019	458	354	-22.71%
	2021	102	86	-15.69%
	2022	369	329	-10.84%
	Mean	**336**	**283**	**-15.00%**
60	2021	95	91	-4.21%
	2022	386	375	-2.85%
	Mean	**240**	**233**	**-3.00%**
42	2016	551	524	-4.90%
	2017	581	559	-3.79%
	2018	555	522	-5.95%
	2019	604	560	-7.28%
	2021	214	175	-18.22%
	2022	575	562	-2.26%
	Mean	**513**	**483**	**-7.00%**
16	2016	356	354	-0.56%
	2017	433	424	-2.08%
	2018	389	385	-1.03%
	2019	373	368	-1.34%
	2021	269	183	-31.97%
	2022	394	393	-0.25%
	Mean	**369**	**351**	**-6.00%**

Observing Table 11, it indicates that the races have a varied number of runners that competed, with the 115km race having the biggest number of runners and, on the opposite side, the 85km being the least competed race in MIUT 2022. The variation between registered runners and finishers tend to have a larger difference in the 115/85km races and the 60/42/16km have a smaller difference

across the years. This can indicate the difficulty of the race when taking the total distance of the race into account.

The number of registered runners and finishers in respect to the MIUT 2022 races are similar to those in other MIUT races from 2016 to 2021 with identical distances, and to visualize the trend of which algorithm has the least error in each event and race, Table 12 presents the global error values of each race from 2016 to 2022. There is a recurring trend in Table 12 that occurs in different races.

115km and 85km Algorithm A_1 had the least error, with A_2 having most error in most cases.

60km We will not consider this race as the event pool size is too small.

42km and 16km Algorithm A_3 had the least error, although the 16km race could not be calculated for MIUT 2022 as the race only had 2 checkpoints. A_2 had the maximum value in the 42km race, and A_1 in the 16km race.

Table 12: Global error values of algorithms A_n in all races of MIUT2016-2022

Total Distance	**Year**	A_1	A_2	A_3
115	2016	1711	2280	2187
	2017	1386	1901	1741
	2018	1306	1599	1510
	2019	1510	1764	1607
	2021	1518	1714	1550
	2022	1580	1586	1751
	Mean	**1501**	**1807**	**1724**
85	2016	1437	1671	1692
	2017	1570	1977	1682
	2018	1444	1671	1462
	2019	1485	1830	1541
	2021	1452	1880	1419
	2022	1741	1597	1619
	Mean	**1521**	**1771**	**1569**
42	2016	685	1187	621
	2017	937	1748	639
	2018	956	1347	574
	2019	1412	2134	724
	2021	942	1504	513
	2022	1002	1154	696
	Mean	**989**	**1512**	**627**
16	2016	456	418	229
	2017	570	496	339
	2018	832	718	456
	2019	825	770	217
	2021	748	655	272
	2022	338	338	0
	Mean	**628**	**565**	**252**

In consideration to the mean values and min/max values of each race, algorithm A_1 had the least error in the 115km/85km races, and algorithm A_3 in the 42km/16km races. The 60km race requires a larger data size in order to be analyzed.

The main insight to conclude out of Table 12 is that A_1 is more accurate at predicting in most races, especially in long-distance races, and it's noticeable that A_3 can be more accurate only towards short-distance races, although using races that do not have over 2 checkpoints are to be avoided when possible.

4.4 Conclusion

In this chapter, our goal was to explore the predictive capabilities of the chosen methods in determining the time at which a runner would cross their next checkpoint. Additionally, we aimed to establish the trustworthiness of these methods in making accurate predictions. This requires insights into the factors influencing the timing of a runner's progress and to evaluate the effectiveness of the employed predictive algorithms. The dataset for this case study is related to the Madeira Island Ultra Trail 2016-2022 and our analysis revealed several insights.

The findings were related with our research questions, with respect to the analysis of MIUT 2022 and five other MIUT races from the editions 2016-2021. These races had different findings in comparison to MIUT 2022. From observing the global error values, the algorithm A_2 had a large error and algorithm A_1 had less error in every case. Checkpoint 7 (checkpoint 6 in MIUT 2022) still contains the same pattern with high error values, as a consequence of the terrain conditions or other unforeseen events. Consequently, Figure 17 also has A_2 skewed with very dark shades compared to algorithm A_1 and A_3.

If we take into account the mean values of the global error values of the races MIUT 2016-2022 of 115km, the algorithm with minimal error is algorithm A_1, and we consider it to be performing better than A_2, and A_3. Contrary to A_2 and A_3, algorithm A_1 contains the context of the entire race in its prediction until the end of the race. On the other hand, algorithm A_2 contains the context of the last travelled checkpoint, and A_3 is analogous but it carries the last two checkpoints crossed into context.

In the case of multiple distance races, the observations vary between the races. Algorithm A_1 has the minimal error in the 115/85km races and A_3 in the 42/16km races. Algorithm A_2 had the maximal error in 3 out of 4 races. It can be possible to select an algorithm with the minimal error based on the distance of the race.

The five selected metrics had different results in providing insights that would answer our research questions. Through analyzing the global errors of the algorithms A_n, we found that the metrics MSE, R^2, and MAPE do not provide valuable insight towards our research questions. For the rest of this case study, only RMSE and MAE metrics would be employed on the algorithms. For further analysis of these global error values, the same analysis was employed for each checkpoint and to each runner, respectively. In order to visualize the data effectively, we opted for heatmaps, which allowed us to detect any prominent patterns or insights.

In regards to the evaluation of the five selected metrics, these yielded varied results in addressing our research questions. After analyzing the global errors of algorithms A_n, it was discovered that metrics such as MSE, RMSE, R^2, and MAPE did not contribute valuable insights to our investigation. Consequently, for the remainder of this case study, the focus shifted exclusively to utilizing MAE metrics in conjunction with the algorithms. Furthermore, a thorough examination of these global error values was conducted at both the checkpoint and individual runner levels.

To facilitate effective data visualization, we opted for heatmaps, enabling us to identify significant patterns and gain valuable insights from the data.

The heatmaps indicate that there is a large error at checkpoint 6, which can be attributed to the race segment's terrain conditions that can affect a runner's performance on the trail. In regards to the algorithm's error values, these oscillate within a certain range in each checkpoint, which may be able to be reduced through additional features of the runner's or race's characteristics. A possible method to optimize the predictions is to purposefully select the best predictions out of the three algorithms and use them in combination. With respect to the analysis of the error values of the runners, the heatmaps indicate that the algorithm A_2 performs well in the case of runners who finished the race first compared to the last finishers. In contrast to algorithm A_1, this pattern is slightly darker than A_2, and algorithm A_3 contains an even more darker shade. If we take into account the global error values of the algorithms, we can consider that A_2 performed best in the case of RMSE, but A_1 performed best in the case of MAE, and this inconsistency led to omitting RMSE from this case study to avoid uncertainty on deciding which algorithm was more accurate. Therefore, the heatmaps of the error values of the checkpoints would suggest that the algorithm A_1 was more accurate out of the three algorithms.

A method to improve these findings is to take into account more features of the race, associated with the race's terrain profile, the weather conditions or any specific data point that may cause new insights into our research questions. As future work, a model could be created in order to determine which of the A_n algorithms will have the least error in a race.

The next chapter is a second case study to utilize machine learning techniques to further improve our findings related to our research questions.

5 Case Study: ML for prediction of the MIUT 115km races

In our previous case study, we explored three algorithms that calculate the prediction of a runner's arrival at the next checkpoint of a race. However, our findings indicated the potential for further improvement by incorporating additional features related to the race's terrain profile, runner's performance, and other specific data points that could provide new insights into our research questions. The motivation for the use of machine learning techniques surges from its uses in real-world applications [11], particularly in the domain of sports, where researchers have employed [31,36,37] machine learning techniques in marathon events to improve forecasting of athletes performance and racing bib number recognition. The improvement of forecasting of athletes' performance is indirectly related to our research questions, as obtaining predictions of when an athlete will reach his/her next checkpoint in a trail running race can be an indicative of how well performing the athlete is in a given race.

The initial exploration of literature related to machine learning and trail running did not yield significant results as a fast search could not indicate any relevant literature related to our scope of work and research questions. The literature searches were conducted to understand if there is research in the literature that can align with our scope of work, but it confirmed that there is limited research in the literature. Marathon racing was found to have works that can inspire us towards our research questions, but marathons cannot be equated to trail running, even more so if ultramarathons ($\geq$ 42km) were addressed for the majority of this project.

The literature review in regard to machine learning presented three paradigms of machine learning, supervised learning, unsupervised learning, and reinforcement learning. Supervised learning will be explored throughout this case study as the provided data is labelled and there are specific prediction tasks to execute. A basic architecture [3] within a supervised learning model, presented in Figure 3 on subsection 2.1.2 page 8, is initiated by partitioning the collected dataset into training and testing data, selecting the preferred features for the algorithm and the model is trained to learn the features associated with each label. With the test data, the model can make predictions on the test data by providing the expected labels.

With respect to trail running and machine learning in section 2.3, the section contains one work [26] that was found to align with our research questions is the trail running assessment of performance (TRAP) framework. The authors employed four models for prediction: Linear Regression, LASSO, Random Forests, and Extreme Gradient Boosting (XGBoost). XGBoost was used to predict the probability of a runner to withdraw from the race, but this is beyond our scope of work. The focus will be directed towards predicting the runner's next crossing at a checkpoint.

The conclusion of the previous case study pinpointed the need for more features related to the race, and machine learning models can fit this requirement to process a larger number of features. This case study aims to utilize machine learning techniques to expand upon our findings and address the identified limitations. The incorporation of a more comprehensive set of race features can enhance the accuracy and predictive capabilities. Specifically, we will focus on exploring how accurately we can predict a runner's next arrival at the checkpoint of a race and what is the most accurate model under this scenario.

The dataset is the collection of MIUT 115km races from 2016 to 2023, in order to compare with the results from the previous case study as a point of reference in regards to global MAE errors of each race. Features will be quantitatively augmented in order to build new capabilities that allow

machine learning models to learn from a higher dimensional dataset. Such models are designed to handle high-dimensional datasets.

The sections of this case study will: **1.** present the selection and preprocessing of features, **2.** approach machine learning models to test and evaluate the features, **3.** designate the validation steps of the models, **4.** display the results of the models and their respective feature importances, and **5.** reflect this case study in comparison to the previous case study.

5.1 Features

This case study continues the investigation of our research questions, of how accurately we can predict a runner's passage time of the next checkpoint in the context of trail running races. The authors in [5] describe features as data that describe properties of the data points, which can be categorical, ordinal, or numerical. The identification of the data type of each feature is a step to understand how to represent features, in order to obtain features that enhance the machine learning models through capturing patterns and relationships in the data that can bring new insights. Data quality and bias are two other aspects in the literature that are fundamental to understand and manage the absence of data and the magnitude that each feature has an effect on the bias of a machine learning model as this can directly impact the accuracy and interpretability of the results. A novel framework [26] for trail running assessment performance, where it constitutes two main sets of features.

Checkpoint-based features These features are linked to attributes associated with checkpoints of a race. They include details of the distance to the checkpoint, elevation, and the type of checkpoint itself.

Runner-based features These features are related to the characteristics of the runner participating in the event. They contain demographic information about the runner as well as historical data aggregated from their past races.

The described set of features captures a range of insights related to the trail running race dynamics, runner attributes, and the progression of the race through the trail checkpoints. This framework was developed under the context of predicting a runner's next arrival at his/her checkpoint and the likelihood that a runner will drop out of the race at the next checkpoint arrival. Our research questions primarily concern the former, involving the accuracy of predicting a runner's passage time in a given checkpoint within a trail running race, as well as the choice of the most accurate prediction algorithm and the optimal method for evaluating its performance.

The features for this case study were organized in a similar method as the researchers in [26] did so, given that their work is directly related to trail running and the prediction of a runner's next arrival to his/her checkpoint. Table 13 presents the set of features aggregated from the dataset.

The set of features in Table 13 is comprised of 14 checkpoint-type features and 26 runner-type features, 40 features (30 numerical, 10 categorical) in total. These features characterize the race properties that a runner has competed and finished, among with properties connected to the runner's demography background and past rank data that can deepen the model's knowledge of the data itself. Runner-type categorical features were subject to feature engineering in order to make an impact on the model's ability to learn from data in categorical form through the use of data preprocessing techniques, in particular One-Hot Encoding and Feature Hasher [41], explained next. One-Hot Encoding is a technique that converts categorical variables into a binary vector

Table 13: Set of features grouped by Checkpoint-type and Runner-type

#	**Type**	**Feature**	**Description**
C101	Numerical	distance	Distance from checkpoint
C102	Numerical	accumul_distance	Cumulative distance from race start
C103	Numerical	altitude	Altitude of checkpoint
C104	Numerical	plus_elev	Positive elevation from checkpoint
C105	Numerical	minus_elev	Negative elevation from checkpoint
C106	Numerical	accumul_plus_elev	Cumulative positive elevation from race start
C107	Numerical	accumul_minus_elev	Cumulative negative elevation from race start
C108	Categorical	time_barrier	If there is a time limit
C109	Categorical	AP	If personal assistants are present
C110	Categorical	L	If liquids and beverages are present
C111	Categorical	+	If first aid kit is present
C112	Categorical	S	If food is present
C113	Categorical	Méd.	If first aid medical support is present
C114	Categorical	MR	If runner can change clothes
R101	Categorical	sex	Sex
R102	Categorical	category	Age category
R103	Categorical	nationality	Nationality
R104	Numerical	total_elev	Total elevation of previous races
R105	Numerical	mean_elev	Average elevation of previous races
R106	Numerical	total_dist	Total distance of previous races
R107	Numerical	mean_dist	Average distance of previous races
R108	Numerical	min_dist	Minimum distance of previous races
R109	Numerical	max_dist	Maximum distance of previous races
R110	Numerical	mean_elev_dist	Average rate of elevation by distance of previous races
R111	Numerical	max_elev_dist	Maximal rate of elevation by distance of previous races
R112	Numerical	min_elev_dist	Minimum rate of elevation by distance of previous races
R113	Numerical	n_races	Number of races the runner has participated
R114	Numerical	avg_competitors	Average number of competitors in previous races
R115	Numerical	years_since_first_race	Time in years since first MIUT race
R116	Numerical	year_of_last_race	Time in years since last MIUT race
R117	Numerical	months_since_last_race	Time in months since last MIUT race
R118	Numerical	avg_female_percentage	Average percentage of female runners in previous races
R119	Numerical	perc_time_overall	Average of $\frac{\text{runner's time - time of first position}}{\text{time of last position - time of first position}}$ in previous races
R120	Numerical	perc_time_first	Average of $\frac{\text{runner's time - time of first position}}{\text{time of first position}}$ in previous races
R121	Numerical	perc_time_last	Average of $\frac{\text{time of last position - runner's time}}{\text{time of last position}}$ in previous races
R122	Numerical	rank_perc_vs_elev_dist	Average of $\frac{\text{elevation}}{\text{distance}} \times \frac{\text{ranking}}{\text{avg\_competitors}}$ in previous races
R123	Numerical	mean_rank	Runner's average ranking considering previous races
R124	Numerical	mean_rank_perc	Runner's average of $\frac{\text{ranking}}{\text{\# of participants}}$ considering previous races
R125	Numerical	max_rank_perc	Runner's maximum $\frac{\text{ranking}}{\text{\# of participants}}$ considering previous races
R126	Numerical	min_rank_perc	Runner's minimum $\frac{\text{ranking}}{\text{\# of participants}}$ considering previous races

representation, and this was applied to features *sex* (R101) with 2 binary vectors and *category* (R102) with 15 binary vectors. The feature *nationality* (R103) was processed through Feature Hasher, to avoid a large number of features derived from one feature, as it contains 67 different categories. Checkpoint-type categorical features are in binary categorical form, so it is not necessary to preprocess them further.

Features R104 to R126 are a subset of the features in Table 13 that were derived from the runner's past[21] participation in MIUT races. These derivations include cumulative sums, averages, ratios, and differences applied to various attributes like elevation, distance, ranking, participation metrics, and finishing times. The primary objective of feature engineering is to provide a more comprehensive understanding of runners' historical performance, participation trends, and ranking dynamics.

