Predicting the next Pogačar: A data analytical approach to detect young professional cycling talents

Bram Janssens[a], Matthias Bogaert[a], Mathijs Maton[a]

[a]Ghent University, Department of Marketing, Innovation and Organisation, Tweekerkenstraat 2, 9000 Ghent, Belgium
Bram.Janssens@UGent.Be, Matthias.Bogaert@UGent.Be (Corresponding author), MathijsMaton@hotmail.com

ABSTRACT

The importance of young athletes in the field of professional cycling has sky-rocketed during the past years. Nevertheless, the early talent identification of these riders largely remains a subjective assessment. Therefore, an analytical system which automatically detects talented riders based on their freely available youth results should be installed. However, such a system cannot be copied directly from related fields, as large distinctions are observed between cycling and other sports. The aim of this paper is to develop such a data analytical system, which leverages the unique features of each race and thereby focusses on feature engineering, data quality, and visualization. To facilitate the deployment of prediction algorithms in situations without complete cases, we propose an adaptation to the k-nearest neighbours imputation algorithm which uses expert knowledge. Overall, our proposed method correlates strongly with eventual rider performance and can aid scouts in targeting young talents. On top of that, we introduce several model interpretation tools to give insight into which current starting professional riders are expected to perform well and why.

Keywords: Sports Analytics, Scouting Analytics, Missing Value Imputation, Predictive Modelling, Interpretable Machine Learning

1. Introduction

Cycling has become a global sport during recent decades (Van Reeth, 2016). The sport attracts up to 25 million live viewers for popular races (Van Reeth, 2019) and budgets skyrocket up to €130 million for an individual race (Andreff, 2016). In comparison to other sports, the margins in cycling are small, with multiple pre-race favorites, and with seven-hour races often ending with differences of only seconds. These small margins are further complicated by the varying race tactics. Two broad types of races can be distinguished: (1) one-day races where the first-arriving participant is also the overall winner, and (2) stage races where participants race over multiple days and the summed-up time over all days (stages) is used to determine the final ranking. A special feature of the latter is that many participants also focus on stage victories or climbers or sprinters classifications, while not competing for the general classification. This leads to several competitors with differing goals, which might profit from the actions of others in achieving their own goals. This highly competitive setting has caused professional teams to search for small competitive gains by deploying various strategies in search of success.

One such strategy is the recruitment of young talented riders, which has become more important during recent years. The most popular cycling race on the planet, the Tour de France, can be seen as an indicative example of this changing trend. During the years 2000-2018, the average age of the overall winner[1] was above 30 years. However, the two most recent winners of the Tour de France, Egan Bernal (2019 winner) and Tadej Pogačar (2020-2021 winner) were respectively aged 22 and 21 years old at the time of their first victories. These figures might seem anecdotal at first, but overall, the sport has seen a rejuvenation, with young riders performing extremely well. For instance, when looking at the ProCyclingStats (PCS) points scored by neopro riders (professional athletes in the first two years of their professional career) there is an increase of around 20% over 10 years. This also resulted in the induced interest of teams in these riders, reflected by a drop in average starting age of a stunning 1.5 years during the period 2018-2020.

Despite the growing importance of young rider recruitment, we observe a limited development of the techniques to detect such young talents. Talent identification has mainly focused on physiological aspects, such as the overall oxygen uptake (Nevill et al., 2005), as predictors of future performance. While this characteristic is suited for detecting whether someone has the ability to become a professional athlete, it is proven to be unreliable in predicting competitive success at the highest levels (Menaspà, Sassi & Impellizzeri, 2010). More recent measures are primarily focused on power output-related measures, which are more reliable (Larson & Maxcy, 2016). Nevertheless, obtaining such measures is a non-trivial task, as this information is not publicly available, and teams need to conduct

[1] The age of the original winner, before any doping-related alterations to results, was used.

costly tests to obtain such information. Doing such tests for a wide range of riders would proof extremely costly and a waste of valuable resources. Hence, teams are forced to put a system in place to cost-effectively pre-select the riders undergoing such tests. Several talks with industry experts revealed that this is still largely done based on a subjective assessment of the young riders, stating the immense importance of mental strength and perseverance. However, the process remains largely subjective and prone to human errors

One possible solution could lay in the development of an analytical tool which automatically detects promising riders based on open-source data, which could then be selected for physical scouting and physiological testing (e.g., power output testing). To the best of our knowledge no such method has been developed in the cycling domain. This while research in other sports fields (e.g., Liu, Schulte & Li, 2018) already experimented with such analytical models, whose suggestions outperform actual team preferences with regard to eventual performance of players. The reason that no such tool has been developed for cycling lays in the uniqueness of the cycling sport and the nature of the data.

The main objective of this study is therefore to design a data analytical system that is capable of accurately predicting future performance of athletes based on youth race performances. The data is scraped from the PCS website (procyclingstats.com), which is a publicly available website containing stats from all cycling races. Since not all riders compete in all races, building an analytical model on the historical results of riders in the individual races of the youth categories is challenging due to the high number of missing values (often over 50%). Therefore, special academic interest is given to the different imputation methods to keep the unique characteristics of each race. For example, a point system were only points are accumulated through races in which a rider participates (e.g., the Official World Golf Ranking) would leave out important details as riders might only thrive in certain types of races. This consideration is of much less importance in other sports, where the sporting environments are much more standardized. To accurately predict race performance we benchmark several prediction algorithms that have achieved superior performance in academic literature. The goal of our system is to come up with a list of potential future top performers and to provide insights into which features are critical to the models. Since interpretability is of major importance in scouting analytics (e.g., Liu, Schulte & Li, 2018), we also put large emphasis on how the results of our black-box prediction algorithms can be visualized and interpreted. Hence, our analytical approach should provide insights to both scouts and fans which results and indicators to keep track off to identify potential future prospects.

The remainder of this study is structured as follows. Section 2 discusses advances made in scouting analytics and cycling analytics, followed by the used methodology in Section 3. Section 4 elaborates on the performance and outcome of the various techniques. Section 5 discusses the implications of these results, while we end with a concluding remark and a critical note in Section 6.

2. Related Work

This section discusses current advances in literature. Section 2.1. describes how the interest in cycling analytics increased in recent years. Nonetheless, this growth in academic interest did not lead to an analytical approach to detect young cycling talents. This while talent identification has seen some interesting developments in a variety of fields. Several interesting developments in talent identification, and how they differ or relate to the cycling field are therefore discussed in Section 2.2.

2.1. Cycling Analytics

In recent years, there has been an emergence of cycling-related data analytical studies. While some studies use analytical approaches to facilitate recreational and commuter cycling (e.g., Kumar, Nguyen, & Teo, 2016), less efforts have been made to use analytics to boost rider and team performances. Initial introductions towards data analytical methods in the field were made by Hilmkil et al. (2018). The authors were able to predict a cyclist's heart rate at various moments in the training ride using a long short-term memory (LSTM) model. The study can be regarded as a proof-of-concept, indicating the feasibility of data analytical approaches in the field of cycling.

This proof-of-concept was quickly followed by a range of studies who focus on practical applications to the cycling community. For instance, Kataoka and Gray (2018) develop a real-time analytical system to predict power performance of professional riders at the Tour de France (deployed on 2017 edition) based on GPS and wind sensor data. This would allow fans to have reliable estimates of the performance of athletes during the race. Another interesting study by De Spiegeleer (2019) predicts the average velocity of a stage, the difference between the average stage velocity and the velocity of a rider and, finally, the head-to-head wins between two riders in a stage, using open data from the PCS website. This popular information-tracking website was also used in other relevant studies. For instance, Kholkine et al. (2020) proofs that it is feasible to build a model which is capable to predict race rankings based on previous race rankings scraped from the PCS website, while Karetnikov (2019) was capable to predict individual rider performance in key mountain stages using a combination of private training data and the open data available on the PCS website. These studies, and the wide usage of the website among fans, has clearly established the PCS website as the go-to source for open cycling data.

Another interesting observation can be made with regard to the used methodology. While several studies (Karetnikov, 2019; Kataoka & Gray, 2018) compare more complicated deep learning architectures with gradient-boosted trees, gradient-boosted trees are consistently ranked as top performing algorithm. While De Spiegeleer (2019) observes ridge regression to outperform gradient boosting in only one out of the three prediction tasks, extreme gradient boosting (XGBoost) is ranked first on two out of three tasks, which further re-enforces the superiority of the algorithm.

While extensive research interest has been paid to the performance prediction of current riders, there has been no interest in the early detection of successful riders. Nevertheless, such talent identification can deliver useful competitive gains when deployed correctly. This is especially true in the current professional cycling context, where riders are turning professional at a very young age and are showing competitive results from the very beginning of their career. In the next section, we will focus more on the theory behind talent identification, and related concepts from other sports.

2.2. Talent Identification

Vaeyens et al. (2008) define talent identification as "the process of recognizing current participants with the potential to excel in a particular sport". It should be distinguished from talent development, which can be regarded as a phase following talent identification, where prospects are "provided the most appropriate learning environment to realize this potential". Our study is situated in the talent identification phase, rather than the talent development phase. This has important implications, as the definition of future performance should not be defined too far into the future as unsatisfactory results in the more distant future may be the result of incorrect talent development rather than false talent identification.

Achieving the highest level in sports is an important goal, not only for the athlete, but also for other instances involved, such as sponsors, governmental bodies, and coaches (Anshel & Lidor, 2012). All these stakeholders wish to allocate their scarce resources as efficient as possible to achieve their respective goals. To do so, teams should allocate their resources towards a number of talented riders who have a high potential. However, in a systematic literature review of Johnston et al. (2018) on talent identification, the authors conclude that little is known about the correct application of talent identification in sports and that further and more diverse research is needed.

Some sports have, however, received more academic interest with regard to analytical scouting than others. In the past the National Hockey League (NHL) has received a lot of attention. This can be attributed to the fact that the NHL is well-suited for the study of decisions and drafting because of four factors: the number of drafting rounds, lack of trading restrictions, hard salary caps, and unique provision of league-wide scouting services (i.e., the NHL even provides a scouting ranking of all youth players) (Tingling, 2016). While developments are made in this area are useful, they cannot be translated directly to cycling. Its unique combination between individual and team sport and the immense impact of the used route on the outcome of a cycling race, makes the sport notoriously hard to compare to other sports (De Spiegeleer, 2019). Nevertheless, the knowledge of related fields is useful to draw some general conclusions from, while we should remain cautious towards in-depth transfer of practices.

One such general conclusion can be made about the accuracy-interpretability trade-off in scouting analytics (Liu, Schulte & Li, 2018). Special attention should be given to interpretability and visualization when dealing with analytical talent identification, since scouts tend to believe their gut feeling is more than analytical models. Hence, a highly interpretable model can be easily justified by scouts and is conversely more easily adopted in the industry (Baesens, 2014). A similar sentiment was detected during our conversations with cycling industry experts. The goal of a successful scouting tool should be to give reliable estimates, while being easily interpretable for experts who have no experience with analytical models. Several suggestions have been made with regard to interpretability. Cohort-based approaches are one of the most popular methods which 'cluster' players into groups of players with comparable profiles and predict future success based on the cohort's success. For example, Weissbock (2015) clusters hockey players according to age, height, and scoring rates. One advantage of the cohort model is that predictions can be explained by referencing to similar players, which many domain experts find intuitive. Another example can be found in baseball where the Player Empirical Comparison and Optimization Test Algorithm (PECOTA) allows to forecast player performance in a highly accurate and interpretable way (Koseler & Stephan, 2017). It is thus advisable to compare prospects with known top performers rather than to simply give a list of potentially interesting prospects. Another interesting insight can be drawn from handball, where player profiles are depicted using spider plots (Blom, 2019). When the dimensions display relatable dimensions, these plots are easy to interpret for sports scouts.

Other outcomes from related domains are less transferable. For instance, consider the three most important NHL scouting features used in Liu, Schulte and Li (2018): Central Scouting Service (CSS) ranking, regular season points total, and plus-minus score. Cycling does not have a scouting ranking, which makes such a variable infeasible. The regular season-points and plus-minus score are available in cycling but due to the uniqueness of cycling including these features is not straightforward. Simply calculating the regular season points total would not consider the fact that riders sometimes act as team leaders and sometimes as helpers ('domestiques'). This would also not incorporate the fact that riders from countries such as Belgium or France have much more access to the popular races than riders from more distant countries, which automatically leads to a higher accumulation of points. This is even further aggravated by the different routes used in each race. While some riders may be quite weak on long and relatively flat stages, they may do quite well on high mountain stages, and vice versa. While the Tour de France is generally regarded as the most important race in the world, we rarely see the winners of that race competing in other high-level races such as Paris-Roubaix. This uniqueness of each race on the calendar should also be reflected when building a model and creating its input features. A natural solution is to take the race results per race, rather than aggregating per season. This translates into a very sparse data matrix with a disproportionate number of missing values, as not all riders participate in all races, hindering the creation of a performant predictive model.

In sum, we observe large distinctions between cycling and other sports, which makes it hard to translate knowledge from other fields into the field. This creates the necessity of the development of a field-specific data analytical methodology for talent identification. When creating such methodology, the unique characteristics of each race need to be represented in the data. This leads to a heavy focus on feature engineering, data quality and interpretation. Therefore, the handling of missing values and the visualization of eventual results is of special interest to this research.

3. Methodology

3.1. Data

The used data were collected from the PCS (procyclingstats.com) website, which keeps track of all youth results. A list of popular youth competitions is created (listed in Appendix Table A1), and for each of these races all the available results are scraped from the period 2005-2020, as almost no youth results were available prior to 2005. These scraped results are used as the basis of the independent variables (see 3.2. Feature Engineering). Given the scarcity of results in earlier years, we decide to only select riders who turned professional in the years 2010-2019. Before 2010, we observe the riders to have too little observed race results (i.e., less than 40 observations), leading to heavy time-based sample bias. Riders who turned professional in 2020 or 2021 are also not selected, as they did not yet have two full years of observed dependent period. Overall, this resulted in a sample of 1,060 athletes. The goal of our model is to predict the performance during a period in the rider's professional career, based on the results he achieved as youth competitor. This implies that, for instance, when modelling a rider turning professional in 2018, all his results up until 2017 will be used as input for the independent variables, while results from 2018 onwards will be used as input for the dependent variable. We choose this set-up as this is the best reflection of how talent identification models would be deployed in real-life situations.

The dependent variable is computed as the PCS points scored in the first two years as a professional athlete. This definition closely follows the regulations of the Union Cycliste Internationale (UCI; international cycling federation), which state that starting professional athletes (defined as competing in one of the top two tier levels) should be awarded contracts of at least two years. By measuring their performance during these two years, we can directly measure the return on investment of the hiring team. This limited time window also filters out potential negative effects of bad talent development. The choice for PCS points rather than the official UCI points is inspired by the fact that this points system has remained stable during the entire period 2010-2020, while the current UCI ranking system only dates back to 2016. The PCS ranking also awards points for riders achieving specific goals such as winning the best climbers' classification in the Tour de France. This is a prestigious result, with multiple competitors each year, yet no points are awarded for winning this classification in the UCI ranking, while the PCS ranking does award points for this and similar achievements. Another interesting feature of the PCS ranking is the fact that it rewards all riders that finish the most prestigious races, while the UCI ranking only awards point to the first sixty riders. It is unlikely that a rider performs a lot of work in the first half of the race and still manages to finish in the top-60. In the UCI system this rider would not receive any point for the, although this rider has performed well for the team as a domestique. Nevertheless, being selected for this race already reflects the fact that the rider in question is of high value to the team. Therefore, we argue that PCS points reward domestiques in a fairer way than the UCI points system does. The adequateness of the ranking also inspired other researchers who quantified professional cycling athlete performance on the PCS ranking (e.g., Miller & Susa, 2018; van Erp, Sanders & Lamberts, 2021).

3.2. Feature Engineering

The independent variables have to be representative of the unique characteristics of each race, thus we look at the performance per race as few races are truly comparable to each other due to distinctiveness in used route as well as level of competition. Therefore, we calculate the best overall position of a rider per race, if they would have participated. This means that for each race listed in Appendix Table A1, we create a variable indicating the best overall result of the observed rider. Riders that participated but abandoned during the race, are awarded the ranking of the last finishing rider plus one. However, the best overall result is an incomplete measure, as there is a significant difference in ending 9th while in the same group as the winner at the finish line, compared to finishing 7th and reaching the finish line two minutes after this winner. Therefore, we also compute the minimal time difference to the winner. Finally, we also add an indicator variable whether someone had ever actually participated to the race.