This section has presented a diverse set of dependent features through a series of aggregations that comprehend the aspects related to the participants' historical race context and the geographical setting of the trail running races. With these enhanced features, we transition into the subsequent section, where machine learning models are explored in order to be employed in the data presented in this section.

5.2 Models

In the previous subsection, Table 13 displays a set of features, with conditions that can determine what models are more adequate to be carried out in this analysis. Table 14 also presents a statistics summary of these features within the entire year period. The data is high-dimensional, has non-linear relationships as there are features that do not scale together along with other features, and is subject to interactions between features that correlate. The model(s) to be employed must be related to regression-based supervised learning models [3], as our dataset is labelled, has continuous values, and the expected predicted output is continuous values.

In connection to our research questions, these are related to obtaining information on the accuracy of these tools to predict a runner's next checkpoint arrival time. The summary description in Table 14 provides a snapshot of the dataset's statistical properties, including the number of distinct values, mean, standard deviation, minimum, first quartile (25th percentile), median (50th percentile), third quartile (75th percentile), and maximum values for each feature. This can aid in the decision of model selection, leading us to choose models that are robust to skewed distributions, outliers, and the relationships associated between each feature.

Table 14: Statistical summary of the features in the dataset

Feature	Distinct	Mean	St. Dev.	Min	25%	50%	75%	Max
distance	45	8286.52	3317.10	3600.00	5300.00	8100.00	10800.00	15300.00
accumul_distance	57	62268.21	30286.36	12400.00	35900.00	67800.00	87500.00	116200.00
altitude	18	870.75	570.53	5.00	312.00	837.00	1496.00	1740.00
plus_elev	55	563.88	541.73	0.00	80.00	415.00	935.00	1515.00
minus_elev	59	578.69	339.66	0.00	305.00	420.00	815.00	1380.00
accumul_plus_elev	66	5087.54	2125.79	1487.00	3030.00	6080.00	6958.00	7854.00
accumul_minus_elev	70	4227.96	2308.33	360.00	2385.00	4275.00	6373.00	8364.00

[21] It is not entirely improbable that multiple runners are unintentionally grouped as a consequence of runners with the same names.

Table 14 contains some features of the complete set of the data as the table size would be too large to accommodate all features. The remaining features of the dataset can be found in Chapter 7.1, section A.

Features `distance` and `accumul_distance` features both offer insights into distance-related aspects and are measured in meters. The feature `accumul_distance` demonstrates a wide range of numbers, from 12400 to 116200 meters, with a mean of 62268.21 meters, as it comprehends a race of 115km. These features collectively reflect the dataset's diverse distance measurements.

Elevation-related features, namely `altitude`, `plus_elev` and `minus_elev` provide insights into terrain and altitude variations with measurement in meters. The feature `altitude` showcases a range of 5 to 1740 meters, with an average of 870.75 meters, indicating varied elevation levels. The features `plus_elev` and `minus_elev` have a mean value of 563.88 meters and 578.69 meters, respectively. These features reveal a spectrum of elevation values, capturing the geographical contexts. The accumulated versions, `accumul_plus_elev` and `accumul_minus_elev` are complements of the singular versions as they portray the race's elevation description over the race.

In TRAP [26], the research questions are related to our research questions, and it identified both LASSO and Random Forests as capable of processing high-dimensional data. Two of the four models (LASSO and Random Forest) will be employed in this case study in order to focus on the predictions of the runner's next checkpoint arrival in a race with the LASSO and Random Forest approaches. The feature importance of the models will be analyzed to understand which features are given a larger importance to the models. The authors used RMSE to calculate the error values, but this case study will proceed with the use of MAE to understand how accurate we can predict a runner's next checkpoint arrival time and if it is the best method to evaluate the accuracy and performance of the model.

LASSO [27] and Random Forests [29] will be used throughout the case study. LASSO acts as a base level reference to compare with Random Forests and these two models are within the set of regression-based supervised learning models. LASSO is a technique that incorporates L1 regularization, which functions by adding a penalty term to the regression cost function. This penalty is proportional to the absolute values of the model's coefficients. The model is encouraged to reduce the magnitudes of less important coefficients to zero, which is another form of feature selection. Random Forests is an ensemble learning method that creates multiple decision trees to make predictions. Each tree is trained on a different subset of the data and features, and the final prediction is obtained through aggregation (averaging or majority voting) of individual tree predictions. This approach reduces overfitting and enhances generalization. Random Forests is suitable for capturing non-linear relationships, interactions among features, large datasets and is robust to overfitting. The feature importance of Random Forests will be presented in this case study as a means of capturing what features influence the model in its decision-making process.

In the subsequent section, model validation techniques will be presented in order to assure that the data is reliable for analysis for this case study. This validation stage serves as a process between the model development and the real-world deployment of models.

5.3 Model Validation

In order to improve the credibility of the results of this case study, we will use more than 1 race for the analysis. The data to be analyzed comprises of 7 MIUT 115km races, from 2016 to 2023,

consisting of 45196 time crossings and 3488 finishers. Table 15 presents the number of samples (time crossings in checkpoints), the number of runners/finishers and the difference of finishers with runners in each race.

Table 15: Count of runners and finishers in the year period of 2016-2023

Year	Samples	Runners	Samples FIN	Finishers
2016	6416	583	5250 (-1166, -18.17%)	375 (-208, -35.68%)
2017	9541	789	8176 (-1365, -14.31%)	584 (-205, -25.98%)
2018	8499	782	6748 (-1751, -20.60%)	482 (-300, -38.36%)
2019	10751	950	8750 (-2001, -18.61%)	625 (-325, -34.21%)
2021	3550	318	2940 (-610, -17.18%)	210 (-108, -33.96%)
2022	7102	765	6160 (-942, -13.26%)	560 (-205, -26.80%)
2023	8392	891	7172 (-1220, 14.54%)	652 (-239, -26.82%)

The data in Table 15 has varied over the years, and we can observe the mean values of finishers as they indicate a skewed distribution of finishers. Two races (MIUT 2016/2021) have a lower number of finishers, whereas the other five races (MIUT 2017/2018/2019/2022/2023) have a larger amount of finishers than the mean value associated. The percentage difference between the number of finishers and runners is between -25.98% to -38.36% over the year period of 2016 to 2023.

An accumulated view of the runners/finishers is provided in Table 16 to obtain more insights about the data. We can observe the data growth and difference over the year period.

Table 16: Accumulated of samples and number of runners and finishers in the year period of 2016-2023

Year	Accumulated Samples	Accumulated Runners	Accumulated Samples FIN	Accumulated Finishers
2016	6416	583	5250 (-1166, -18.17%)	375 (-208, -35.68%)
2017	15957	1372	13426 (-2531, -15.86%)	959 (-413, -30.10%)
2018	24456	2154	20174 (-4282, -17.51%)	1441 (-713, -33.10%)
2019	35207	3104	28924 (-6283, -17.85%)	2066 (-1038, -33.44%)
2021	38757	3422	31864 (-6893, -17.79%)	2276 (-1146, -33.49%)
2022	45859	4187	38024 (-7835, -17.08%)	2836 (-1351, -32.27%)
2023	54251	5078	45196 (-9055, -16.69%)	3488 (-1590, -31.31%)

The accumulated samples and number of runners have a maximum number of 54251 samples and 5078 runners, respectively. In regards to the finishers, the maximum number of samples and the number of finishers is 45196 samples and 3488 finishers. Throughout the years, the samples of finishers and the number of finishers have percentage changes between -15.86% to -18.17%, and -30.10% to -35.68%.

As a means to compensate for the number disparity of finishers over the years since 2016, we will join the two feature sets approached in subsection 5.1. The concatenation of both feature sets can enhance the context that the dataset represents, as it combines both the context of the race's checkpoints and profile, along with demographic data and historical data of the runner.

Authors of the TRAP framework conducted [26] the leave-one-year-out (LOYO) cross-validator (CV) on data from the UTMB (Ultra Trail du Mont Blanc) races starting from training on years 2015 up to 2017 and to test on 2018. We will carry out the same cross-validator and temporal ordering of the data as otherwise this breaks the temporal dependency of the annual races. In this way, this guarantees that each year is tested exactly once in an iterative manner. Evaluating the models was done through the analysis of the mean absolute error (MAE) of the results presented in Section 5.4).

Figure 18 illustrates the process of Leave-One-Year-Out Cross-Validation. The dataset is divided into multiple folds, each representing a year's worth of data. In each iteration, the model is trained on all but one year's data and tested on the remaining year. This process is repeated for each year, allowing us to assess the model's performance across different temporal periods.

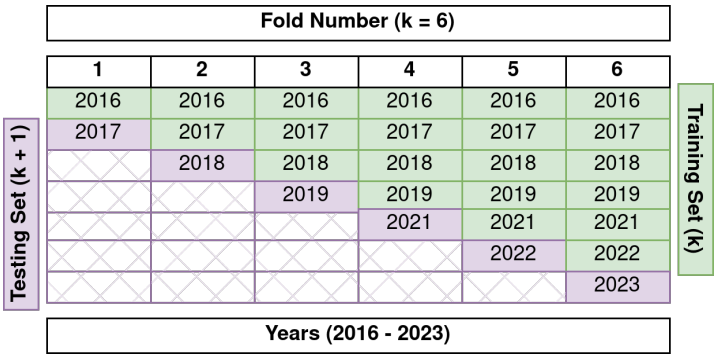


Fig. 18: Illustration of the training and testing sets in each fold number in LOYO CV

The number of folds is obtained through $k-1$ (k = number of years) folds. Figure 18 has $k = 7$ years, so we require 6 folds for the cross-validator.

Hyperparameter optimization is a validation step used to adjust a model's hyperparameters to enhance performance. Optimization can be conducted through brute-force selection of hyperparameters, but this approach is computationally intensive as it processes every possible combination of hyperparameters. Instead of utilizing the model with every possible combination, we approached a randomized search which randomizes hyperparameters within a given domain and reduces the time required for optimization computation. Table 17 describes and shows the search space of four parameters for the Random Forest model.

Table 17: Hyperparameter description and search space

Model	Hyperparameter	Description	Search space
RF	n_estimators	The number of trees in the forest	(10, 500)
	max_depth	The maximum depth of the tree	(1, 20)
	min_samples_split	The minimum number of samples required to split an internal node	(2, 20)
	min_samples_leaf	The minimum number of samples required to be at a leaf node	(1, 20)

The randomized search was executed with 5 iterations, with the least MAE error obtained of 838 seconds with the optimal parameters: `n_estimators`: 56, `max_depth`: 12, `min_samples_split`: 9 and `min_samples_leaf`: 10.

The next section will present the results and analysis of the MAE errors of the selected models for this case study in order to obtain insights related to our research questions. A subsection is also presented in respect of feature importance to understand what features have the most effect on the result.

5.4 Results & Analysis

This section presents the results of the four models, in regards to the MAE metric, conducted by data from the year period 2016 to 2023 of the MIUT 115km race. The two models were employed to train and test on two different feature sets: the Checkpoint and Checkpoint + Runner feature sets, in order to obtain improved MAE errors in reflection of Chapter 4. Table 18 presents the MAE and accuracy results of the predictions done to the MIUT 2023 115km race.

Table 18: Results of the models

Model	Feature Set	MAE	Accuracy (Train Set)	Accuracy (Test Set)
LASSO	Checkpoint	7141	0.882	0.880
	Checkpoint + Runner	5070	0.968	0.929
RF	Checkpoint	6916	0.886	0.885
	Checkpoint + Runner	1134	0.998	0.995

In Table 18, the results are presented with mostly errors above the 5000-second mark. The Checkpoint feature set indicates that the MAE errors of 7141 (LASSO) and 6916 (RF). RF netted less error than LASSO, but the error magnitude is larger than the results of the Checkpoint + Runner feature set. In this case, the MAE errors are 5070 (LASSO) and 1134 (RF). The MAE error of RF is significantly smaller than LASSO, and it is the least MAE error achieved. There is a noticeable pattern in regard to the accuracy of each set between both feature sets.

Given that an MAE error of 1134 seconds was achieved, we will employ the same model and feature set with 3 iterations in order to confirm if we can further improve this error, and also present the mean values of these 3 iterations for each race edition. Table 19 displays the MAE errors and accuracies.

Table 19: Results of the Random Forest model with 3 iterations

Year	MAE	Train Accuracy	Test Accuracy
2016	1358	0.998	0.996
2017	1302	0.998	0.995
2018	1621	0.998	0.991
2019	883	0.998	0.998
2021	1135	0.998	0.995
2022	839	0.998	0.998
2023	1129	0.998	0.995
Mean	**1181**	**0.998**	**0.995**

Table 19 presents the average MAE values for each predicted year in succession, with each successive year accumulated as training data to use the next year as testing data. In the case of testing the race edition from 2019, the training set has time passages from the year 2016, 2017, and 2018. This netted an error of 883 seconds, which is a substantial improvement from the previous tested years (2016-2018). The tested years after 2019 do not display a very significant improvement, but the errors and accuracies do not oscillate so much as the years 2016 to 2018, with errors from 1302 to 1621 seconds. From a global point of view, the mean value of the averaged values of Table 19 is 1181 seconds, which overall is an improvement in regards to the global error values achieved in Chapter 4. Under this scenario, it's suggested that increasing the amount of data and features with the employment of Random Forests can improve the MAE error and train/test accuracies.

For further investigation, we wanted to employ the two models against the same dataset from Chapter 4 in order to understand what results can occur out of a dataset with a very reduced set of features. Table 20 exhibits the MAE results for each of the two models.

Table 20: Results of the models with data from Chapter 4

Model	**MAE**	**Accuracy (Train Set)**	**Accuracy (Test Set)**
LASSO	4310	0.964	0.960
RF	1293	1.000	0.992

RF has an error of 1293 seconds which is significantly smaller than the error of 4310 seconds in LASSO, and has accuracies of train set and test set of 1.000 and 0.992 respectively. The accuracies of the two models are similar, with RF being more accurate than LASSO.

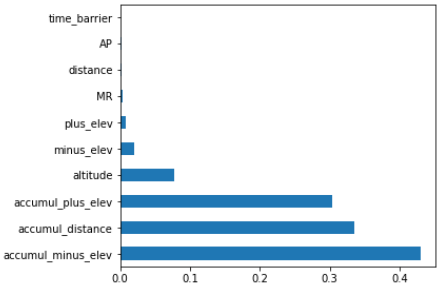
In observation of Table 18 and Table 20, it can be seen that the feature set Checkpoint returned large errors in comparison to the feature set Checkpoint + Runner, that has the least MAE error of 1134 seconds, from the RF model with accuracies of the train set and test set: 0.998 and 0.995, respectively. The results associated with the data from Chapter 4 do not have lower errors than the minimum MAE error result in Table 18, but it may be interesting to look into why the result of 1293 seconds in Table 20 is the only error in proximity of the error of 1134 seconds (159 seconds of difference) from the feature set Checkpoint + Runner. Unlike the two features sets in Table 18, the amount of features analyzed in Table 20 is 5 features only, compared to the 14 and 40 features of the Checkpoint and Checkpoint + Runner feature sets, respectively.

The subsequent subsection will illustrate the importance attributed to each feature out of the results associated to the Random Forest model.

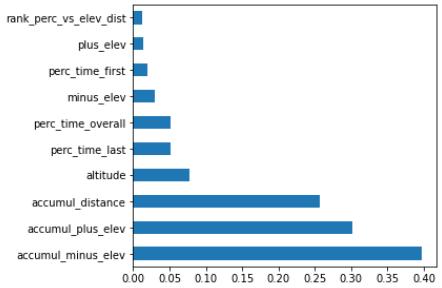
5.4.1 Feature Importance

This subsection will analyze the feature importances of the results from the Random Forests model in order to observe how the model attributes significance to each feature. Figure 19 presents the first ten features with most importances out of the results displayed in Tables 18 to 20 in Section 5.4.

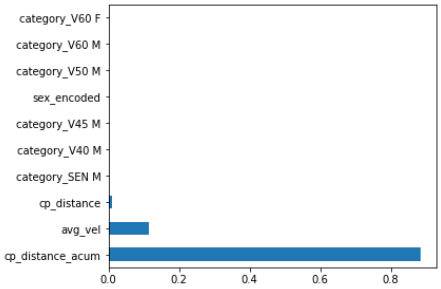
Figure 19a indicates a significant importance to the first three features, such features are associated with positive and negative accumulated elevation, and the accumulated distance of the race, similar to what Figure 19b represents with the first three features. 4 out of 10 features presented



(a) Checkpoint (14 features)



(b) Checkpoint + Runner (40 features)



(c) Chapter 4 data (5 features)

Fig. 19: Importances of features for each feature set analyzed with Random Forests

in Figure 19b are related to runner-type features, which are in majority within the corresponding feature set, with 14 checkpoint-type features and 26 runner-type features.