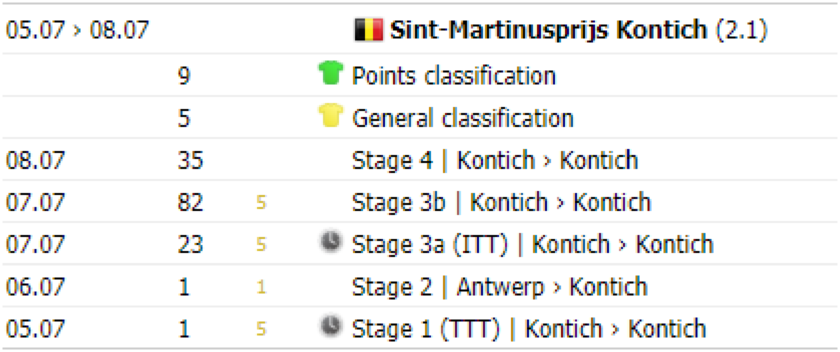
While these features already incorporate many dynamics of the races, they still do not account for the fact that, during stage races, many participants simply aim for stage success, rather than for the overall ranking. Therefore, we also add the number of stage victories per race (across all participations), and the best result one has achieved across all stages.

Several other, more general, features are computed as well, like the number of observed results (i.e., number of race participations), total number of abandons, and the ratio between both (abandon ratio). These variables indicate how present the rider was in the youth circuit as more successful riders are more likely to be selected for most of the races, as well as having some indication of the mental strength of riders. Mental strength was indicated by the domain experts as being a large determinant in how they evaluate talents. Mentally stronger riders are arguably less likely to abandon after misfortune during the race.

Other features make a distinction between the U23 (aged 19-22) and Junior (aged 17-18) categories. Some athletes develop quicker than others as they mature at an earlier age. Therefore, there are many examples of talents that performed very well in very early age categories, while performing relatively weak at later age. This combined with the stronger competition in the U23 circuit, makes the U23 and Junior categories of differing importance. Therefore, we define several features for both the U23 and Junior categories: the number of victories, victory ratio (wins/participations), podium (places 2 and 3) and top 5 (places 4 and 5) ratios. Finally, we also measure the ratio of the number of wins, podium ratio, and top 5 ratio in the U23 category compared to the Junior category as upwards or downwards trends may be extrapolatable to the professional category.

Rider age is not included as this variable could hinder the performance of the model in several ways. First of all, there exists a ‘too-good-too-soon’-effect, where very talented riders turn professional at an extremely young age resulting in a somewhat disappointing start of their careers. This would translate into a low amount of PCS points scored in the first two years of a professional career. A good predictive model would then pick up the combination of young age and good youth results as indicative of bad performance. At first this would seem like a good prediction as the first two years could be underwhelming. Nevertheless, these riders are still talented and should be targeted. Therefore, simply looking at performance would filter out the presence of this effect, leading to the detection of talented, extremely young riders. Second, an even more detrimental cohort effect could be in play as well. For example, it could be that the set of 19–21-year-old riders in the training sample (i.e., the ones that turned professional at that age) is more talented than the 23–25-year-old sample. In that case our model would again put much emphasis on age as a predictor, whereas the impact of age is actually based upon an external effect.

Tables 1 and 2 provide an overview of all the (groups of) features and how many features there are per feature group. In Table 2 *best result*, *participation*, and *minimum time difference* are computed for the 24 stage races and 36 one day races, but *stage wins* and *stage best results* are only computed for the stage races. In total there are 242 features: 14 aggregated features + 24 stages races x 5 features + 36 one day races x 3 features. To give some further insight in how these features are created, consider the results from the stage race Sint-Martinusprijs Kontich (Junior race) depicted in Figure 1. The depicted rider won the first two stages (i.e., position 1; indicated in black): a team time trial (TTT) and an individual stage (no special indication). This resulted in him leading the general classification after stage 2 (i.e., position 1; indicated in yellow). He dropped to place 5 after his 23rd place in the individual time trial (ITT), his further results in stages 3b and 4 (i.e., 82nd and 35th) had no further impact on his general placing, resulting in his 5th place in the general classification and 9th in the points classification.



05.07 › 08.07			Sint-Martinusprijs Kontich (2.1)
	9		Points classification
	5		General classification
08.07	35		Stage 4 | Kontich › Kontich
07.07	82	5	Stage 3b | Kontich › Kontich
07.07	23	5	Stage 3a (ITT) | Kontich › Kontich
06.07	1	1	Stage 2 | Antwerp › Kontich
05.07	1	5	Stage 1 (TTT) | Kontich › Kontich

Figure 1: Results from one rider in the sample used for example calculation. Rider would have one stage win + best result 5th.

Table 1: Aggregate features: feature names and description. Features are calculated across all races.

Feature Name	**Description**
# Results	Total number of scraped youth results
# Abandons	Total amount of abandons among results
Abandon ratio	Ratio results/abandons
Victory ratio Junior	Number of victories in Junior category divided by number of Junior results
Podium ratio Junior	Number of podiums (places 2-3) in Junior category divided by number of Junior results
Top 5 ratio Junior	Number of top-5's (places 4-5) in Junior category divided by number of Junior results
Victories Junior	Absolute number of victories as Junior
Victory ratio U23	Number of victories in U23 category divided by number of U23 results
Podium ratio U23	Number of podiums (places 2-3) in U23 category divided by number of U23 results
Top 5 ratio U23	Number of top-5's (places 4-5) in U23 category divided by number of U23 results
Victories U23	Absolute number of victories as U23
Evolution wins	Number of victories in U23 category divided by wins in Junior category (0 if divided by 0)
Evolution podium ratio	Podium ratio U23 category divided by Podium ratio Junior category (0 if divided by 0)
Evolution wins	Top 5 ratio U23 category divided by Top 5 ratio Junior category (0 if divided by 0)

Table 2: Non-aggregate (single race-based) features by group: feature names and description. Features are calculated per race. Stage races have five features, one day races three.

Feature Group	Type of Race	Number Features	Example Features
Best Result	Stage	24	*giro-ciclistico-d-italia--GC-best-result*
	One day	36	*gp-industria-e-commercio2--best-result*
Participation	Stage	24	*driedaagse-va* *n-axel--participation*
	One day	36	*liege-la-gleize--participation*
Minimum Time Difference	Stage	24	*tour-des-pays-de-savoie--minimum-TimeDiff*
	One day	36	*Tour-des-flandres--minimum-TimeDiff*
Stage Wins	Stage	24	*grand-prix-ruebliland--stage-victory, tour-du-pays-de-vaud--stage-victory*
Stage Best Results	Stage	24	*trofeo-karlsberg--stage-best-result, int-junioren-rundfahrt-niedersachsen--stage-best-result*

If a rider would only have these race results, he would have the following feature values: a value of 6 for number of results (5 stages + general classification), a Junior victory ratio of 0.20 (1/5: his win in stage 2, the win in stage 1 is not counted in either the nominator or the denominator as this was a team time trial), a podium ratio in Junior races equal to 0 (no placings 2 or 3), a top 5 ratio equal to 0.20 (result in general classification), and the number of abandons and abandon ratio equal to 0 as there was no abandon, just like all other aggregate features (e.g., victory ratio U23). The participation feature of the Sint-Martinusprijs Kontich receives a value of 1, while all other participation values receive a value of 0, as no participation to other races was recorded. His best GC result in the race would be a value of 5, and the minimum time difference 30 seconds (not displayed in Figure 1). The stage wins and stage best result features would both receive a value of 1. The rider would receive missing values for all other non-aggregate features. This would imply a total of 164 missing values for the rider in question (59 *best result* + 59 *minimum time difference* + 23 *stage wins* + 23 *stage best results*).

3.3. Value Imputation Overview

Our sample contains a very high number of missing values, with only 49.57% of all the possible feature values observed since rider and coaches choose the races which suit the rider's capacities best. This results in an atypical situation in which no single complete case (i.e., an observation which has a value for each unique feature) is observed in the data, this while most data analytical models can only be used with complete datasets (Kowarik & Templ, 2016). A typical method of handling missing values in a predictive setting is value imputation. However, most of these methods need to be adjusted when dealing with extreme missing-rates (Piri, 2020).

The three most popular single imputation methods are: mean imputation, regression imputation, and k-nearest neighbours (KNN) imputation (Jadhav, Pramod & Ramanathan, 2019). Mean imputation replaces the non-observed values with the mean of the observed values of the variable and is commonly used due to its simplicity (e.g., Piri, 2020; Dolatsara et al., 2020). Regression imputation uses a regression model, which can be any type of regressor, to predict the missing values, by using the complete cases as training set and the missing cases as deployment set. KNN imputation (Troyanskaya et al., 2001) is similar to regression imputation as it also uses the neighbours of the missing case from the complete cases to see which average value the *k* nearest neighbours have.

Mean imputation can be directly implemented as no complete cases are needed. This ease-of-usage might explain the popularity of the method despite the large reduction in data variance it induces. On top of this, we also observe the simple technique to yield highly competitive results towards final predictive performance when extremely high missing-rates are encountered (e.g., Luo et al., 2018). Regression imputation and KNN imputation are less directly applicable due to the need for complete cases to estimate missing values.

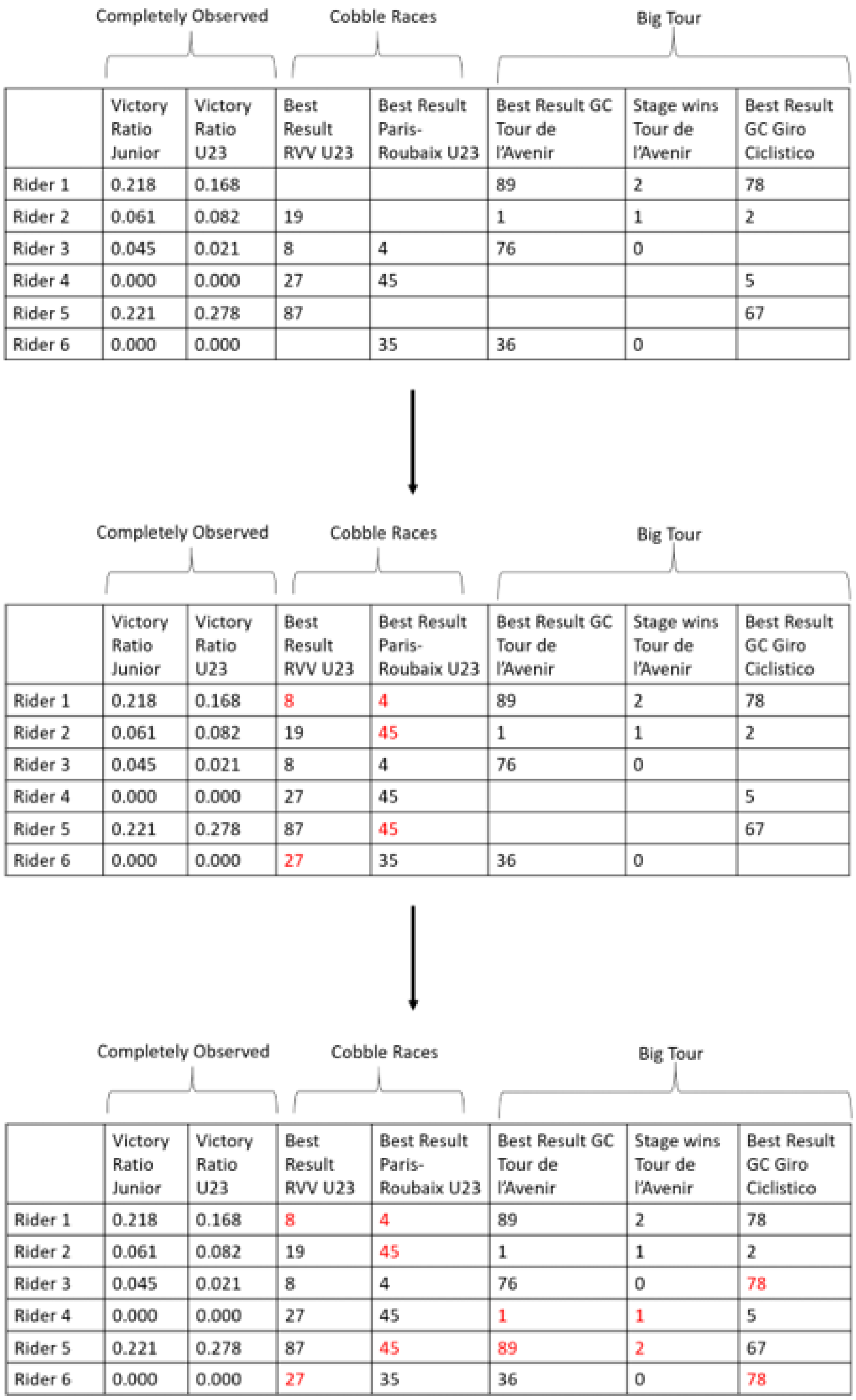
Multivariate Imputation by Chained Equations (MICE; Van Buuren & Groothuis-Oudshoorn, 2010) is a solution proposed to handle this issue for regression imputation. The method iteratively regresses estimates for the missing values after an initial random imputation. We set the number of iterations at 10, acting as a trade-off between computational time and reaching of convergence. As base regressor, we choose random forest (Breiman, 2001) due to the algorithm's capacity to handle non-linear relationships as well as its good performance without parameter tuning (Fernández-Delgado, Cernadas, Barro & Amorim, 2014).

KNN imputation has no adaptation that handles situations without complete observations. This also translates to the situation where KNN imputation is only used in situations where complete cases are observed. This makes the algorithm unsuited for our problem at hand. Nevertheless, the algorithm is identified as the best imputation method for predictive modelling (Jadhav, Pramod & Ramanathan, 2019). Therefore, we suggest an alternative method for value imputation, where we group the races based on domain knowledge into groups that do have complete cases on which KNN imputation can be applied (see Appendix Table A1 for the assigned imputation group per race). The addition of this imputation method brings the number of tested imputation methods to three: *simple mean imputation*, *chained equation regression imputation*, and *race group-based KNN imputation*. Each of the three imputation methods will be deployed for the other steps in the experimental set-up.

3.4. Proposed Imputation Method

We create eight race categories in total. A first category is the *Big Tour* category, which are races that take place during a period of over a week and over varied terrain. Diverse riders with good recuperation skills excel in the overall classification of this type of race. The importance of the races also attracts riders from quite wide geographical origins and the longitude and importance of the races make it more interesting for some to solely focus on stage victories rather than the overall classification. A related category is the *Stage Race Climb* category of French stage races over very mountainous terrain, attracting many riders from France and neighbouring countries who can climb well. Both categories exist for the U23 stage races, while Junior stage races are all summarized into one *Stage Race Junior* category, which is more diverse. This due to the fact that Juniors have more limited calendar options. Regarding the one-day races, we also follow a similar method, with the *One Day Junior* races forming one category, and the U23 races divided into *Cobbles* and *Hilly U23*. Cobbled races are quite unique as they are the sole type of races which favor more heavy riders, while also being located in and near Belgium. This as opposed to the hilly one-day races, which are on a hilly terrain, favoring more light-weight riders, while also being primarily located in Italy. All other races are categorized as *Rest.*

Our extension to the KNN-imputation method is explained in Figure 2 in a simplified example with only 6 observations, 7 features, and 2 imputation categories (as opposed to <1,060 observations (fold-dependent), 242 features and 8 categories in the full data set). We observe the data as it is before imputation in the upper panel. The features are divided into groups, based on whether they are completely observed and the above discussed imputation groups. If we would use traditional KNN imputation, we would impute the missing observations (indicated by white blocks), using the non-missing values (black numbers) of complete cases. However, in our case, there are no complete cases (i.e., there is no single row where all cells are filled). Yet, if we would split up the data based on imputation category, we do get complete cases. This split would result into two groups: *completely observed + cobble races* and *completely observed + big tour*. Note that we always incorporate *completely observed* in order to have more accurate estimations. In Figure 2, this would result into riders 3 and 4 being complete cases of *completely observed + cobble races* and riders 1 and 2 being complete cases of *completely observed + big tour.* This enables KNN imputation for all features in both groups. In this simple example, we set K (i.e., the number of neighbours considered) equal to 1. In the middle panel, this is deployed for the *completely observed + cobble races* group. For instance, rider 1 gets the same results as rider 3, as he is more similar to rider 3 than to rider 4. Similarly, rider 6 gets an imputed result, based on his similarity to rider 4. In the lower panel, this methodology is repeated for the *completely observed + big tour* group of variables, using riders 1 and 2 as complete cases, thus not using the results created in the middle panel. When combining the results from the two steps, one gets a fully imputed data set as is displayed in the lower panel. Note that in the actual case we use more observations and set K equal to 5, rather than 1, leading to more reliable estimates.