Figure 19c presents a different view on the importance of features, with the first feature (`cp_distance_acum`) having more than double the importance of the first three features in Figures 19a and 19b. The feature `avg_vel` was placed in second place, which can hint at the use of a runner's average velocity at a crossed checkpoint to improve the prediction of a runner's time crossing at his/her next checkpoint.

This subsection presented observations towards the importance of the features of each feature set in order to extract insights about what features Random Forest relies on in its prediction process. It can be observed that a large portion of the importance was allocated to checkpoint-type features, regardless of the feature set. This implies that the runner-type features are not assembled correctly, although this can be caused by the small number of runners edited in order to match with duplicate runners. This hinders the historical view that the model will have on the runners which in consequence can affect the results of the model.

5.5 Conclusion

This chapter approached machine learning techniques as a tool to answer the questions left in the conclusion of Chapter 4, but also to continue our research questions, primarily how accurate can we predict a runner's passage time in a given checkpoint within a trail running race?

The assembled features for this case study are composed of two sets with 14 checkpoint-type features, and the same 14 checkpoint-type features joined with 26 runner-type features in order to try and maximize the potential of the features, based on the literature [26] that has similar research questions to this project.

The model selection for this case study was done through the consideration of the high-dimensional and non-linear relationship nature of our data. This consideration brought about the deployment of two models: LASSO and Random Forests. With respect to the statistical summary presented, there is a possibility that the magnitude of distinct values of checkpoint-type features may be influencing the predicting process of the models. Feature engineering can reduce the number of distinct values within the features, but this can reduce the meaning that the data holds, which in turn will decrease the interpretability of the models.

The model validation step, in this case study, displayed data of the proportion of finishers throughout the year period 2016-2023 of the MIUT 115km races, as the Leave-One-Year-Out Cross-Validation was employed as a means to analyze the race data over the years cumulatively. Hyperparameters of the Random Forest were also set up within a randomized search of four hyperparameters, but this part of the model validation can be enhanced in the future to allow for more optimization of the hyperparameters.

The MAE error values presented in this case study show that the Checkpoint feature set had high magnitudes of errors, 7141 and 6916 seconds for the LASSO and RF models, respectively. On the side of the feature set Checkpoint + Runner, the errors are 5070 seconds for LASSO and 1134 seconds for RF. We can observe that the RF model had the least error within this case study, and also when compared against the MAE error values in Chapter 4, with the minimum mean value being 1501 seconds from Algorithm A_1. This noticeable improvement in the MAE error can incline us to further explore the use of Random Forests and raise the size of data and features, as the model also achieved MAE errors under 1000 seconds in the MIUT 115km race editions of 2019 and 2022. The feature importances related to these two feature sets highlighted mostly features related to checkpoint-type features, despite having 14 features when compared against the runner-type features, which sum up to 26 features. It is possible that the data has to be altered in order to eliminate the problematic of duplicate runners since many of them are identified as new runners.

As an experiment with the data from Chapter 4, we also employed LASSO and RF, and the error value from RF is 1293 seconds, which is lower than other error values in this case study, with the exception of RF on the feature set Checkpoint + Runner. With respect to the annotated feature's importance in this case study, it can be observed that the average velocity calculated for each runner's time crossing is relied on by the model in order to compute the predictions. As future work, the average velocity can be implemented as another feature (or features) to enhance the capabilities of our dataset for analysis.

This chapter has approached the MIUT 115km races only, but we pretend to obtain more insights from other races in order to understand if similar results tend to occur so that we can respond more easily towards our research questions.

The subsequent chapter will employ the same methodology, the Random Forests models of this chapter, as it performed significantly better than the LASSO model. The data to be approached is of larger quantities in order to examine what results occur within a larger set of races of different distances.

6 Case Study: ML for prediction of multiple distance races

This chapter will delve in-depth into races of different distances of MIUT and races that do not belong in the MIUT event, to further investigate if races of different distances can improve the error results with the use of Random Forests and the full set of checkpoints and runner-related features, as the combination of these in Chapter 5 proved to be an improvement in respect to the MAE error values when compared to LASSO and the presented results of Chapter 4. In order to obtain predictions of time passages with less error, we are extending the pool size of races to comprehend if any significant improvements can occur based on more quantity of data.

The exploration of trail running races outside of the 115km race from MIUT can provide new insights and patterns about the use of Random Forests in predicting races under the 115km total distance mark. This approach encompasses a larger pool of races with various runners and checkpoints, with both ultramarathon and non-ultramarathon races included in this case study. Ultramarathon races have different race profile characteristics when compared to non-ultramarathon races, and considering that they can be distinct from one another, these races can be aggregated in an effort to eliminate the heterogeneity of the data, with the result of having various training data sets, with each set representing a certain level of trail running level. In this manner, we can analyze what effects the compilation and aggregation of data can have in relation to the MAE error values.

In order to predict a larger pool of races, all of the presented races in this case study will be organized into their training sets according to each race's total distance and total positive elevation. The organization of training data based on the race's total distance and total positive elevation can lead to training data that is more homogenous, which may suggest the improvement of the MAE errors.

The sections of this case study will: **1.** present the classification of endurance level for trail running races, **2.** predict the arrival of runners in MIUT races by endurance level, **3.** predict the arrival of runners in MIUT races and other races by endurance level, **4.** predict the arrival of runners in MIUT races without endurance level, **5.** predict the arrival of runners in MIUT races and other races without endurance level, **6.** discuss the results obtained in this case study, and **7.** reflect this case study in comparison to the previous case study.

6.1 Classification of races by distance and positive elevation

This section will approach a method to classify races according to their distance and elevation, in order to have homogenized sets of races with an approximate description of their endurance level.

The ITRA Points system [42] is a method to estimate the endurance level of a race, and is considered as proof of a runner's experience to achieve races of a certain level. Race organizers can request runners to provide this information as a pre-requisite to register for a race. ITRA Points are calculated based on the notion of "Km-Effort". The total Km-Effort is calculated by taking into account the distance and elevation gain of a race in the form of:

- Distance: 1km = +1 Km-Effort
- Elevation: 100m = +1 Km-Effort

The Km-Effort formula can penalize a race depending on the number of aid stations, and the average interval between two aid stations is calculated as follows:

$$\text{Average Interval} = \frac{\text{accumul\_distance} + (\text{accumul\_plus\_elev}/100)}{\text{aid\_stations}} \tag{18}$$

Table 21 represents how many Penalty Points a race can have, according to the Average Interval obtained out of equation 18.

Table 21: Average Interval intervals and their respective penalty points

Penalty Points	0	10	15	20	25	30
Average Interval	[13, $+\infty$)	[11, 12.99]	[9, 10.99]	[7, 8.99]	[5, 6.99]	($-\infty$, 5]

After calculating both the Average Interval and Km-Effort, we can subtract the Km-Effort obtained from the race's total distance and total positive elevation with the Penalty Points obtained from the Average Interval. This outputs the Km-Effort with an adjustment from the Penalty Points.

The Km-Effort of each race is then classified from 0 to 6 ITRA Points, as Table 22 illustrates:

Table 22: ITRA Points group classification and their Km-Effort

ITRA Points	0	1	2	3	4	5	6
Km-Effort	[0, 24]	[25, 44]	[45, 74]	[75, 114]	[115, 154]	[155, 209]	[210, $+\infty$)

To facilitate the naming reference of each ITRA Points group classification presented in Table 22, the designation I_n where n corresponds to ITRA Points, is attributed to each group classification. For example, 0 ITRA Points correspond to I_0, and 6 ITRA Points correspond to I_6, where I_0 are the easiest races and I_6 are the hardest races classified by the ITRA Points system.

The following section will approach the set of MIUT races from 2008 to 2023 classified under the ITRA Points system that evaluates the endurance level of a race as a method of aggregating a larger pool of races that will result in having their own race characteristics in common, in order to use race data from MIUT within the years 2008 to 2022 as training data and to utilize each individual race of MIUT 2023 as testing data for the prediction of arrival of runners.

6.2 Prediction of arrival in MIUT races by endurance level

This section seeks to continue from the previous chapter, by adding more MIUT races and taking advantage of the ITRA Points system described in the previous section. The classification of the endurance level of such races without a system like the ITRA Points can compromise the judgment of difficulty, as it's very likely that the MIUT races have been altered across the years since its first edition in 2008. The use of the ITRA Points can facilitate the classification of the endurance level of the races, as in this manner we are relying on a race's total distance, positive elevation, and the number of aid stations to classify the endurance level of a race.

We also intend to use MIUT races of different distances in this case study to extend our understanding of the Random Forest model in the case of races with different distances, as we had already explored the 115km race of MIUT 2023 in the previous chapter with Random Forest. This requires us to gather all MIUT races from 2008 to 2023, which totals 53 races, however some races

were found to have missing data related to the geographical profile and therefore we must omit 14 races from this case study as this can significantly impact our results. In this situation, the pool size of races for this section is 39 MIUT races from 2008 to 2023, with total distances ranging from 16km to 115km. A description of each endurance level I_n with respect to the race's total distance and positive elevation characteristics can give us insights in regards to how extensive one race can be on average in each endurance level. This is presented in Table 23.

Table 23: Descriptive statistics of the number of races, accumulated distance and accumulated positive elevation in each endurance level

Endurance Level	**# Races**	**Total Distance**				**Total Positive Elevation**			
		Min.	**Max.**	**Mean**	**St. Dev.**	**Min.**	**Max.**	**Mean**	**St. Dev.**
I_6	∅	∅	∅	∅	∅	∅	∅	∅	∅
I_5	9	101	115	113.44	4.66	6090	7854	7121.22	554.43
I_4	11	85	105	89.64	8.03	3525	5339	4652.55	608.25
I_3	5	53	60	57.20	2.95	2165	2980	2653.00	413.39
I_2	6	42	50	43.33	3.27	1685	1924	1780.33	90.07
I_1	∅	∅	∅	∅	∅	∅	∅	∅	∅
I_0	8	16	20	16.75	1.39	262	390	372.00	44.80

Table 23 presents an aggregation of races according to their endurance level, as observed in the distance and positive elevation scaling of each successive endurance level. The total of races sum to 39 races, with 51% of the races having at least 85km total distance and 3525m of positive elevation. The remainder 49% are races with a total distance between 16 to 60km, and positive elevation between 262m to 2980m. The set with the most races corresponds to I_4 and the least are the sets I_1 and I_6 that contain two and zero races, respectively. Of all races, 79% of the races correspond to ultramarathon races.

From these 39 MIUT races, 5 of the races correspond to the latest edition of MIUT 2023, and under the ITRA Points system, these were attributed to the following endurance difficulties: **1.** 115km - I_5, **2.** 85km - I_4, **3.** 60km - I_3, **4.** 42km - I_2, and **5.** - I_0. Given that there are not enough races from MIUT 2023 to compensate for the 7 endurance difficulties, we are omitting I_6 and I_1 as they cannot be included in this case study. In this scenario, we can see the MIUT 2023 races have distinct endurance difficulties, meaning that each race does have distinct characteristics between each of the 5 races. The number of training and testing sets for this case study is equal to the number of the MIUT 2023 races, as we intend to train and test data that corresponds to their respective endurance level. As an example, if we wish to predict the time passages of the runners in the 42km race of MIUT 2023, then we must train the time passage data of races whose endurance level is classified as I_2. Table 24 presents an overview of the sample size and number of runners and validated runners of each I_n.

In Table 24, the endurance level sets I_5 and I_4 have 73% (69413 samples) of the total validated data (94512 samples), which portrays the data as mostly data related to ultramarathon races that account for 79% for the number of races that will be assessed in this case study to predict the MIUT 2023 races. The data was processed on the Random Forest model. For reference, it had the least MAE error results (1181 seconds) in Chapter 5 whose data in the case study was the MIUT 2023 115km race. This case study continues to not just use the 115km race data in the previous chapter, but also the remaining 4 races of MIUT 2023 with distances of 85km, 60km, 42km, and 16km, respectively. The model was employed on each endurance level with five times

Table 24: Number of races in each endurance level and their size of sample before and after validation

Endurance Level	**# Races**	**Samples Runners**	**Runners**	**Validated Samples**	**Validated Runners**
I_5	9	61217	5250	**46619** (-23.85%)	3607 (-31.29%)
I_4	11	28122	2725	**22794** (-18.95%)	2279 (-16.37%)
I_3	5	7161	927	5950 (-16.91%)	859 (-7.34%)
I_2	6	15288	2576	12292 (-19.60%)	2433 (-5.55%)
I_0	8	9541	2638	6857 (-28.13%)	2529 (-4.13%)
Total	39	121329	14116	94512 (-22.12%)	11707 (-17.06%)

for correctness, and the results are calculated in average terms. Table 25 presents the results of the MAE error and accuracies.

Table 25: MAE results of each I_n, in average with 5 tests

Endurance Level	**Race Test**	**MAE Value**			**Train Time**	**Test Time**	**Train Accuracy**	**Test Accuracy**
		Min.	**Max.**	**Avg.**				
I_5	115km	1209	1222	1217	9.977	0.134	0.998	0.997
I_4	**85km**	1703	1712	**1709**	4.001	0.071	0.998	0.987
I_3	60km	892	896	894	1.748	0.230	0.995	0.989
I_2	42km	447	453	450	1.255	0.045	0.994	0.993
I_0	**16km**	299	302	**301**	3.074	0.122	0.995	0.942

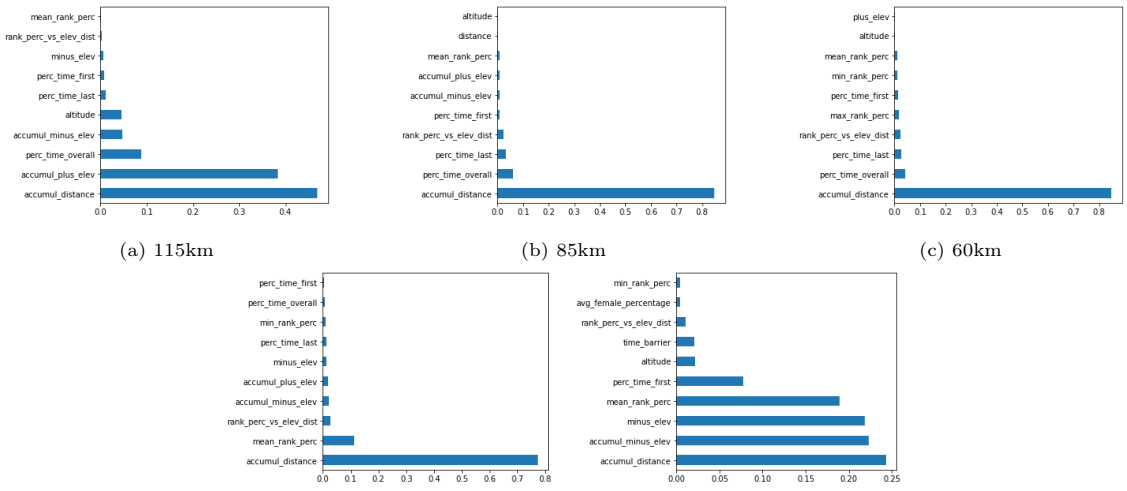
In Table 25, the minimum and maximum of the test accuracy attained correspond to I_0 and I_5, respectively. In nominal terms, the endurance level I_0 had the least error of 301 seconds, but this does not imply that the MAE error in level I_0 is an improvement over the MAE error in level I_5, given the train, test accuracies presented in Table 25. The level I_4, has the maximum error of 1709 seconds, while I_5 has a smaller error of 1217 seconds, despite the tested race on level I_5 corresponding to a 115km race. The time to train I_4 and I_5 is considerably longer than the reminder I_n sets, and the time to test does not follow a specific pattern. If we take into account both the process of training and testing the data, the time to train comprises at least 87% of the total time process of both training and testing the data.