	Completely Observed		Cobble Races		Big Tour		
	Victory Ratio Junior	Victory Ratio U23	Best Result RVV U23	Best Result Paris-Roubaix U23	Best Result GC Tour de l'Avenir	Stage wins Tour de l'Avenir	Best Result GC Giro Ciclistico
Rider 1	0.218	0.168			89	2	78
Rider 2	0.061	0.082	19		1	1	2
Rider 3	0.045	0.021	8	4	76	0	
Rider 4	0.000	0.000	27	45			5
Rider 5	0.221	0.278	87				67
Rider 6	0.000	0.000		35	36	0	

	Completely Observed		Cobble Races		Big Tour		
	Victory Ratio Junior	Victory Ratio U23	Best Result RVV U23	Best Result Paris-Roubaix U23	Best Result GC Tour de l'Avenir	Stage wins Tour de l'Avenir	Best Result GC Giro Ciclistico
Rider 1	0.218	0.168	8	4	89	2	78
Rider 2	0.061	0.082	19	45	1	1	2
Rider 3	0.045	0.021	8	4	76	0	
Rider 4	0.000	0.000	27	45			5
Rider 5	0.221	0.278	87	45			67
Rider 6	0.000	0.000	27	35	36	0	

	Completely Observed		Cobble Races		Big Tour		
	Victory Ratio Junior	Victory Ratio U23	Best Result RVV U23	Best Result Paris-Roubaix U23	Best Result GC Tour de l'Avenir	Stage wins Tour de l'Avenir	Best Result GC Giro Ciclistico
Rider 1	0.218	0.168	8	4	89	2	78
Rider 2	0.061	0.082	19	45	1	1	2
Rider 3	0.045	0.021	8	4	76	0	78
Rider 4	0.000	0.000	27	45	1	1	5
Rider 5	0.221	0.278	87	45	89	2	67
Rider 6	0.000	0.000	27	35	36	0	78

Figure 2: Imputation using feature categorization. Missing features are imputed per race category: in this example cobble races vs. big tours. E.g., Paris-Roubaix results do not influence Tour de l'Avenir imputations.

3.5. Feature Selection

Our extensive feature engineering leads to 242 features in total, which is an enormous amount in relation to the sample size of 1,060 athletes. As an implication, a feature selection method has to be applied to cope with the curse of dimensionality (Jain, Duin & Mao, 2000). Without deploying a feature selection method, the analytical models will be slower, less comprehensible, and likely to perform worse in terms of accuracy (Kursa & Rudnicki, 2010).

Two broad types of model-agnostic selection algorithms exist: filter and wrapper methods (Guyon & Elisseeff, 2003; Verbeke et al., 2012; Kocheturov, Pardalos, & Karakitsiou, 2019). While filter methods only look at the features independently of each other, wrapper methods also try to filter out redundant features. These are features that do correlate with the dependent variable but contain no *additional* information over the already included features. This results in a more optimal set of features when compared to filter methods.

A very popular wrapper method is the Boruta algorithm (Kursa & Rudnicki, 2010). The base algorithm used is random forest and the method is based on the idea of 'shadow variables'. These are created by replacing the actual feature values with random permutations of these values. When the resulting feature importances are not significantly lower than the actual feature importances, it is decided that the feature in question is redundant and can be excluded from the eventual feature list. Typically, the mean decrease in impurity (Gini coefficient) or accuracy is used as estimator of variable importance. However, these methods are unstable and have no theoretical foundation (Molnar, 2020). A solution to these shortcomings lays in the use of Shapley additive explanations (SHAP; Lundberg & Lee, 2017). By averaging the SHAP values per feature across all individual predictions, one can derive the importance of that variable, resulting in stable estimates with a theoretical foundation. These importances are used to statistically test the differences between actual features and shadow features in the Boruta algorithm. A more elaborate explanation of the SHAP framework is given in Section 3.8. Interpretability. Note that the feature selection step is performed after the value imputation step, but before any algorithm is applied.

3.6. Algorithms

Due to its interpretability and simple mathematics, Ordinary Least Squares (OLS) regression is widely used in the field of statistics and predictive analytics (Gujarati & Porter, 2009). By minimizing the squared error for the parametric formulation of an additive linear model, one can create an easily interpretable model, which can be quite competitive in situations where the underlying relationship is not overly complex. This is also observed in the field of cycling analytics, where it is commonly included in the list of tested algorithms and where ridge regression is the only algorithm which has outperformed XGBoost in any cycling-related setting (De Spiegeleer, 2019). We implement a simple linear regression algorithm, rather than ridge regression as it is assumed that both algorithms would act very similar due to the regularization effects being canceled out due to the Boruta-SHAP feature selection step. This was also validated in preliminary testing.

Next to OLS, we also include a regression tree using the Classification And Regression Tree (CART) implementation. This algorithm is easily interpreted, while the splitting mechanism allows for some more variation in the fitted relationship compared to the linear relationship in OLS regression. In regression trees, splits are made in order to maximally decrease either the mean-squared-error. The observations in the 'leaves' at the end of the tree are averaged to get the dependent value . Decision trees are extremely likely to overfit, which can be handled by cost-complexity pruning (Breiman, Friedman, Stone & Olshen, 1984). By adding a complexity term ($\alpha|Terminal\ Nodes|$) to the cost of the tree, the tree is penalized for unnecessary splits. Larger values for the cost-complexity parameter (α) result into heavier penalization and 'smaller' trees, hence cross-validation of the cost complexity parameter is necessary.

To minimize the danger of overfitting, Breiman (2001) proposed the use of random forest models in regression. These models are ensemble learners which combine the regression outcome of multiple single regression trees. These single regression trees are then iterated several times, with slight changes in order to make the final averaged result more robust. Two important steps are used to ensure enough variability across trees: (1)each tree is trained on a new bootstrap copy of the training data set rather than the actual training set, and (2)during the splitting (i.e., creating branches of the tree) only a random subset of variables is considered. The algorithm is selected in our experimental set-up as it is very robust and performs well without heavy parameter tuning. The number of trees is set sufficiently large at 500 and the number of random predictors to select at each tree split equals the square root of the number of predictors.

Gradient boosting algorithms learn to predict the error by iteratively adjusting these towards the optimum, which means that the loss function is optimized based on the gradient. Instead of only using first order derivates of the loss function, XGBoost (Chen & Guestrin, 2016) also uses second order derivatives to reach the optimum. This allows for a computationally efficient calculation of the most accurate predictions. This high performance is also translated into related cycling analytics studies, where the algorithm is consistently identified as the top performer (e.g., De Spiegeleer, 2019; Karetnikov, 2019; Kataoka & Gray, 2018). While being highly performant, the algorithm is also extremely sensitive to the hyperparameter settings. Therefore, careful tuning of these parameters is required (Chen & Guestrin, 2016).. An important parameter is the number of boosting rounds, as this determines how many iterations are used to determine optimal tree structure. Just like in regression trees, the optimal structure is also determined by the cost-complexity parameter α. The learning rate is also checked as this determines improvement step size which ensures an optimum between non-convergence and local optima being fitted. Maximal tree depth is also validated, as this performs regularization . Finally, both the default squared error and the Tweedie loss function are considered as objective function.

Since the overall number of observations is rather limited (1,060), no deep learning methods are deployed, as they are assumed to underperform when dealing with small training samples (e.g., Kataoka & Gray, 2018). Rather, we add a shallow multilayer perceptron that uses a feed-forward propagation algorithm optimized by BFGS with one hidden layer. This simple implementation is preferred in real-life business cases with limited sample size over deeper structures and gradient descent-based optimizers (Dreiseitl & Ohno-Machado, 2002). In this implementation, the weight decay and hidden layer size are the most important parameters to validate as these determine model complexity (Bogaert, Ballings and Van den Poel, 2018). The candidate values for all implemented algorithms are summarized in Table 3.