In this section, we covered the MIUT events from 2008 to 2023 and aggregated the races according to their endurance level with the basis on the ITRA Point system to have a better understanding of each race's endurance level to more adequately assimilate each race's characteristics into their own context in each endurance level, thereby resulting in five different training sets that correspond to a certain MIUT 2023 race, in order to train data that has similar characteristics to the MIUT 2023 races. The use of the MIUT 2023 edition allows us to consistently use the edition as a benchmark example for prediction, specifically in regards to the results associated with the error values of the checkpoints and runners for the MIUT 2023 115km race in previous case studies. The next section will present the feature importance of the features for each I_n.

6.2.1 Feature Importance

This subsection will present the features with the most importance attributed by the model, with respect to the last prediction of each corresponding race of MIUT 2023. The list of features con-

ducted in this Chapter can be found in Table 13, Section 5.1. Figure 20 illustrates the feature importance of the tested races.



(a) 115km (b) 85km (c) 60km

(d) 42km (e) 16km

Fig. 20: Feature Importances of the tested races

We can observe that `accumul_distance` is the feature with the highest importance in the five races. In Figure 20a, the feature has at least 0.4 of importance. In the case of Figures 20b and 20c, the feature obtained at least 0.8 of importance, Figure 20d and 20e presents the feature with at least 0.7 and 0.2 of importance, respectively. Figure 20e, illustrates four features with at least 0.1 importance, with `accumul_distance` having the most importance, but is the smallest importance of the feature `accumul_distance` in Figure 20, indicating that the distance of the race may not be as important as seen in other tested races in the case of the 16km race. The presence of checkpoint-type or runner-type features is sparse within the most important features presented in Figure 20, but the feature `accumul_distance` is a checkpoint-type feature that holds the most importance in Figure 20, and tends to have a feature importance between 0.4 to 0.8 in most cases, while Figure 20e is an exception to the rule.

The next section will employ the same method, although not only with MIUT races but with races outside of the MIUT event, in order to verify if the results observed in Table 25 can occur under a larger set of races.

6.3 Prediction of arrival in all races by endurance level

This section follows in an analogous form as the previous section but with the inclusion of races from events outside of MIUT to understand if we obtain similar results as observed in the previous section. The total number of races gathered for this section totalled 275 races, with 102 races purged due to missing geographical data, as well as 34 races purged as these were classified with endurance level I_1, and as referred in the previous section, none of the MIUT 2023 races were classified with endurance level I_1.

Table 26 presents a description of each I_n with respect to its number of races aggregated and the variety of race distances and positive elevation.

Table 26: Descriptive statistics of the number of races, accumulated distance and accumulated positive elevation in each endurance level

Endurance Level	# Races	Total Distance				Total Positive Elevation			
		Min.	Max.	Mean	St. Dev.	Min.	Max.	Mean	St. Dev.
I_5	12	100	115	110.33	6.93	5630	7854	6941.83	746.95
I_4	15	68	105	86.19	9.03	3525	5672	4852.12	597.84
I_3	21	44	60	51.91	4.98	2165	4121	3460.91	555.00
I_2	23	23	50	38.21	7.82	1685	3917	2580.17	715.88
I_0	**63**	3	25	10.73	5.72	95	1641	639.92	357.71

In Table 26, the description of each I_n is more dispersed in regards to the total distances and total positive elevations. The number of races in each endurance level I_n has been significantly increased, in highlight to level I_0 with 63 races and level I_5 with the least amount of races, 12 races. The level I_0 includes 47% of the total amount of races in this section (139 races), with the remaining levels containing between 23 to 12 races. The mean and standard deviation attributes reveal that there is more variety of races in each I_n with respect to the total distance and positive elevation.

In Table 27, the number of races, samples and runners are presented for endurance levels I_n.

Table 27: Number of races in each endurance level and their size of sample before and after validation

Endurance Level	# Races	Samples Runners	Runners	Validated Samples	Validated Runners
I_5	12	61858	4633	**40561** (-34.43%)	**3387** (-26.89%)
I_4	15	31214	2743	21912 (-29.80%)	2343 (-14.58%)
I_3	21	23830	2157	15118 (-36.56%)	1879 (-12.89%)
I_2	23	24139	3152	16947 (-29.79%)	3041 (-3.52%)
I_0	63	38362	5921	**21463** (-44.05%)	**5581** (-5.74%)
Total	139	179403	18606	116001 (-35.34%)	16231 (-12.76%)

The total amount of samples and runners shrunk by 35.34% and 12.76%, respectively. The data is also disperse as it is rather more concentrated on endurance levels I_0 and I_5, and the percentage difference in data between non-validated and validated data oscillates between 29.80% to 44.05% and 3.52% to 26.89% in respect to the number of samples and runners. In terms of sample size of passages, at least 65% of the 116001 validated samples are related to ultramarathon races, despite I_0 having the most amount of races (63 races).

In Table 28, we present the error values of endurance levels I_n and its accuracy values.

Table 28: MAE results of each I_n, in average with 5 tests

Endurance Level	Race Test	MAE Value			Train Time	Test Time	Train Accuracy	Test Accuracy
		Min.	Max.	Avg.				
I_5	**115km**	**1066**	**1075**	**1070**	12.471	0.140	0.999	0.997
I_4	85km	1798	1803	1801	6.098	0.105	0.998	0.986
I_3	60km	820	830	825	2.111	0.061	0.997	0.990
I_2	**42km**	**498**	**512**	**504**	3.941	0.088	0.992	0.991
I_0	**16km**	**842**	**883**	**861**	1.716	0.027	0.990	0.631

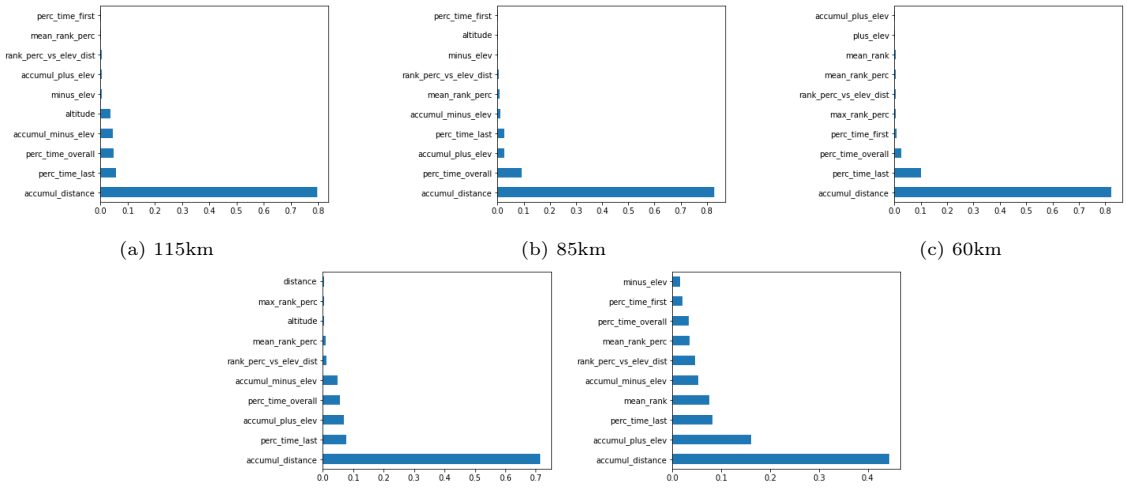
In Table 28, the race with the lowest MAE error corresponds to the 42km race and the 85km race has the highest MAE error out of the five endurance levels, with an error of 1801 seconds and it is a trend previously observed in Section 6.2.

The endurance level I_5 has an improvement of 147 seconds over the error result observed in the previous section. On the other extreme point of endurance levels, I_0 has a larger error by 560 seconds from the previously observed error in the previous section, which can be implied by the data size difference between analyzing exclusively MIUT races or both MIUT races and other races. With respect to the error scaling to each endurance level, this trend is less evident, more so due to the larger error of the endurance level I_0. The test accuracy of the same endurance level was also revealed to be 0.631, which is considerably less than the test accuracy observed in the previous section (0.942).

In this section, we included the races outside of MIUT into this approach, with the endurance level classification of races by the ITRA Points system in order to experiment and observe what new insights could be obtained with a larger amount of races to serve as additional training data. The significant observation in this section is the error increase with respect to the endurance level I_0 indicated in Table 28, and the 147 second error reduction of the endurance level I_5 error value. The next section will present the feature importance of the features of each predicted race of this section.

6.3.1 Feature Importance

This subsection will present the features with the most importance attributed by the model. The list of features conducted in this Chapter can be found in Table 13, Section 5.1. Figure 21 illustrates the 10 features with the most importance accordingly to the endurance levels I_n.



(a) 115km (b) 85km (c) 60km

(d) 42km (e) 16km

Fig. 21: Feature Importances of the tested races

The illustrated Figure 21 indicates that the Figures 21a, 21b, 21c and 21d have at least 0.7 of importance attributed to the feature `accumul_distance`, with the remaining importance being distributed not more than 0.1 to the remaining features. In Figure 21e, `accumul_distance` holds the importance of at least 0.4, and the referred race continues to present as the race with least

feature importance in the feature `accumul_distance` and has more feature importance dispersed in other features.

The next section will not use the ITRA Points system for endurance-level classification of races. In this way, we will have one single training set with the MIUT races, in order to investigate if the results can be improved.

6.4 Arrival prediction in MIUT races (without levels)

This section will approach the organization of the training sets in a different method, instead of organizing the data in endurance levels, we will have one single training set containing all of the MIUT races that will be used to predict each one of the five races of MIUT 2023, in order to understand if the results can have an improvement without the ITRA Points system endurance level classification.

In Table 29, we illustrate the descriptive statistics of the race's total distance, positive elevation and total race count.

Table 29: Descriptive statistics of the number of races, accumulated distance and accumulated positive elevation

	Total Distance				**Total Positive Elevation**			
# Races	**Min.**	**Max.**	**Mean**	**St. Dev.**	**Min.**	**Max.**	**Mean**	**St. Dev.**
39	16	115	68.90	36.23	262	7854	3645.95	2504.22

Table 29 presents only one set with 39 races, with mean values indicating that this set mostly corresponds to ultramarathon races. The calculated mean of the total distance of the set is 68.90 and the total positive elevation mean value is 3645.95, which places the context of the data well into ultramarathon trail running.

The number of samples and runners of this section is represented in Table 30.

Table 30: Number of races and their size of sample before and after validation

# Races	**Samples Runners**	**Runners**	**Validated Samples**	**Validated Runners**
39	121329	14116	94450 (-22.15%)	11701 (-17.11%)

Within the presented data in Table 30, we can observe that it is the same data size as the complete sum of the races analyzed in Section 6.2. In this part of the case study, the data is trained and then will predict each MIUT 2023 race, similar to the previous sections. The obtained results from the model can be observed in Table 31.

In Table 31, we can examine that the error values associated with the 85km race (939 seconds) and the 16km race (200 seconds) are smaller than the error values observed in the previous section. The remaining races have larger errors, with the 60km error value having the largest difference. The time to train the data is 24 seconds, as the training data used is composed of just one training set. In regards to the test accuracy, it is noticeable that the 60km race has decreased its accuracy, and the 16km race still tends to have the least test accuracy out of the five tested races.

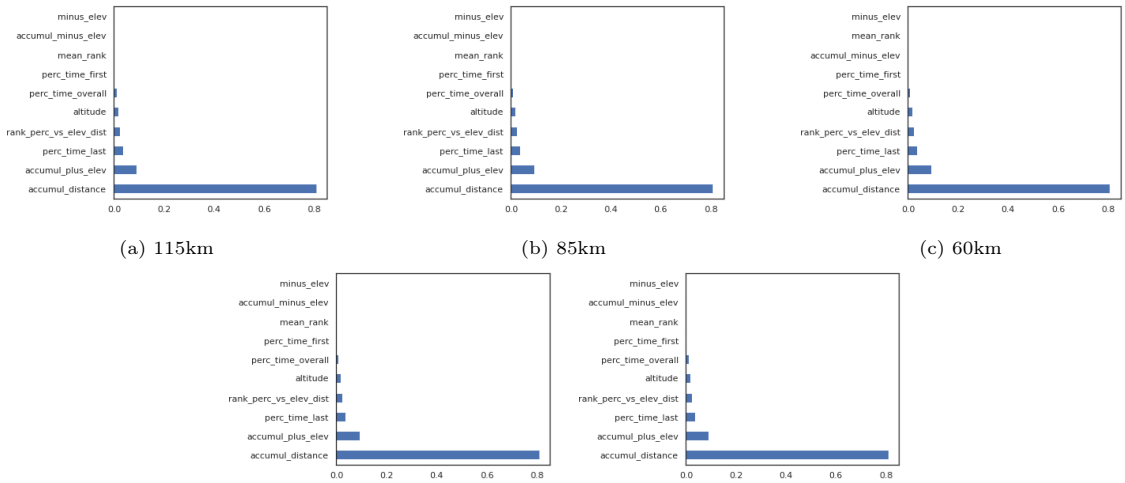
Table 31: MAE results, in average with 5 tests

Race Test	MAE Value			Train Time	Test Time	Train Accuracy	Test Accuracy
	Min.	Max.	Avg.				
115km	1254	1263	1260	24.564	0.141	0.999	0.996
85km	**936**	**942**	**939**	24.680	0.167	0.999	0.995
60km	1381	1405	1390	24.659	0.146	0.999	0.971
42km	427	429	428	24.611	0.164	0.999	0.993
16km	199	202	200	24.627	0.132	0.999	0.950

In this section, we structured the race data without the classification of endurance level in races according to the ITRA Points system in order to experiment and observe if the results of the model would have an improvement. A notable observation in this section is the error reduction with respect to the 85km race indicated in Table 31. The next section will present the feature importance of the features related to this section.

6.4.1 Feature Importance

This subsection will present the features with the most importance attributed by the model. The list of features conducted in this Chapter can be found in Table 13, Section 5.1. Figure 22 illustrates the 10 features with the most importance.



(a) 115km (b) 85km (c) 60km

(d) 42km (e) 16km

Fig. 22: Feature Importances of the tested races

In Figure 22, we can observe that the feature `accumul_distance` holds the majority of the importance, with the remaining features holding a value of less than 0.1. A majority of the importance attributed to are very similar and homogeneous, as the training set used in this section was the same training set employed for all five tested races.

The next section will examine all races available for training without the classification of endurance level of races and will use only one single training set to predict the five races of MIUT 2023.

6.5 Arrival prediction in all races (without levels)

This section is analogous to the previous section, but the training data is extended to all races, not just the MIUT races. This training data will be used to predict each one of the five races of MIUT 2023, in order to understand if the results can have an improvement without the ITRA Points system classification.

In Table 32, we present the total distance and total positive elevation characteristics of the races.

Table 32: Descriptive statistics of the number of races, accumulated distance and accumulated positive elevation

	Total Distance				**Total Positive Elevation**			
# Races	**Min.**	**Max.**	**Mean**	**St. Dev.**	**Min.**	**Max.**	**Mean**	**St. Dev.**
139	3	115	39.58	33.77	95	7854	2470.62	2095.73

In observation, the data totals 139 races, with a mean value of 39.58 regarding the total distance, and the total positive elevation has a mean value of 2470.62. The mean value of the total distance hints that many of the races are not ultramarathon races.

In order to further understand the size of the data in this section, we present Table 33.

Table 33: Number of races and their size of sample before and after validation

# Races	**Samples Runners**	**Runners**	**Validated Samples**	**Validated Runners**
139	179403	18606	124945 (-30.36%)	14102 (-24.21%)

In the displayed sample and runners size in Table 33, the data count is the same as the presented data in Section 6.3. In this part of the case study, the data is trained and then proceeded to predict the passage times of each MIUT 2023 race. The obtained results of the model can be observed in Table 34.