Table 3: Candidate search grids. Hyperparameters were optimized per algorithm through an exhaustive grid search. Values display validated parameters with their candidate settings.

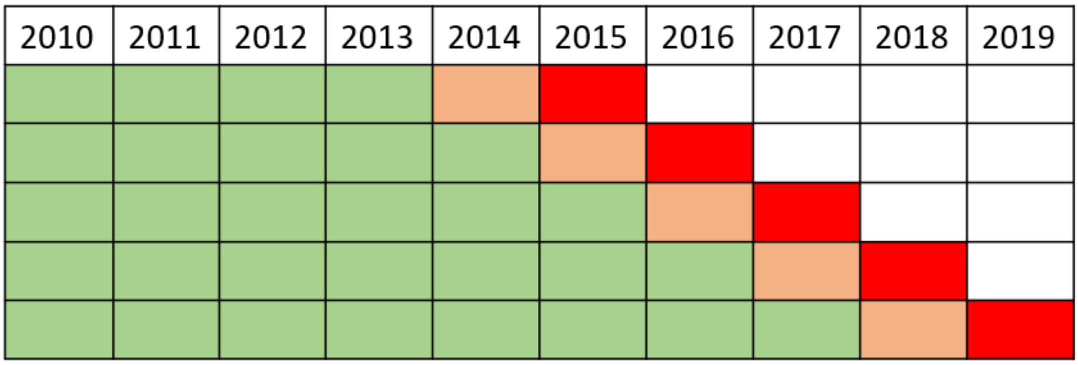
Algorithm	**Parameter**	**Candidate Settings**
Linear Regression	/	/
Decision Tree	α	[0.001,0.005,0.01,0.02, 0.05, 0.1, 0.2, 0.5, 1, 2]
Random Forest	/	/
XGBoost	Learning rate	[0.1,0.3,0.5,0.7,0.9]
	Boosting round	[100,200,300,400]
	α	[0,4,8,12,16,20]
	Maximal tree depth	[6,8,10,12]
	Objective function	[Squared error, Tweedie]
Perceptron	Decay	[0.001, 0.01, 0.1]
	Hidden layer size	[1, 2, 3, …, 20]

3.7. Experimental Set-up

As a season follows the subsequent one and riders compete against each other in the same season, rather than act as individualistic competitors, one can safely say that the assumption of independent and identically distributed data is clearly violated. This influences our experimental set-up, as a traditional cross-validated approach is not adequate in this situation. Rather, we follow a rolling window approach where all available information is used up until the moment of prediction (Vomfell, Härdle & Lessmann, 2018). In order to have an unbiased estimation of performance, we use five different periods for testing: starting years 2015-2019. The process is visualized in Figure 3, with the green training period for building the model, orange validation period for tuning the hyperparameters (and final training) and red testing period for evaluating final performance. A single train-validation split is used to optimize the hyperparameters as is common in predictive modelling studies (Schetgen, Bogaert & Van den Poel, 2021). Hence, the validation period is only used for hyperparameter tuning and the combined training and validation period is eventually used for fitting the final model. This process is repeated five times (i.e., once per evaluation period) in order to ensure robust results.

The rolling window approach is deployed for each value imputation – regression algorithm combination as discussed above. Since XGBoost has the unique advantage that it can cope with missing values without imputation required by treating these missing value as a unique observation value, we also deploy XGBoost without any imputation method. By adding this implementation, we have a baseline model that can validate the added value of the various imputation methods. This baseline method does not involve a feature selection step, as this computation is infeasible with the missing values. Besides a model-based benchmark, another baseline method is added based on a simple but sensible heuristic. We look at the rider's consistency in the last five years of his youth career by determining a weighted average of the top-10 ratios during this period. Eq. (1) computes our heuristic-based benchmark with t being the number of years before the start of the career. For instance, the top10-ratio achieved 2 years before the start of the rider's career is given a weight of 0.50 (1/2). In total we compare 17 unique configurations: (3 imputation methods x 5 regression algorithms + 2 baseline models). Table 4 provides an overview of the deployed configurations and how they are referred to in the results.

$$\sum_{t=1}^{5} \frac{top10\ placings\ (-t)/participations\ (-t)}{t} \qquad (1)$$



2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
🟩	🟩	🟩	🟩	🟧	🟥				
🟩	🟩	🟩	🟩	🟩	🟧	🟥			
🟩	🟩	🟩	🟩	🟩	🟩	🟧	🟥		
🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟧	🟥	
🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟧	🟥

Figure 3: Rolling cross-validation time window. Green stands for training, orange for validation, and red for testing.

Table 4: Overview Configurations. Each configuration was evaluated through the 5k rolling cross-validation. Two baselines were considered and 5 algorithms x 3 imputation techniques.

Model Name	Imputation Method	Algorithm
BASELINE 1	None	XGBoost
BASELINE 2	None	Aggregation Heuristic
linreg_knn	Grouped KNN	Linear Regression
dt_knn	Grouped KNN	Decision Tree
xgb_knn	Grouped KNN	XGBoost
rf_knn	Grouped KNN	Random Forest
mlp_knn	Grouped KNN	Perceptron
linreg_mean	Mean Imputation	Linear Regression
dt_mean	Mean Imputation	Decision Tree
xgb_mean	Mean Imputation	XGBoost
rf_mean	Mean Imputation	Random Forest
mlp_mean	Mean Imputation	Perceptron
linreg_regression	Chained Equation Regression	Linear Regression
dt_regression	Chained Equation Regression	Decision Tree
xgb_regression	Chained Equation Regression	XGBoost
rf_regression	Chained Equation Regression	Random Forest
mlp_regression	Chained Equation Regression	Perceptron

3.8. Evaluation

A typical way of measuring the predictive accuracy of a continuous outcome is by calculating the root-mean-squared-error (*RMSE*) between predicted value $\hat{y}_i$and observed value y_i, according to the formula depicted in Eq. (2), with N the number of observations.

$$\sqrt{\sum_{i=1}^{N}\frac{(\hat{y}_i - y_i)^2}{N}} \quad (2)$$

However, this measure is only indicative of how well the algorithm predicts the eventual points scored, which is not in line with the goal of most cycling teams. Rather than having the most accurate estimates, cycling teams want an estimation of how the riders will perform comparatively against each other. Some generations may be more or less talented than others, yet the teams will still want to have the most performant riders per generation. This entails that the ordering of the algorithm is important. The strongest riders should be ranked on top, regardless of the eventual points they will be scoring. A good measure to compare the actual rankings with the predicted rankings is the *Spearman rank correlation*, which is a nonparametric technique for evaluating the degree of linear association between two independent variables (Gauthier, 2001). Compared to the traditional Pearson correlation, it operates on the ranks of the data rather than the raw data, which makes it highly suitable for the task at hand. It is calculated according to Eq. (3), with d_i being the difference between ranks of predicted value $\hat{y}_i$and observed value y_i.

$$\frac{1 - 6\sum_{i=1}^{N} d_i}{N^3 - N} \quad (3)$$

Another interesting way of dividing professional athletes is by assigning them into the top 10%, top 25%, or top 50% buckets of all athletes (Persson et al., 2020). By doing so, you divide the athletes into an absolute top bin, a just-below-the-top bin, an intermediate bin, and a less successful bin. One major advantage of applying this technique to both the predicted values $\hat{\boldsymbol{y}}$ and the observed values $\boldsymbol{y}$ is that we can deploy typical measures from classification tasks. Accuracy calculates how many riders are in the same bin in the observed values $\boldsymbol{y}$ and the predicted values $\hat{\boldsymbol{y}}$. However, an issue with simple accuracy is the fact that the measure does not incorporate the fact that we are working with a continuum. For instance, consider a rider which the algorithm ranks between the 9th and 10th percentile and the observed rank is between the 10th and 11th percentile. It is obvious that this is a small error, yet accuracy will count this as a misclassification. On the other hand, if the algorithm ranks the observation between the 20th and 21st percentile, this will be regarded as correct as both are in the same bin (top 25%). While such errors are inherently linked to the binning of continuous observations, *accuracy within n* (Gaudette & Japkowicz, 2009) filters out this type of misevaluation. Predicted classifications that are n (in our case 1) classes of the actual classification are also considered correct. For instance, someone that actually ranks in the top 10% will be evaluated as accurate if he or she is predicted to be in either the top 10% or top 25%. This measure may be too lenient but does give a fairer representation of the risks the team managers are taking with decisions based on the model. Therefore, we report accuracy within one rather than accuracy.