Table 34: MAE results, in average with 5 tests

Race Test	**MAE Value**			**Train Time**	**Test Time**	**Train Accuracy**	**Test Accuracy**
	Min.	**Max.**	**Avg.**				
115km	1834	1845	**1842**	48.739	0.170	0.999	0.992
85km	1433	1449	1444	49.240	0.166	0.999	0.989
60km	742	754	**748**	48.797	0.121	0.999	0.990
42km	594	601	596	48.494	0.198	0.999	0.988
16km	118	124	**122**	48.623	0.114	0.999	0.984

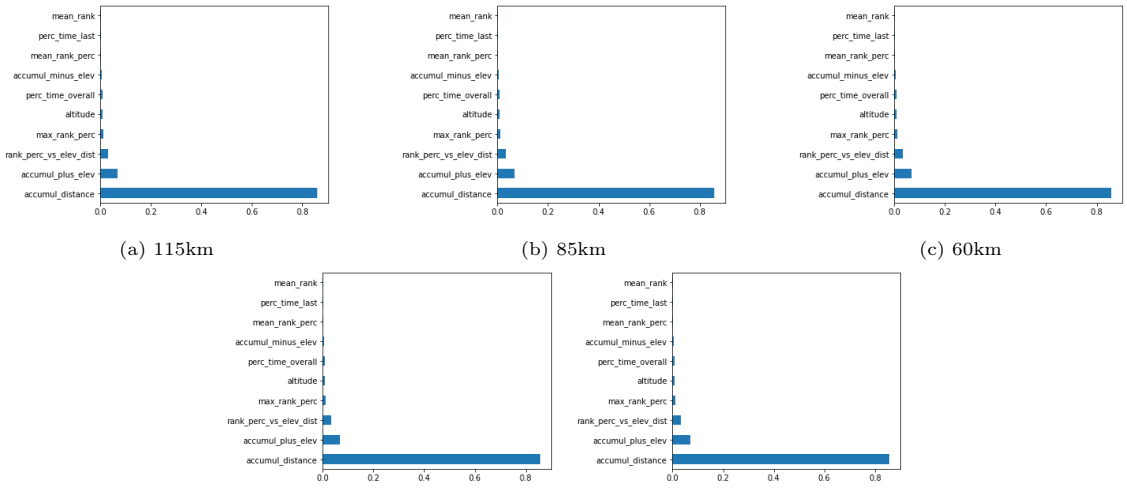
In Table 34, we can observe that the error values associated with the 115km race, of 1842 seconds, is the largest MAE error obtained in this chapter regarding the 115km race. In regards to the 60km, the obtained MAE error was 748 seconds, the lowest MAE error associated with the 60km race of this case study. The 16km race also reached the lowest MAE error of the correspondent

race, of 122 seconds. The time required to train the data is of 48 to 49 seconds, as the training data employed was built with just one training set.

In this section, we employed one training set with all of the races available without any endurance level classification, to predict each of the five MIUT 2023 races to understand if the results of the model could have further improvements. This section's results highlight the 60km race and 16km race as having the least MAE errors in this case study, although the 115km race has the largest error regarding the prediction done to the 115km race throughout this case study. The next section will present the feature importance of the features related to this section.

6.5.1 Feature Importance

This subsection will present the features with the most importance attributed to Random Forests. The list of features conducted in this Chapter can be found in Table 13, Section 5.1. Figure 23 illustrates the 10 features with the most importance.



(a) 115km (b) 85km (c) 60km

(d) 42km (e) 16km

Fig. 23: Feature Importances of each tested race

In Figure 23, the feature `accumul_distance` holds most of the importance value, and the remainder features below `accumul_distance` have a value of less than 0.1. A majority of the importance attributed to are very similar and homogeneous, as the training set used in this section was the same training set employed for all five tested races.

The next section will present a discussion about the results and achievements of this case study.

6.6 Results & Discussion

This chapter has presented 4 sections (sections 6.2 to 6.5) that have the same objective: to predict the runner's time passages over the course of a trail running race, in particular the 5 races of MIUT 2023, with a wide variation of total distances, from 16km to 115km and total positive elevations, from 390m to 7100m. The course of development of this chapter led first to the presentation of the ITRA Points system, as a means to classify a race's endurance level based on the total distance, positive elevation, and number of aid stations. The contribution of this classification leads to better

knowledge about how we can aggregate races into their corresponding characteristics. The previous chapter has utilized the MIUT 2023 edition as a benchmark for the prediction of the runner's time passages, and this chapter continues the trend, by including the remainder 4 MIUT races of the 2023 edition, which in turn included four more testing sets, each corresponding to their ITRA Points, based on their race profile. The ITRA Points system represents 7 endurance levels, although two endurance levels, I_1 and I_6 were omitted due to the absence of a MIUT 2023 race with 1 or 6 ITRA Points, therefore making it impossible to use the referred endurance levels for this chapter.

In every section of this case study, we presented the description of the total distances and total positive elevation of the race's profile, in order to have a clear notion of what kind of races the data in each section represents. That is, if the race data is mostly related to ultramarathons or non-ultramarathons, this is a significant characteristic that can potentially affect the bias of the data. A description of the amount of raw data and post-processed data is provided in order to understand the distribution and skewness of the sample sizes of the races. After the description of the sample sizes, the Random Forests results are presented to analyze the MAE errors and to do a brief comparison to discover which setups perform best with respect to their returned MAE errors. The feature importance is displayed for every section, to better know what features are most relevant to the Random Forest model.

For a deeper understanding of the findings of this chapter, we present an analysis employed to the checkpoints and runners of the runner's time passage prediction for each testing set. The visualization of MAE errors on average for each checkpoint of a race and also for each runner competing in a race allows us to have an atomic view of where the error tends to be more concentrated. This was similarly done in Chapter 4, although the chapter did not include the MIUT 2023 event in the case study as the event had occurred before the case study was written.

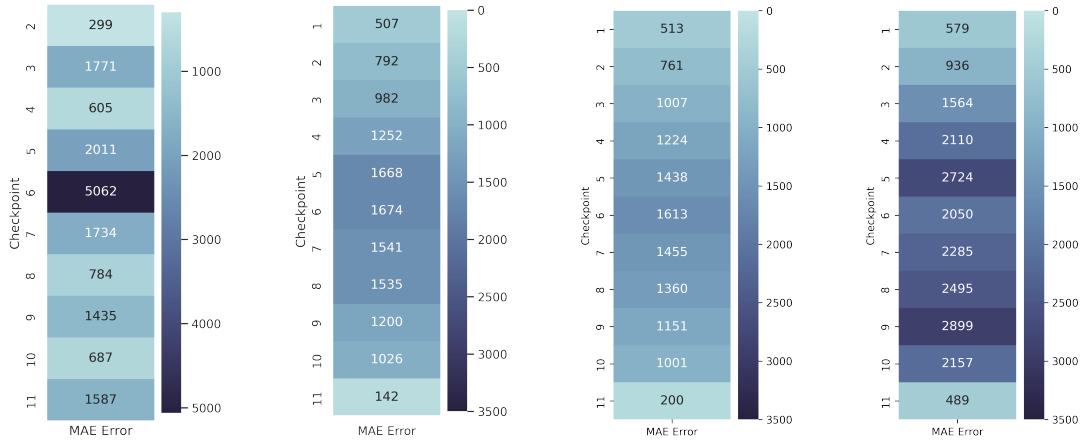
The next subsection will present the visualization of averaged MAE errors with respect to checkpoints and runners of the races that were used as testing sets in this case study.

6.6.1 Longest Distance Analysis

This subsection presents a visualization of the MAE errors obtained on average for each checkpoint and runner with respect to the MIUT 2023 115km race that was utilized as a testing set in this chapter. As this type of visualization was carried out in Chapter 4, we will repeat it for Chapter 5 and 6, in order to have a means to compare the average MAE errors, which can provide further insights of which case study has the minimal MAE error.

Figure 24 presents an overview of the MIUT 2023 115km checkpoint error values represented in four heatmaps, corresponding to the first case study, second case study, and third case study, although we're displaying the minimum and maximum MAE error value cases. The remaining two heatmaps in respect to Section 6.2 and 6.4 are presented in Appendix B, Section B.1 and B.3, respectively.

As illustrated in Figure 24, Figure 24a does not account for CP1, as it only begins to predict crossings times at CP2. The largest error within Figure 24a is 5103 seconds at CP6, and has the rest of the errors dispersed in each successive checkpoint, with even numbers (an exception to CP6) having the least error and odd numbers having the most error. In Figure 24b, the heatmap over the checkpoints tends to have a larger error over time but then is significantly smaller at the end of the race, and the larger error is 1674 seconds, corresponding to CP6. In respect to the



(a) Chap. 4, Velocity Alg.: 1588

(b) Chap. 5, RF: 1120

(c) All Races (I_5 - with EL), RF: 1066

(d) All Races (EL absent), RF: 1844

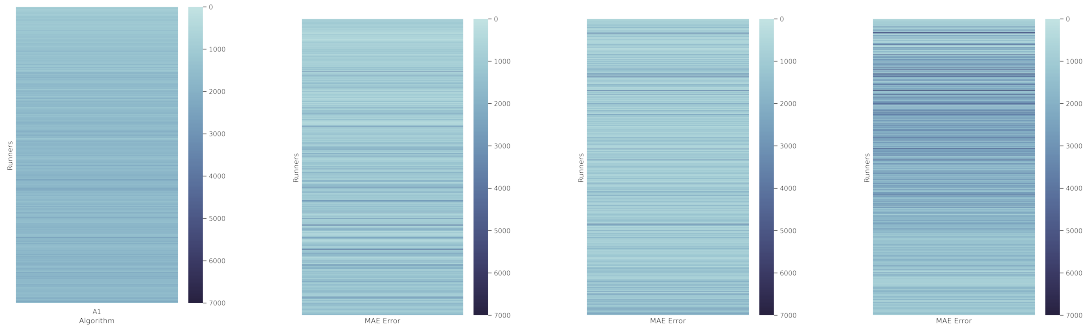
Fig. 24: Checkpoint error values for MIUT 2023 115km in each chapter case study, with MAE mean value

results of this chapter, Figure 24c has the largest error of 1613 seconds in CP6, and minimized the error over the checkpoints, with the last checkpoint (CP11) having an average MAE error of 200 seconds. In Figure 24d, the average MAE value is of 1844 seconds, and the maximum error is obtained. The largest error result in the checkpoints is CP9 with 2971 seconds of error on average. The checkpoints CP4 to CP10 hold errors above 2000 seconds, until the last checkpoint (CP11).

In respect to the MAE mean values observed in Figure 24, we can see that Figures 24b and 24c have a 29.47% and 32.87% improvement over Figure 24a's MAE mean value, respectively. In comparison with Figure 24d, the average MAE error value is 13.88% larger than the average error in Figure 24a.

We can observe that there is an often recurring trend of the checkpoints at the middle of the race having the largest amounts of error which was previously discussed in Chapter 4. The different patterns seen in Figure 24 can suggest the creation of adaptive methods for deciding which training data to use for prediction depending on the characteristics of a checkpoint.

As another point of reference to the discussion of the results of this case study, Figure 25 illustrates a previously employed visualization in Chapter 4 that displays the error values in the average of each runner in a race within a heatmap.



(a) Chap. 4, Velocity Alg.: 1588

(b) Chap. 5, RF: 1120

(c) All Races (I_5 end. lvl.), RF: 1066

(d) All Races (end. lvl. absent), RF: 1844

Fig. 25: Runner error values for MIUT 2023 115km

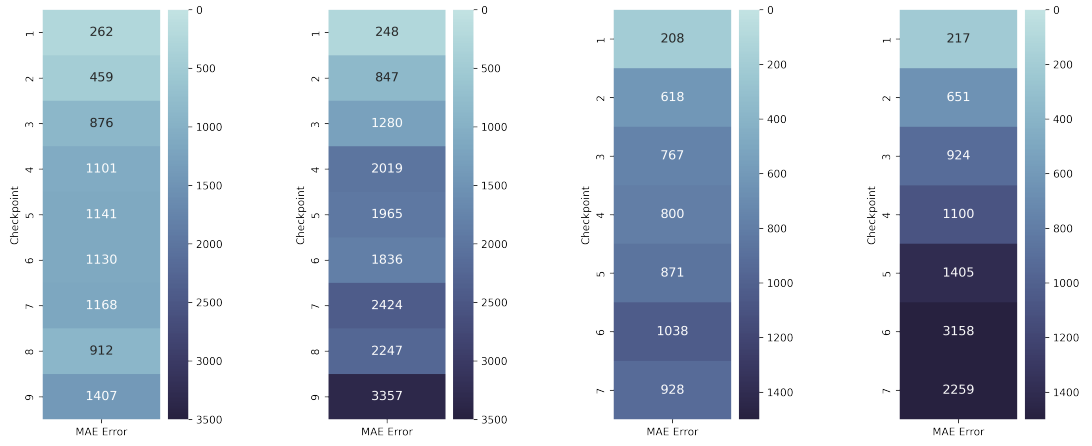
In observation, Figure 25a displays darker tones of colour throughout the heatmap, with noticeable darker lines in the centre of the heatmap. Figure 25b has a lighter heatmap but with more evident darker lines indicating a larger amount of error. In reflection to Figure 25c,

The next subsection will approach the races of MIUT 2023 of 85, 60, 42, and 16km regarding the visualization of average MAE errors associated with the checkpoints, and runners.

6.6.2 All Distance Analysis

In this subsection, we employ the visualization as in the previous subsection, in respect to the multiple distance races of MIUT 2023. The minimum and maximum MAE error results will be discussed throughout this subsection. The results that are not mentioned in this subsection are presented in Appendix B, Section B.1, B.2, B.3 and B.4.

Figure 26 presents four heatmaps related to the average MAE checkpoint values of the MIUT 2023 85km and 60km races.



(a) MIUT Races (end. lvl. absent)
RF: 940 - 85km

(b) All Races (I_4 end. lvl.)
RF: 1803 - 85km

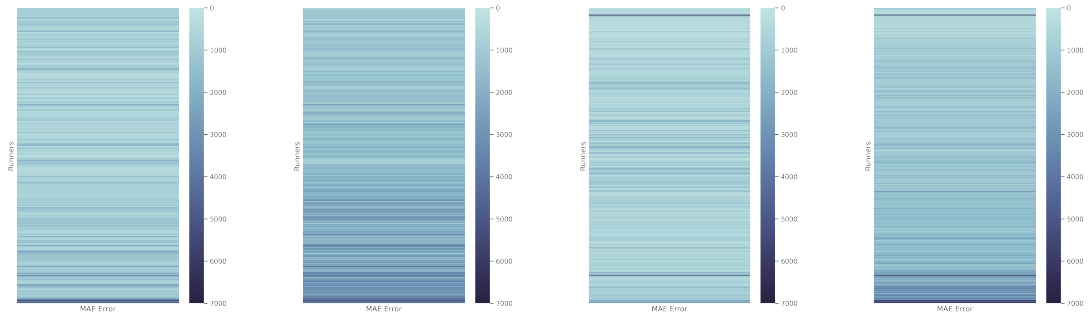
(c) All Races (end. lvl. absent)
RF: 747 - 60km

(d) MIUT Races (end. lvl. absent)
RF: 1388 - 60km

Fig. 26: Checkpoint error values for MIUT 2023 85/60km

In Figure 26a, the heatmap presents errors between 1140 to 1175 seconds in checkpoint CP4 to CP7, with CP4 and CP5 corresponding to the checkpoints at highest altitudes of the race, 1740m and 1548m, respectively. Despite this, the maximum error in the heatmap is 1389 seconds, corresponding to the last checkpoint of the race. On average, the MAE error is 940 seconds, while 26b holds an error of 1803 seconds on average, with a similar distribution of the error in the middle of the race and in the last checkpoint of the race. In Figure 26c, the average MAE error in each checkpoint grows in a successive manner, with the last checkpoint, CP7, holding the largest error of 1068 seconds. In terms of its average MAE error of the race, it has an error of 747 seconds. On the side of the case with the largest average MAE error of the race, Figure 26d also presents the average MAE error in each corresponding checkpoint with larger errors in each successive checkpoint, although both Figures hold the trend of decreasing the average MAE error in CP6 after CP5 but gaining more error in the last checkpoint.

Figure 27 presents the average MAE errors for the predicted time passages of each runner in the race MIUT 2023 85/60km.