Of special interest to the professional teams, is the absolute top bin of the top-10% riders. These riders are the ones they want to contact by preference. By considering this bin as the desired class, we can deploy typical measures from binary targeted marketing campaigns such as churn or acquisition. A popular measure based on those top decile bins is the *lift* which compares the actual top 10% riders in the suggested bin to the actual rate (i.e., 10%) in the dataset (Eq. 4). In other words, the lift derives how much better the model is compared to randomly contacting riders

$$Lift = \frac{Number\ of\ suggested\ riders\ who\ are\ actually\ top10\% - performers}{actual\ overall\ rate} \quad (4)$$

All four performance measures (i.e., RMSE, Spearman rank correlation, accuracy within one, and lift) are calculated for each possible configuration in each fold. Solely for baseline 2 (weighted average top-10 ratios) the RMSE is not calculated, as this baseline simply ranks the riders and does not provide point estimates. This means that RSME is calculated 80 times (16 configurations x 5 evaluation periods), and all other metrics 85 times (17 configurations x 5 evaluation periods) in the final evaluation step.

3.9. Interpretability

As discussed above, interpretability is particularly important for sports scouting tools. Prediction algorithms can be divided into white-box and black-box models. The former have the advantage of being interpretable, while the latter are much harder to comprehend, leading to ‘blind’ suggestions. We have selected two white-box algorithms (i.e., linear regression, and regression tree) and three black-box algorithms (i.e., XGBoost, random forest, and multilayer perceptron), which may suffer from the accuracy-interpretability trade-off. However, over the past years many efforts have been made to make these black-box models more interpretable, with Shapley additive explanations (SHAP; Lundberg & Lee, 2017) being the current state-of-the-art in interpretable data science. SHAP combines strengths of local surrogate models with Shapley values by creating an additive feature attribution method. In other words, it builds a model of the predicted values, with a linear addition of input variables. Those ‘attributions’ are embedded in game theory, with each feature being a player and the prediction error being the ‘game’ outcome. By doing so, it explains the marginal contribution of each feature towards the individual predictions. This means that the model will not only tell which riders to focus on, but also why this rider is selected and what features increase or decrease the rider’s ranking.

While this methodology is useful to teams to explain the ranking of an individual rider, one downside of this implementation is that it does not discriminate well enough towards the different rider types. Four main types of riders exist, both on the professional level as well as during youth categories: flat terrain, uphill, all terrain, and sprinters (Menaspà et al., 2012). This is interesting, as these types would be interpretable for professional team scouts. By comparing how the new targets compare against current top performers during their youth period, scouts can have a decent idea about the rider type. An easy-to-interpret visualization tool to compare sport performance are spider charts (Blom, 2019). However, a spider chart only works well with a limited number of variables. Therefore, we derive the principal components of the features to uncover the underlying rider types, which are then visualized on the resulting spider chart. By comparing the prospect’s value on the chart against current top performers who were in the training sample, the scout should gain insights into the rider type of the prospect. Thus, while the SHAP framework is used for explaining individual performance predictions, principal component analysis will be used for indicating rider type, independently from the expected performance.

4. Results & Discussion

4.1. Results

As discussed in the Methodology the performance measures are calculated per evaluation period. These (five) unique results per configuration-evaluation period tuple are aggregated by reporting the median as the mean would be biased by outlier years. These median results are displayed in Table 5. The top performer per performance metric is indicated in bold, while outperformance of the baseline methods is underlined.

Table 5: Median rolling cross-validated results across period 2015-2019. Underlined values indicate a better performance than the baselines; the top performer is indicated in bold. Grouped KNN imputation before random forest performs best with regard to ranking riders. Chained equation regression imputation before linear regression performs best with regard to identifying top performers.

	RMSE	**Spearman**	**Accuracy Within One**	**Lift**
BASELINE 1 (XGB no impute)	272.13	0.4658	0.8304	3.3333
BASELINE 2 (Heuristic)	/	0.4015	0.8218	2.7272
linreg_knn	265.92	0.4718	0.8571	3.6364
dt_knn	360.97	0.2590	0.7692	1.6667
xgb_knn	**251.86**	0.5001	0.8440	2.7273
rf_knn	261.33	**0.5420**	0.8273	3.3333
nn_knn	338.75	0.2756	0.7778	0.8333
linreg_mean	264.11	0.4659	0.8550	3.6364
dt_mean	344.82	0.2696	0.7818	2.5000
xgb_mean	273.67	0.4200	0.8273	2.5000
rf_mean	271.25	0.4833	**0.8716**	2.7273
nn_mean	328.96	0.0498	0.7321	0.0000
linreg_regression	253.86	0.4810	0.8550	**4.2857**
dt_regression	395.92	0.3168	0.7818	2.1429
xgb_regression	286.06	0.4128	0.8034	2.5000
rf_regression	265.31	0.5355	0.8532	2.5000
nn_regression	313.99	0.2652	0.7768	1.8182

The results indicate that it is feasible to use predictive models to facilitate scouting for professional cycling teams. The top performing algorithms have a Spearman rank correlation between 0.40 and 0.69, which indicates a strong relation between actual performance and predicted performance (Dancey & Reidy, 2007). This means that the resulting lists as provided by the algorithms are good indications of which riders to target. The results imply that the resulting lists from our study are capable of creating business value when provided to the scouts. A similar observation can be made when inspecting the lift score of the models. Several lift scores go above 3.00, indicating that the models would perform over three times as good as randomly selecting riders in the top 10% bucket, with the best performing configuration in this top decile (i.e., linreg_regression) even having a lift above 4.00.

The same conclusions hold for the accuracy of the models. Regarding accuracy within one the learners yield satisfying results with values approaching the 90%, which means that teams can select riders from the top 10% of the list with large confidence that they will not be weak performers (i.e., the bottom 50%). Overall, the algorithms do not score particularly well regarding the RMSE. Hence, our methodology is capable of deriving which riders will perform best, without knowing the actual points performance of the riders.

Several configurations outperform the baseline methods, as indicated by their underlining in Table 5. The configurations linreg_knn, xgb_knn, rf_knn, linreg_mean, rf_mean, linreg_regression, and rf_regression outperform both baselines across most of the reported metrics. Interestingly, we observe that single regression tree, and 1-hidden-layer-perceptron are consistently among the weak performers. Our suggested KNN adaptation is the only imputation method which, in combination with XGBoost, is capable of outperforming the baseline methods. This implies that the proposed KNN method is the most informative imputation method, as it is the only imputation method which seems to add value to the XGBoost regression algorithm compared to XGBoost without imputation. With regard to the baseline methods, we notice that baseline 1 (i.e., XGBoost without imputation) outperforms baseline 2 (i.e., heuristic method). This indicates that advanced analytical approaches are preferred over simple heuristics. In general, the baseline methods never outperform the imputed predictive learners. We observe the differences to be quite large, indicating the added value of the proposed method above non-handling of missing values and the use of simple heuristics.

Table 6: Results riders turning professional in 2019: underlined if better than baselines; top performer in bold. Reported results are based on unique measurement from last fold. Results are biased through COVID19-influenced season 2020. Nonetheless, the methods perform adequate besides an increase in RMSE.