(a) MIUT Races (end. lvl. absent)
RF: 940 - 85km

(b) All Races (I_4 end. lvl.)
RF: 1803 - 85km

(c) All Races (end. lvl. absent)
RF: 747 - 60km

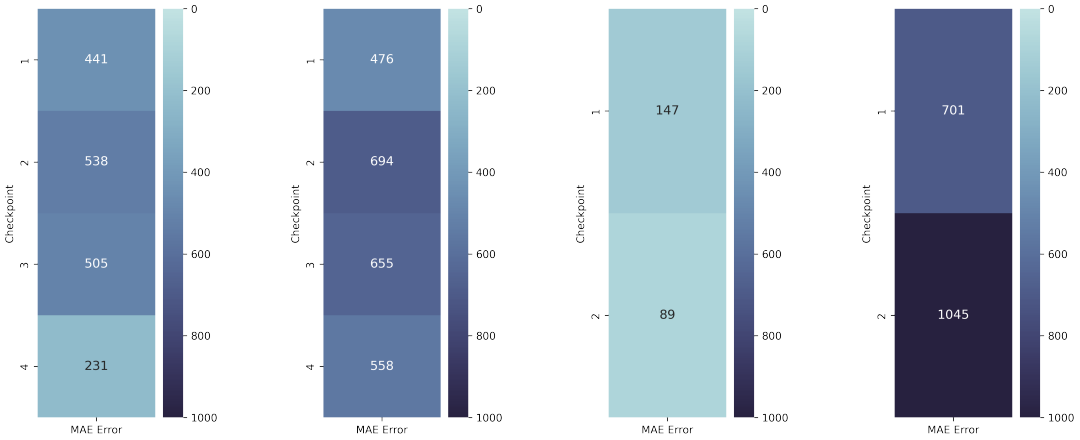
(d) MIUT Races (end. lvl. absent)
RF: 1388 - 60km

Fig. 27: Runner error values for MIUT 2023 85/60km

In both cases, Figures 27a and 27b have larger amounts of error towards the runners that were placed at last positions of the end the race, implying that there's a higher occurrence of error in runners that require more time to finish the race. Figures 27c and 27d continue the trend of displaying runners with larger errors that were placed at the last positions of the race.

Figure 28 presents the average MAE errors for the predicted time passages in each checkpoint of the race MIUT 2023 42/16km.

In Figures 28a and 28b, the error in the checkpoints scales up to checkpoint CP2, and decreases in each successive checkpoint. The average MAE error of Figure 28a is of 429 seconds, 169 seconds less than the error of 596 obtained in Figure 28b. The small difference between the minimum and maximum error obtained out of the race MIUT 2023 42km is related to its shorter race distance, which is the minimum distance of a race to be considered an ultramarathon. Figure 28c is the case where the MAE errors are its minimum in this chapter, given that this is the shortest race of MIUT 2023 and is not an ultramarathon. In comparison to Figure 28d, it holds a global MAE error of 873 seconds, 755 seconds more than the global reported error of 28c, which may imply that the training data related to all races of the endurance level I_0 have a significant number of



(a) MIUT Races (end. lvl. absent)
RF: 429 - 42km

(b) All Races (end. lvl. absent)
RF: 596 - 42km

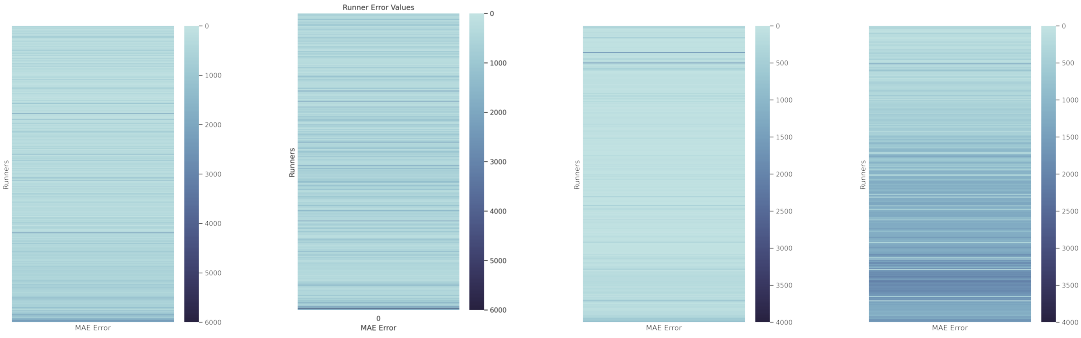
(c) All Races (end. lvl. absent)
RF: 118 - 16km

(d) All Races (I_0 end. lvl.)
RF: 873 - 16km

Fig. 28: Checkpoint error values for MIUT 2023 42/16km

races that do not correspond to the MIUT 2023 16km's race profile, leading to larger errors when considering the race's total distance and total positive elevation.

Figure 29 displays the average MAE errors for the predicted time passages of each runner in the race MIUT 2023 42/16km.



(a) MIUT Races (end. lvl. absent)
RF: 429 - 42km

(b) All Races (end. lvl. absent)
RF: 596 - 42km

(c) All Races (end. lvl. absent)
RF: 118 - 16km

(d) All Races (I_0 end. lvl.)
RF: 873 - 16km

Fig. 29: Runner error values for MIUT 2023 42/16km

Figures 29a and 29b continue to present the trend of the largest errors being represented in the bottom finishers of the race, although Figure 29b, despite holding the largest error of 596 seconds in average, does not present a very different heatmap when compared to Figure 29a. In Figure 29c and 29d, both cases continue to contribute to the trend of the last finishing runners holding the largest MAE errors.

The next section will present the conclusions and remarks of this case study.

6.7 Conclusion

This chapter's objective is to further attempt to improve the obtained results from Chapter 5, but strictly with the model Random Forests as it had improved results in the corresponding chapter. We

intended to also extend the selection of races for this chapter, in an effort to augment the quantity of data, and also continue to use the MIUT 2023 event as the main benchmark for testing data. This resulted in the inclusion of more MIUT races, from editions 2008 to 2023 with distances between 16km to 115km. Given that the MIUT races can be varied, we decided to use the ITRA Points system to classify the race's endurance level, in order to better understand the characteristics of the races. This allowed us to have training sets for each endurance level I_n, from level I_0 (easiest) to level I_6 (hardest). In this way, we have the races' time passage data embedded in sets that describe their endurance level in common.

The classification of these races resulted in discarding I_1 and I_6 as these endurance levels did not have a race from MIUT 2023 classified as either I_1 or I_6, therefore it was not possible to use such endurance levels as testing sets. Ideally, one would have seven races, but at the moment the MIUT 2023 event has five races, and these races serve as five testing sets to be used with the Random Forest model as an effort to minimize the errors calculated throughout the case studies in this dissertation.

In Sections 6.2 and 6.3, we composed five training sets, each with its set of races that were classified by endurance level. The classification of the race's endurance levels facilitates the task of joining the race's time passage data. In both sections, the error results attained from the model do not give us an immediate answer as to which approach attains the least error in each MIUT 2023 race that was used as a testing set according to its endurance level. In Section 6.2, it was reported that the training sets and testing sets corresponding to endurance levels I_4, I_2, and I_0 netted smaller average errors in comparison to the results obtained in Section 6.3. The latter attained less error in the endurance levels I_5 and I_3. Given that this brief study did not yield a definite answer as to which approach can return minimal errors, we attempted the same method presented in Section 6.2 and 6.3, but with the absence of the endurance level classification of races, in an effort to understand if the error results can be further reduced.

In Sections 6.4 and 6.5, we removed the endurance level classification of the races and instead used only one training set in each Section. In the case of Section 6.4, the least MAE errors were obtained in the races of 115km, 85km, and 42km, when compared to the MAE errors attained in Section 6.5, where it managed to return the least MAE errors in the 60km and 16km races. Similarly to Sections 6.2 and 6.3, this does not indicate a stable solution that can net us the least MAE errors in all of the five tested races. As an attempt to investigate this at an atomic level, we employed a visualization technique that was presented previously in Chapter 4, as it allowed us to understand how the MAE errors are distributed in average terms across each checkpoint and also each runner in the races.

In Subsection 6.6.1, we employed the same visualization technique with heatmaps with respect to the 115km race. The MAE errors displayed are in connection with the three case studies conducted, as this dissertation's interest was more concerned with the longer race of MIUT 2023, considering that the 115km race has more influence and attention from the public. It was found that the use of the Random Forest model netted less MAE error when compared to the method employed in the first case study, and such a method did not resource the use of machine learning techniques. The minimal error that was attained in the first case study was 1588 seconds, while the second case study returned 1120 seconds, which is a 29.47% improvement. In the third case study, the minimal MAE error value attained from the four approaches is 1066 seconds, with respect to Section 6.3 and endurance level I_5. This netted a 4.82% improvement over the second case study,

and we delved further into other races of different distances of the MIUT event to verify if we could obtain smaller errors than the obtained results in the first case study.

The presented heatmaps with respect to the checkpoint averaged MAE error value always tended to present larger MAE errors in the middle of the race, hinting that the occurrence of larger MAE errors can arise in race segments where the distance and elevation can vary significantly, resulting in a non-uniform pattern of race profiles, which in turn can make the race challenging on a runner's physical endurance, leading to predictions with larger errors. The data source of the checkpoint's topographical details may also be inaccurate, as this can be another source of bias that can influence the errors of the time passage predictions.

In regards to the heatmaps related to the runner's MAE error values, the heatmaps tend to indicate that, on average, predicting runners that are placed at the top finishing ranks have the least MAE error assigned to them, where in the case of runners that finish towards the last ranks, the MAE errors attained are larger, hinting that runners that do not perform better than the runners placed at the top of the ranks can have unexpected performance issues during the race, leading to predictions with larger magnitudes of error.

In Subsection 6.6.2, we presented the analogous visualization approach to the other four races of MIUT 2023, but with respect to this case study only, as these four races were not subject to the same visualization analysis in previous case studies. The minimal MAE error attained in each of these four races is associated with the results presented in Section 6.4 and 6.5, although the first-mentioned section had minimal MAE error results in the 85km and 42km races, while the latter section netted minimal MAE error results in the 60km and 16km races. In regards to the MAE errors of the checkpoints in the 85km race, there tend to be larger errors in the checkpoints with more varied checkpoint segments in terms of the distance and elevation required to traverse these checkpoint segments. In a similar shape, we can also observe that the heatmaps with respect to the runner's MAE error values also present the trend where the prediction of a runner's passage times tends to have less error in cases of runners placed at the top ranks after the finishing line of the race. Even though the 85km race is shorter than the 115km race, it was the race that netted the most MAE error across this case study in Sections 6.2 to 6.5. This could hint at the possibility of the data not being able to generalize or the presence of inaccuracies from the data, as this can further complicate the model to be able to learn from the data.

The 60km race presents a different trend in regards to the visualization of the MAE errors in each checkpoint, as we observed that the MAE error is larger over the checkpoints until the last checkpoint of the race. This however could be caused by the fact that the 60km race only came into existence starting from the MIUT 2021 race edition, leading to insufficient training data associated with this race. The MAE errors with respect to the runners do present a similar trend as mentioned in this conclusion, the first-ranked runners tend to have less MAE error when compared against the last-ranked runners.

The 42km race presents the heatmaps regarding the checkpoint MAE errors with the larger MAE errors condensed in the middle of the race, and the heatmaps regarding the runner's MAE error values present the trend of attaining less MAE error in the first-ranked runners. As a highlight, this race's MAE results in this case study had the least variation out of all races, considering that the minimal result was of 429 seconds, and the maximal of 596 seconds. The 16km race displays a wider minimal and maximal difference in MAE errors, with the minimal MAE error attained at

118 seconds and the maximal MAE error of 873 seconds. The latter may have such a larger error due to the composition of the training set in Section 6.3 in respect to the endurance level I_0, as it has a considerable number of 63 races, although the majority of the races in that training set are not related to the MIUT 16km races, leading to harder generalization of the MIUT 2023 16km race that was used as testing set to predict the runner's passage times. In terms of the heatmaps related to the runner's MAE errors, we observed the trend of first-ranking runners returning less MAE error, when compared to last-ranking runners.

The feature importances presented throughout this case study lead to similar conclusions of what features the model uses to generalize the data when attempting to predict the time passages of the runners in the testing set. In all testing sets, the feature `accumul_distance` had the most importance out of all features, even in situations where the respective feature had at least 0.8 of importance attributed to it by the model. This implies that the feature `accumul_distance` is significant to the model in regards to understanding what features have the most strength to guide the model's decision-making process when testing the MIUT 2023 races.

This case study presented the prediction of the arrival of runners in four different ways: in Section 6.2 we employed the prediction of the arrival of runners through the classification of the endurance level of the MIUT races in order to have separate training sets wherein each of the five training sets are used as training data to predict the corresponding endurance level of the testing set. This was analogous in Section 6.3 but with the addition of all races available from the pertaining organization that holds the data source. The results from these two sections conducted us to further investigate the prediction of arrival instead without any consideration for the endurance levels grouping of the races. This was employed in Sections 6.4 and 6.5, the first with MIUT races only and the latter with all of the available races, resulting in just one training set in each Section to be used as training data for each of the five testing sets. The attained results did not indicate which approach had the least MAE error in all of the five testing sets, which led us to inspect the error results at the level of the checkpoints and runners to try and further understand what other insights can be found.

The visualization of the heatmaps with respect to the checkpoints and runners does aid us in understanding better which areas of the races tend to have the most MAE error but did not yield a conclusive view on which approach is more favourable in regards to netting the least MAE error in all of the five testing sets. The fact that this case study did not yield a certain choice on a certain approach leads us to suggest a real-world implementation of an adaptive algorithm that can choose the approaches that return the least MAE error, therefore avoiding relying on one singular approach.

The next chapter will present conclusions and proposals for future work related to this work.

7 Conclusion

The primary focus of this dissertation is on predictive analysis within trail running races, particularly in predicting a runner's passage time to the next checkpoint. This can assist race organizers in informed decision-making regarding the safety of the runners in the race, in an effort to guarantee the safety and well-being of runners. These predictions can also be a source of information to improve the opening and closing times of aid stations, the transportation scheduling of runners who could not finish the race, and to enable personal assistants and spectators to be more aware of the probable arrival time of a runner in the next checkpoint.

In Chapter 2, we conducted a search regarding machine learning techniques, in respect to supervised, unsupervised, and reinforcement learning techniques. Supervised learning was often presented as it was more closely connected to the analysis of labelled data, which is the case of our collected data. We presented simple models as the Linear Regression, up to more advanced techniques like the Random Forest model, in order to find inspiration as to what models could be chosen for the development of our case studies. We had initially performed a search to verify if there exists literature related to predicative analysis and trail running, but this search yielded no results. Instead, we conducted a literature search focused on the topic of trail running to understand what relevant topics are researched within trail running. The first 100 search results were collected and were organized into categories and classes to have a general overview of what is studied across trail running. Here, we found literature related to health (61), sports (21), social sciences (16) and 2 articles that were discarded due to being unrelated to trail running. Out of these 100 results, only 1 was found to be closely related to our research questions.

In a second literature search, we had searched for both trail running and machine learning, which yielded 33 search results, but alas the same 1 result we found related to our works was found once again and the other 32 search results were not related to trail running, as in most cases it was often related to marathons. Regardless, we mentioned some of the works related to marathons as inspiration for this dissertation.

The first literature search demonstrated that there is an interest in researching topics associated to health, sports, and social sciences. On the side of trail running and machine learning, the second literature search demonstrates that the use of machine learning techniques in the context of trail running are limited, although the use of machine learning is not a novelty in marathon events.

In Chapter 3, we presented an initial exploratory data analysis of the collected data that was acquired throughout the trail running races, in respect to the specification of the checkpoints, the time passages of the runners in the races and the demographical data of the runners. Post-processing steps were also presented as a means to filter out data that we do not have intention on analyzing, such as runners that did not finish the race or have incomplete time passages of a race. Data transformation was also involved as we had to modify string data types over to the integer data type in order to avoid situations where a model cannot accept string-based input. This chapter established the groundwork for the development of the case studies as we could observe that due to the data types of our data, this would imply a regression-based problem, and this contributes to the choice of models.

In the first case study, outlined in Chapter 4, we proposed three algorithms (A_1, A_2, and A_3), with the goal of predicting runner passage times. These three algorithms are based on the runner's average velocity, with variation in regard to the checkpoint data that each algorithm

holds. Algorithm A_1 holds the average velocities of all checkpoints in a race, while A_2 only holds the average velocity of the previous traversed checkpoint and A_3 is in the case of the previous two traversed checkpoints.

This case study approached five metrics: MSE, RMSE, MAE, R^2, and MAPE. MSE, R^2, and MAPE were found to be hard to interpret and compare with each other in order to draw conclusions about the errors of the predicted values. We found that we could use RMSE and MAE for evaluation and comparison to understand which metric would be most meaningful to us, and we inferred that the MAE metric would be more ideal as RMSE penalizes outliers with larger errors when compared with MAE. We employed the three algorithms with respect to the 115km races of MIUT 2016 to 2022 and used the MAE metric for the evaluation of the errors. The average error value for each algorithm was calculated for comparison. The algorithm A_1 had the least error, of 1501 seconds, while A_2 yielded 1807 seconds, and A_3 yielded 1724 we found that Algorithm A_1 outperformed the others in relation to MAE. Multiple distance races (3), of total distances 85km to 16km were also used in this case study and it was observed that A_1 had favorable results in the 85km and 42km races. Presenting the MAE error values did not feel sufficient for the case study, so we included visualization techniques of the average errors attained in each checkpoint and as well for each runner, in order to understand how the error values are dispersed. We discovered that in checkpoint segments with significant distance and elevation change, these were more prone to larger errors, but in the later checkpoints of the race, the error tended to decrease.

Chapter 5 approached the same problematic with the 115km races of MIUT 2016 to 2023, but with the use of machine learning techniques, more so the supervised learning methods, as our data is labelled. This is in common with the Trail Running Assessment Performance [26], as the authors employed supervised learning methods in their works. LASSO and Random Forests were employed in this case study, in order to predict runner passage times at their next checkpoint. This case study yielded significant improvements, with Random Forests exhibiting notably lower MAE values, with the achieved minimum of 1134 seconds, while LASSO achieved 5070 seconds and the A_1 algorithm's minimum was 1501 seconds from the first case study. The importance of feature selection and consideration of checkpoint-related features emerged as key factors in achieving accurate predictions as the data analyzed was found to have inconsistencies, in particular to the absence of historical data in relation to runners that have competed in previous races, in an effort to have a historical view of the runner's performance times in previous races that they have participated.

The last case study presented in Chapter 6 approached the Random Forest model to employ the training and testing for a larger number of races from the MIUT event between 2008 to 2023 and other races that are not related to the MIUT event. Organizing an expanded number of races was the initial challenge of this case study, and in Section 6.1 we described the ITRA Points system to be adequate as it classifies the endurance level of trail running races according to the race profile. In this way, we managed to organize the races into training and testing sets according to their endurance level classification, as it allowed us to have less heterogeneous sets of races with similar race characteristics. This adjustment is experimented with the first two approaches of this case study: exclusively with MIUT races and with MIUT/non-MIUT races, in order to verify if the errors from previous case studies could be improved with the endurance level classification applied.

Section 6.2 is concerned with only MIUT races, and we obtained the least error in the 85, 42 and 16km races with errors of 1709, 450 and 301 seconds, respectively.

Section 6.3 includes both MIUT races and races outside of the MIUT event. The smallest error was obtained in the 115 and 60km races, with errors of 1070 and 825 seconds, respectively. The error result of 1070 seconds from the 115km race is a 5.81% improvement over the minimum error associated with the 115km race in Chapter 5. The 16km race had a significantly larger error of 861 seconds, which may be caused by the model unable to generalize 63 races into the 16km race as it's a possibility that many of the 63 races do not occur under the exact same race course as the 16km race of MIUT 2023. The error results indicated improved results over the error results obtained in the previous case studies. As a means to obtain more experiments, we decided to conduct the same approaches in Section 6.4 and 6.5, but with the absence of the race endurance level classification.

With the absence of endurance level classification, the least obtained errors in Section 6.4 were obtained in the 115, 85 and 42km races with errors of 1260, 939 and 428 seconds, respectively. This yielded very large errors with respect to the 60km race, with an error of 1390 seconds. With respect to the 85km race, the error of 939 seconds is a 38.26% improvement over the error obtained in Chapter 4 from algorithm A_1.

In Section 6.5, the least attained error values were with respect to the 60 and 16km races, with error values of 748 and 122 seconds, respectively. With respect to the 115km race, with an error of 1842 seconds. This yielded a similar outlook to the previous sections, where we could find reduced errors in certain races of certain sections of this case study, but there was not an implicit direction on which of the approaches had the smallest error in the five tested races. This can suggest the development of an adaptive decision-making algorithm that produces different predictions of races but chooses the prediction with the least error as the one chosen for presentation.

In regards to the feature importance presented in Sections 6.2 to 6.5, a general trend that was observed is that the feature `accumul_distance` yielded the most importance in the tested races. This indicates the model's reliance on features that describe the aspects of a checkpoint in the trail running races. There should be further feature engineering to understand what other aspects of the collected data can surge new features that can help the model's capability to improve the obtained errors in this dissertation. However, it is also necessary to recognize the need for more accurate data in respect of data regarding the runners that compete in the races and the information associated with checkpoints, in an effort to obtain insights with less error.

The visualization techniques presented in Sections 6.6.1 and 6.6.2 were employed mainly to the checkpoints and competing runners of each race in order to have an atomic view of the error values. This allowed us to confirm some trends: we can better understand where the errors tend to be most condensed, in the case of checkpoints, we observed that the error tends to be the highest for checkpoints in the middle of the race, more often than not in checkpoints where there is a significant change in distance and elevation. This trend is easier to visualize with long-distance races, as these are composed of more checkpoints, given their total distances.

Regarding the visualizations with respect to the competing runners, we can generalize that the last-ranking runners tend to have a larger error, given that the last-ranking runners will require more time to finish the race and it may be a possibility that there could be unexpected moments where a runner is underperforming. Contrasting with the first-ranking runners, these have smaller errors, so ideally if a runner maintains stable performance over a race, it's likely that the model can produce less error. These techniques can help analyze error results in a different way to understand

how the model distributes errors across checkpoints and competing runners, providing a different interpretation of the results.

In reflection on the development throughout this dissertation, in Chapter 3 it would be pertinent to complement this work with the platform(s) that can manage different case studies experiments in order to have an approach of continuous experimentation to understand what approach can yield the most favourable error results. This can contribute towards an implementation of real-time prediction of the runner's time passages during an event.

In Chapter 4, the case study can be used as a base benchmark for other experiments such as attempting to predict a runner's time passage on the next one, two, three or more checkpoints ahead. Other variables of interest can be included and experimented with to find out if the error results can be reduced.

In Chapter 5, the presented features in the case study should be studied in order to understand if it is possible to optimize such features, this can mean both adding new features or removing features that are deemed to be noise within the model. Access to historical data on runners' time passages and the use of lagged features, which are clones of checkpoint-type features from past checkpoints, can improve the model's awareness of the context of each runner competing in races. In Chapter 6, given the number of approaches presented and the conclusion of the chapter, it is suggested to attempt to attain more qualitative data in order to avoid absent data. The collected data from the trail running races should have more detail, in specific less missing information regarding the checkpoint information of each race, as it was the case that over 50 races had to be removed from the data to be analyzed due to lack of information. Trail running races outside of MIUT's scope are more prone to lack the necessary data to input in the model for prediction of the runner's time passages.

As we conclude this dissertation, it is essential to recognize that while substantial progress has been made in enhancing predictive accuracy, further research options remain open. In the scheme of open questions, one should explore the incorporation of additional features such as terrain conditions, weather variables, and runner-specific attributes to enhance predictive models further. The existence of data that can contextualize a runner's past performances can provide more accurate results of predictions of the runner's passage times, along with a standardized data structure of the checkpoints information of all races within the data source, in an effort to decrease the dimensionality of checkpoint-type features. These efforts that have been documented in this project can further improve the logistics of a trail running event and its enthusiasts of trail running competition.

7.1 Future work

This section will present suggestions and limitations that have occurred throughout this dissertation.

To ensure that all races are included in the analysis despite missing data, it is crucial to have complete information regarding the elevation changes across checkpoints in different races. This can be accomplished by supplementing the existing information with the assistance of race organizers or by approximating and imputing new data. In Chapter 6, as a result of the existence of races that do not have the complete information on the checkpoint elevation changes, 102 races had to be omitted from a collection of 275 races. Under the possibility of imputing such missing data,

one could raise the pool of races to be used for runner passage prediction. As a suggestion, GPS information of the race track could be used to gather the new data. It is common practice for race organizers to provide the race track in a digital file to the runners. With respect to the data regarding the runners themselves, there should be data that can identify runners that have participated in previous races, therefore resulting in data with more context about a trail runner's race history, which in turn may affect the accuracy of the runner-type features associated to each runner.

The presented features in Chapter 5 and Chapter 6 may be extended to accommodate for lagged features, a technique seen in TRAP [26] where the authors include checkpoint-type features of the previous 1 and 2 checkpoints. This can allow for models to carry more context of the current checkpoint and previous checkpoints that the runner has traversed. The feature importance of such lagged features is also relevant to analyze how much they can contribute to the model's ability to predict the runner's time passages.

Predicting a runner's time passage more than one checkpoint ahead can provide valuable information about their whereabouts and aid in estimating their arrival time at a checkpoint. This is particularly useful for the first and last runners, as it helps determine how long an aid station will be in operation at a checkpoint. Although not directly related to predicting a runner's time passage, one can manipulate the results of each race to predict their performance in a specific race. An adaptive algorithm can be developed to continuously learn from MAE error results of different predictions of different races, in an effort to better select which training and testing sets should be used in the models like the Random Forests to predict the runner's passage times.

Supervised machine learning models should be further investigated to understand if there exist models that can improve the error results of the predictions. It may be helpful to reference models commonly used in the context of road running.

A platform for predicting runners' passage times can be implemented by continuously collecting time passage data during a race and applying machine learning techniques like the Random Forest in order to provide real-time predictions of runners' times in a trail running race.

References

[1] R. Bolt. Skyrunning in the USA. ATRA. [Online]. Available: https://trailrunner.com/trail-news/skyrunning-in-the-usa/

[2] "MIUT 115," https://www.miutmadeira.com/en/race/115-course.

[3] S. Dridi, "Supervised Learning - A Systematic Literature Review," Apr. 2022.

[4] H. Tatsat, S. Puri, and B. Lookabaugh, *Machine Learning and Data Science Blueprints for Finance*. O'Reilly Media, 2020.

[5] S. Badillo, B. Banfai, F. Birzele, I. I. Davydov, L. Hutchinson, T. Kam-Thong, J. Siebourg-Polster, B. Steiert, and J. D. Zhang, "An Introduction to Machine Learning," vol. 107, no. 4, pp. 871–885.

[6] ITRA History. [Online]. Available: https://itra.run/About/History

[7] ITRA Discover Trail Running. [Online]. Available: https://itra.run/About/DiscoverTrailRunning

[8] R. Pinheiro, "Análise Dados de Trail Running," Bachelor's Project, University of Madeira, Jul. 2019.

[9] A. Chase and N. Hobbs, *Ultimate Guide to Trail Running: Everything You Need To Know About Equipment * Finding Trails * Nutrition * Hill Strategy * Racing * Avoiding Injury * Training * Weather * Safety*, 2nd ed. Falcon Guides.

[10] I. El Naqa and M. J. Murphy, "What Is Machine Learning?" in *Machine Learning in Radiation Oncology: Theory and Applications*, I. El Naqa, R. Li, and M. J. Murphy, Eds. Cham: Springer International Publishing, 2015, pp. 3–11.

[11] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects."

[12] T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke, "Special issue on feature engineering editorial," *Machine Learning*, Aug. 2021.

[13] K. Afifah, I. N. Yulita, and I. Sarathan, "Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier," in *2021 International Conference on Artificial Intelligence and Big Data Analytics*, Oct. 2021, pp. 22–27.

[14] "K-means Clustering: Centroid," https://www.programsbuzz.com/article/k-means-clustering-centroid.

[15] R. S. Sutton and A. G. Barto, *Reinforcement Learning, Second Edition: An Introduction*. MIT Press.

[16] S. Kim and H. Kim, "A new metric of absolute percentage error for intermittent demand forecasts," *International Journal of Forecasting*, vol. 32, no. 3, pp. 669–679, Jul. 2016.

[17] D. J. Casa, R. L. Stearns, R. M. Lopez, M. S. Ganio, B. P. McDermott, S. Walker Yeargin, L. M. Yamamoto, S. M. Mazerolle, M. W. Roti, and L. E. Armstrong, "Influence of hydration

on physiological function and performance during trail running in the heat," vol. 45, no. 2, pp. 147–156.

[18] R. M. Lopez, D. J. Casa, K. A. Jensen, J. K. DeMartini, K. D. Pagnotta, R. C. Ruiz, M. W. Roti, R. L. Stearns, L. E. Armstrong, and C. M. Maresh, "Examining the influence of hydration status on physiological responses and running speed during trail running in the heat with controlled exercise intensity," *The Journal of Strength & Conditioning Research*, vol. 25, no. 11, pp. 2944–2954, 2011.

[19] R. L. Stearns, D. J. Casa, R. M. Lopez, B. P. McDermott, M. S. Ganio, N. R. Decher, I. C. Scruggs, A. E. West, L. E. Armstrong, and C. M. Maresh, "Influence of hydration status on pacing during trail running in the heat," *The Journal of Strength & Conditioning Research*, vol. 23, no. 9, pp. 2533–2541, 2009.

[20] C. S. Easthope, C. Hausswirth, J. Louis, R. Lepers, F. Vercruyssen, and J. Brisswalter, "Effects of a trail running competition on muscular performance and efficiency in well-trained young and master athletes," *European journal of applied physiology*, vol. 110, no. 6, pp. 1107–1116, 2010.

[21] D. Rojas-Valverde, B. Sánchez-Ureña, J. Pino-Ortega, C. Gómez-Carmona, R. Gutiérrez-Vargas, R. Timón, and G. Olcina, "External workload indicators of muscle and kidney mechanical injury in endurance trail running," *International Journal of Environmental Research and Public Health*, vol. 16, no. 20, p. 3909, 2019.

[22] S.-L. Ng, Y.-F. Leung, S.-Y. Cheung, and W. Fang, "Land degradation effects initiated by trail running events in an urban protected area of Hong Kong," *Land Degradation & Development*, vol. 29, no. 3, pp. 422–432, 2018.

[23] D. G. Havlick, E. Billmeyer, T. Huber, B. Vogt, and K. Rodman, "Informal trail creation: Hiking, trail running, and mountain bicycling in shortgrass prairie," *Journal of Sustainable Tourism*, vol. 24, no. 7, pp. 1041–1058, 2016.

[24] F. Vercruyssen, C. Easthope, T. Bernard, C. Hausswirth, F. Bieuzen, M. Gruet, and J. Brisswalter, "The influence of wearing compression stockings on performance indicators and physiological responses following a prolonged trail running exercise," *European Journal of Sport Science*, vol. 14, no. 2, pp. 144–150, 2014.

[25] H. A. Kerhervé, P. Samozino, F. Descombe, M. Pinay, G. Y. Millet, M. Pasqualini, and T. Rupp, "Calf compression sleeves change biomechanics but not performance and physiological responses in trail running," *Frontiers in Physiology*, vol. 8, p. 247, 2017.

[26] R. Fogliato, N. L. Oliveira, and R. Yurko, "TRAP: A predictive framework for the Assessment of Performance in Trail Running," *Journal of Quantitative Analysis in Sports*, vol. 17, no. 2, pp. 129–143, 2021.

[27] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," vol. 58, no. 1, pp. 267–288.

[28] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. Association for Computing Machinery, pp. 785–794.

[29] L. Breiman, "Random Forests," vol. 45, no. 1, pp. 5–32.

[30] J. D. Malley, J. Kruppa, A. Dasgupta, K. G. Malley, and A. Ziegler, "Probability machines: Consistent probability estimation using nonparametric learning machines," *Methods of Information in Medicine*, vol. 51, no. 1, pp. 74–81, 2012.

[31] H. T. El-Kassabi, K. Khalil, and M. A. Serhani, "Deep Learning Approach for Forecasting Athletes' Performance in Sports Tournaments," in *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*, ser. SITA'20. Association for Computing Machinery, pp. 1–6.

[32] L. Medsker and L. C. Jain, *Recurrent Neural Networks: Design and Applications*. CRC Press.

[33] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," vol. 9, no. 8, pp. 1735–1780.

[34] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling.

[35] J. Berndsen, B. Smyth, and A. Lawlor, "Pace my race: Recommendations for marathon running," in *Proceedings of the 13th ACM Conference on Recommender Systems*, ser. RecSys '19. Association for Computing Machinery, pp. 246–250.

[36] J. Atterfors, J. Lamm, and M. Johansson, "Machine Learning of Pacing Patterns for Half Marathon," 2022.

[37] M. A. Rayhan and K. M. Lhaksmana, "Racing Bib Number Recognition Method using Deep Learning," vol. 4, no. 3, pp. 815–824.

[38] J. Redmon and A. Farhadi. YOLOv3: An Incremental Improvement.

[39] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[40] S. K. Bansal and S. Kagemann, "Integrating Big Data: A Semantic Extract-Transform-Load Framework," vol. 48, no. 3, pp. 42–50.

[41] C. Seger, *An Investigation of Categorical Variable Encoding Techniques in Machine Learning: Binary versus One-Hot and Feature Hashing*, 2018.

[42] "ITRA Organizers," https://itra.run/FAQ/Organizers.

Appendix

A Statistical Summary of MIUT 2016-2023 115km features

In Table 35, we present a statistical summary of the features from the MIUT 2016-2023 115km race data.

Table 35: Statistical summary of the features in the dataset

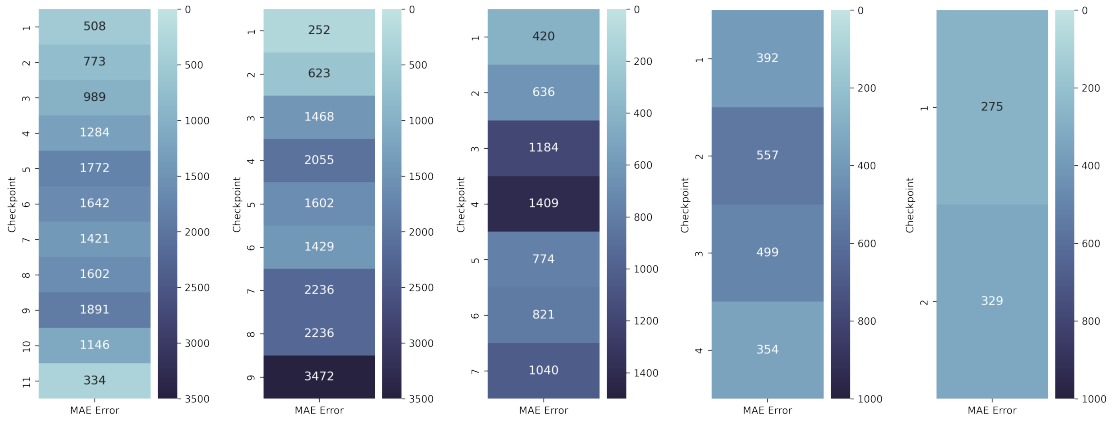
Feature	Distinct	Mean	St. Dev.	Min	25%	50%	75%	Max
distance	45	8286.52	3317.10	3600.00	5300.00	8100.00	10800.00	15300.00
accumul_distance	57	62268.21	30286.36	12400.00	35900.00	67800.00	87500.00	116200.00
altitude	18	870.75	570.53	5.00	312.00	837.00	1496.00	1740.00
plus_elev	55	563.88	541.73	0.00	80.00	415.00	935.00	1515.00
minus_elev	59	578.69	339.66	0.00	305.00	420.00	815.00	1380.00
accumul_plus_elev	66	5087.54	2125.79	1487.00	3030.00	6080.00	6958.00	7854.00
accumul_minus_elev	70	4227.96	2308.33	360.00	2385.00	4275.00	6373.00	8364.00
time_barrier	2	0.77	0.42	0.00	1.00	1.00	1.00	1.00
AP	2	0.40	0.49	0.00	0.00	0.00	1.00	1.00
L	2	0.92	0.28	0.00	1.00	1.00	1.00	1.00
+	2	0.92	0.28	0.00	1.00	1.00	1.00	1.00
S	2	0.87	0.33	0.00	1.00	1.00	1.00	1.00
Méd.	2	0.28	0.45	0.00	0.00	0.00	1.00	1.00
MR	2	0.08	0.27	0.00	0.00	0.00	0.00	1.00
total_elev	52	8163.10	3105.94	7066.00	7090.00	7095.00	7715.00	37294.00
mean_elev	6	7330.11	307.00	7066.00	7090.00	7095.00	7540.00	7854.00
total_dist	44	120791.50	47351.36	101000.00	101000.00	103000.00	114900.00	549900.00
mean_dist	5	107899.82	6695.86	101000.00	101000.00	103000.00	114900.00	116200.00
min_dist	5	107873.79	6666.15	101000.00	101000.00	103000.00	114900.00	116200.00
max_dist	5	108392.88	6740.57	101000.00	101000.00	103000.00	114900.00	116200.00
mean_elev_dist	51	0.07	0.01	0.06	0.06	0.07	0.07	0.07
max_elev_dist	6	0.07	0.01	0.06	0.06	0.07	0.07	0.07
min_elev_dist	6	0.07	0.01	0.06	0.06	0.07	0.07	0.07
n_competitors	7	534.55	119.94	210.00	481.00	583.00	625.00	648.00
n_races	5	1.24	0.65	1.00	1.00	1.00	1.00	5.00
avg_competitors	60	532.56	115.58	210.00	481.00	571.50	625.00	648.00
years_since_first_race	8	0.20	0.81	0.00	0.00	0.00	0.00	7.00
year_of_last_race	7	2019.56	2.40	2016.00	2017.00	2019.00	2022.00	2023.00
months_since_last_race	14	2.01	8.07	0.00	0.00	0.00	0.00	84.00
mean_rank	771	256.27	158.63	1.00	122.00	242.00	379.00	612.00
mean_rank_perc	3116	0.48	0.27	0.00	0.25	0.48	0.71	0.96
max_rank_perc	2971	0.49	0.28	0.00	0.25	0.49	0.72	2.65
min_rank_perc	2956	0.47	0.27	0.00	0.24	0.47	0.70	0.96
rank_perc_vs_elev_dist	3294	0.03	0.02	0.00	0.02	0.03	0.05	0.10
avg_female_percentage	2759	0.02	0.01	0.00	0.02	0.02	0.02	0.09
perc_time_overall	3273	0.62	0.24	0.00	0.45	0.64	0.82	1.00
perc_time_first	3279	0.82	0.32	0.00	0.59	0.85	1.09	1.46
perc_time_last	3279	0.22	0.14	0.00	0.10	0.20	0.31	0.59

B Arrival prediction error visualization of checkpoints and runners

This section presents the average MAE produced for each checkpoint in each race, with the employment of the Random Forest model.

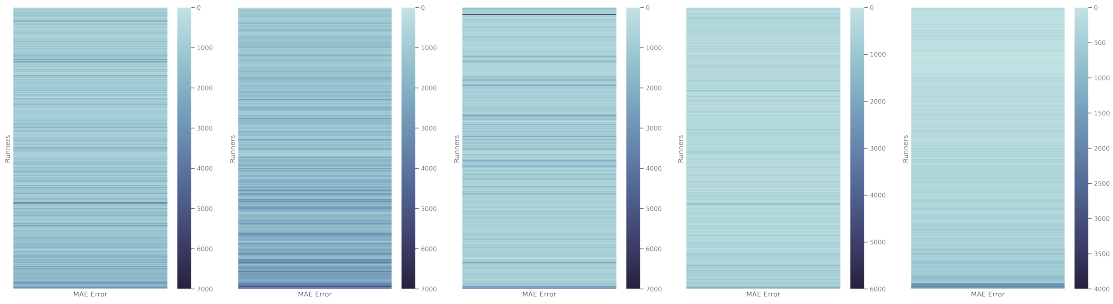
B.1 Prediction of arrival in MIUT races by endurance level

This section represents the MIUT races for each tested race, with their correspondance to an endurance level I_n.



(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 30: Checkpoint error values for the five MIUT races, with MAE mean value

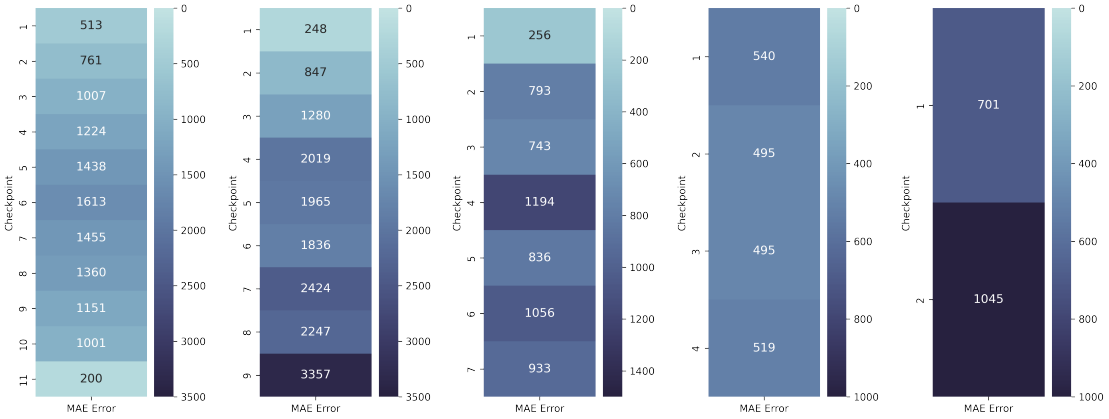


(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 31: Runner error values for the five MIUT races, with MAE mean value

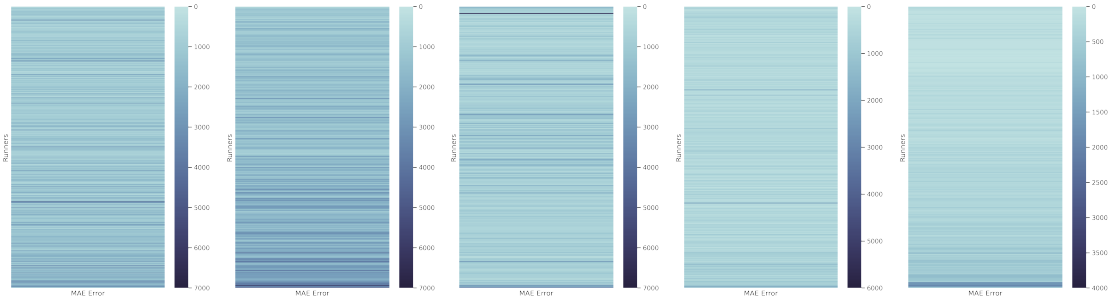
B.2 Prediction of arrival in all races by endurance level

This section represents all races for each tested race, with their correspondance to an endurance level I_n.



(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 32: Checkpoint error values for the five MIUT races, with MAE mean value

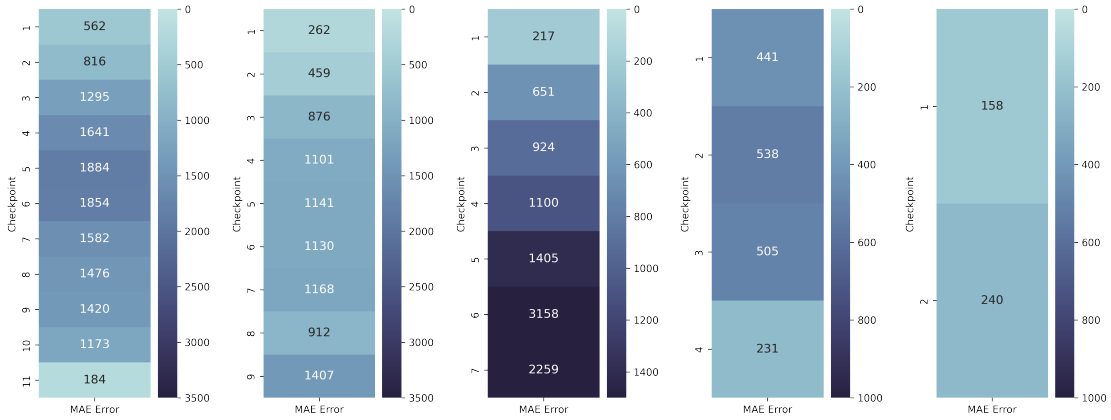


(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 33: Runner error values for the five MIUT races, with MAE mean value

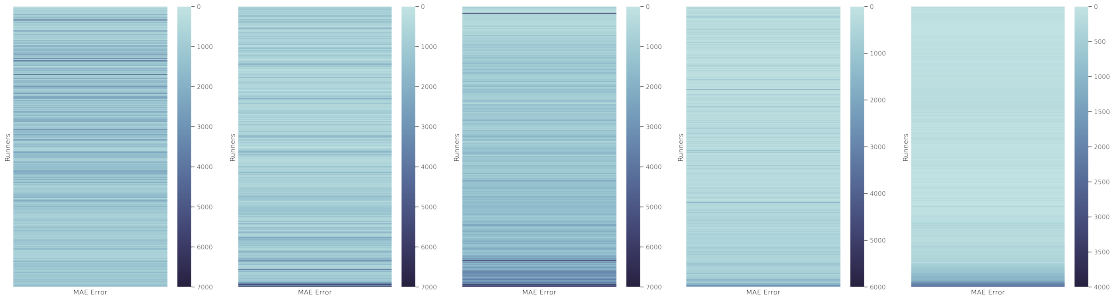
B.3 Arrival prediction in MIUT races (without levels)

This section represents the MIUT races for each tested race, with no endurance level classification.



(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 34: Checkpoint error values for the five MIUT races, with MAE mean value

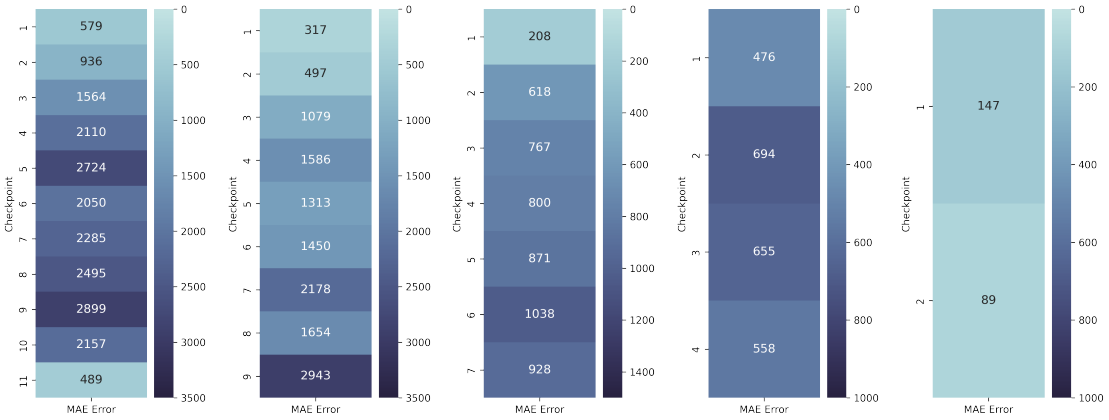


(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 35: Runner error values for the five MIUT races, with MAE mean value

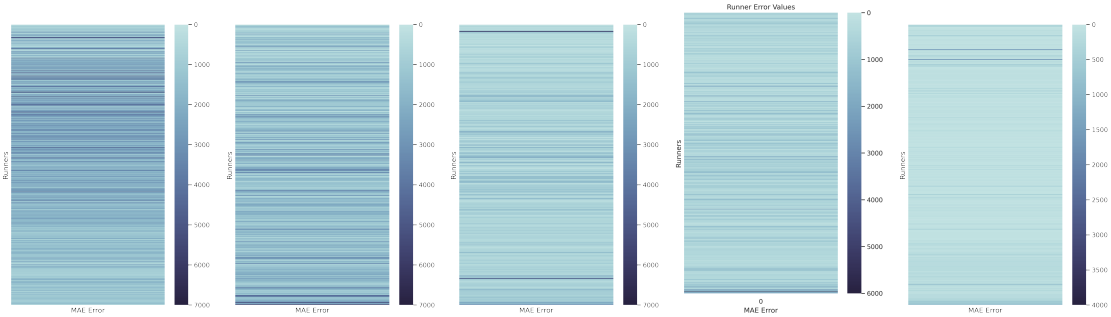
B.4 Arrival prediction in all races (without levels)

This section represents all races for each tested race, with no endurance level classification.



(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 36: Checkpoint error values for the five MIUT races, with MAE mean value



(a) 115km (b) 85km (c) 60km (d) 42km (e) 16km

Fig. 37: Runner error values for the five MIUT races, with MAE mean value