	RMSE	**Spearman**	**Accuracy Within One**	**Lift**
BASELINE 1 (XGB no impute)	458.93	0.4931	0.7890	0.9090
BASELINE 2 (Heuristic)	/	0.4553	0.8315	3.3333
linreg_knn	435.33	0.4718	**0.8991**	3.6364
dt_knn	493.96	0.2590	0.7523	0.9091
xgb_knn	444.22	0.4271	0.8440	2.7273
rf_knn	430.56	0.5308	0.8257	2.7273
nn_knn	479.37	0.2756	0.7982	3.6364
linreg_mean	437.23	0.4396	0.8716	3.6364
dt_mean	501.09	0.2685	0.7798	1.6667
xgb_mean	489.48	0.3748	0.8257	1.8182
rf_mean	432.85	0.4833	0.8716	2.7273
nn_mean	494.54	-0.2409	0.6330	0.0000
linreg_regression	435.67	0.4810	0.8807	**4.5455**
dt_regression	521.40	0.2590	0.7982	0.9091
xgb_regression	433.02	0.4081	0.8257	3.6364
rf_regression	**417.00**	**0.5779**	0.8532	3.6364
nn_regression	466.07	0.3714	0.7890	1.8182

The performance across the various evaluation periods was observed to remain fairly stable, with top performing algorithms having similar performance in most years. This is helpful, as this means that the expected performance of the models is quite reliable. Only in the starting year 2019 (first two years 2019-2020), we observe a significant rise in RMSE as displayed in Table 6. This rise in RMSE is due to the year 2020, where a lot of races were canceled due to the COVID-19 pandemic. This led the algorithms to overestimate the actual points scored. However, the algorithms still score well with regard to the other performance measures. This means that the methodology is capable of still ranking the top performers on top regardless of shocks such as the COVID-19 pandemic.

To investigate whether configurations are significantly different from each other, we apply the Friedman test (Demšar, 2006). The Friedman test indicates significant differences across configurations for the RMSE measure ($\chi_F^2 = 51.26, p < 0.001$), Spearman rank correlation ($\chi_F^2 = 54.00, p < 0.001$), accuracy within one ($\chi_F^2 = 46.14, p < 0.001$), and top decile lift ($\chi_F^2 = 44.25, p < 0.001$). The Nemenyi post-hoc test (Verbeke et al., 2012) is performed for pairwise comparisons between configurations. The test results provide evidence towards the rf_knn configuration being preferred with regard to ranking the riders, as it is capable of statistically outperforming 4 other configurations at the 5% significance level in terms of RMSE and Spearman rank correlation. A similar argumentation can be made towards the extremely weak performance of the nn_mean configuration on all measures besides RMSE. No statistically significant differences are detected among the top performers across the various metrics. Detailed test results can be provided upon request.

With regard to the best overall configuration, one could make two distinct suggestions. The combination of the KNN imputation with random forest regression prediction (rf_knn) is the best in generally ranking the riders. This is demonstrated by its top performance in Spearman correlation, and highly competitive RMSE scores in Table 5, as those measures are the only measures that do not consider the arbitrary binning. The regression imputation with linear regression prediction combination (linreg_regression), on the other hand, is the best method regarding the top decile, as measured by the lift score. The major downside of this methodology may be the longer computation time due to the iterative nature of the imputation method. Overall, both configurations are competitive compared to all other configuration with rf_knn being slightly preferred over linreg_regression when modelling overall rider ranking and linreg_regression slightly preferred when targeting the top decile of riders.

A final argument in model selection can be the time required to come up with suggested rider rankings. The clear bottleneck in model computation is the value imputation step. The computation time of this step per fold is depicted in Table 7. Whereas the grouped KNN and mean imputation methods take only a couple of seconds, the chained equation regression step takes almost 10 hours for the calculation of the largest imputed dataset. This will only further increase with the addition of additional riders to the dataset. As fast imputation allows quick interpretation of new youth race results, this could potentially hinder teams in moving fast with regard of the contacting of a new interesting prospect. Therefore, grouped KNN and mean imputation are suggested above chained equation regression imputation, which is a further argument towards selecting rf_knn over linreg_regression. Overall, we propose two distinct methods depending on the situation. If fast computation times and overall ranking are of main interest, the rf_knn configuration should be deployed. While the linreg_regression solution would be preferred when time is no limitation and when the top decile of riders is of primary interest.

Table 7: Computation time imputation methods (in seconds). Regression imputation is computationally much more expensive than KNN imputation or mean imputation. KNN imputation gives a good trade-off between predictive performance and computation time.

Fold	KNN imputation	Mean imputation	Regression imputation
2015	1.92	0.06	13025.15
2016	1.59	0.03	15035.95
2017	2.46	0.06	20488.19
2018	2.75	0.03	26363.01
2019	3.46	0.03	32997.33

4.2. Algorithm Drivers

The subsequent section elaborates on which features heavily influence the predictions given by the algorithms. As we propose two distinct solutions, both are individually discussed. For each interpretation, we are using the configurations fitted on the dataset of professional riders turning professional until 2018, as the lower points scored in 2020 (from riders starting in 2019) could introduce significant noise in the training data.

Table 8 depicts the resulting coefficient values of the linreg_regression configuration. A first observation is that the Boruta-SHAP feature selection method retains a fairly limited number of features: only seven features are selected. This exclusivity implies the importance of the races that are selected. The Tour des Flandres U23, Tour du Pays de Vaud, and the Driedaagse van Axel should all, although not exclusively, be considered by scouts who want to detect talents that will already score during their first two professional years. It is rather surprising that no features are selected which are based on the very famous youth races Tour de l'Avenir and Giro Ciclistico. The Tour des Flandres U23, Tour du Pays de Vaud, and the Driedaagse van Axel are races that could be regarded as suiting 'classics specialists' (all terrain + flat terrain), while the races Tour de l'Avenir and Giro Ciclistico might attract more sprinters and climbers (uphill). However, it could be that stage race riders and sprinters score better on overall consistency, as measured by Podium ratio U23 and Top 5 ratio Junior. It is remarkable how those two variables dominate the model. The maximal value of Podium ratio U23 in the fitted sample is 0.6667. This would imply that the variable on its own could influence predicted value by almost 1,000 points. The best performing riders as young professionals are thus the ones that are the most consistent in both the Junior and U23 categories. Rather than having one top performance, it is important to be consistent throughout each race. Remarkably, the victory ratio is not selected. It seems that the model favors riders that are always among the top performers rather than the specific rider that has the best end sprint. Note how trend features (comparing consistency evolution from Junior category to U23 category) have no effect on the model.

Table 8: Coefficients final regression imputation - linear regression model. Model highly depends on consistency, as measured through podium ratio U23 and top 5 ratio Junior.

Variable	**Coefficient Value**
Tour des Flandres U23 best result	-0.001
Tour du Pays de Vaud GC best result	-0.061
Podium ratio U23	1391.267
Driedaagse van Axel stage best result	-0.123
Tour des Flandres U23 participation	-2.267
Top 5 ratio Junior	1190.966
Trofeo Comune di Vertova participation	-0.106

Another interesting observation is the positive effect of non-participation to the Tour des Flandres U23 and Trofeo Comune di Vertova. This is especially surprising given the positive effect of a good result in the Tour des Flandres U23. This opposing effect can be explained by the extensive imputation method applied. If one has a good imputed best result in the Tour des Flandres U23, this value is based on a range of other results from the chained equations step. This means that the model implicitly incorporates a wider range of results as one would initially identify based on Table 8. However, this phenomenon reduces the extent to which the model can be regarded as a white-box model. Overall, the model mainly fits on the consistency of the riders in the races as selected for this study. This means that it fails to learn more nuanced differences based on race-based results. This could explain why rf_knn performs better with regard to overall ranking the riders.

Figure 4 visualizes the feature importances as measured by the average SHAP value across all training observations for the rf_knn configuration. While consistency remains the key influencer, we see some large distinctions. First of all, the importance of consistency in the Junior category is removed. It is replaced by the victory ratio in the U23 category. This places much more importance on the winning capabilities and the U23 results. The influence of specific race results is much larger than in the regression imputation - linear regression model. This is not only reflected in the larger importance values, but also in the wider range of races selected. Second, the races picked up by the rf_knn model are more diverse than the race features in Table 8. World Championships U23, Tour de l'Avenir, European Championships, ZLM tour, Tour de Normandie, and Olympia's Tour are all famous youth race events. The fact that these races are all selected, combined with the rf_knn configuration's top performance in terms of RMSE and Spearman correlation, really indicates that this configuration is the best in deriving the drivers of eventual points ranking.

Remember that the feature selection step is performed after the value imputation step, but before any algorithm is applied. This implies that the selected feature set is solely dependent on the used imputation technique. As a result, one can conclude that the adapted KNN method results in the most diverse, informative feature set. This feature set can be leveraged by a complex ensemble method such as random forest. On the other hand, the chained equation regression imputation method yields less interpretable features which works very well with the linear regression model when the sole goal is to target the top decile.

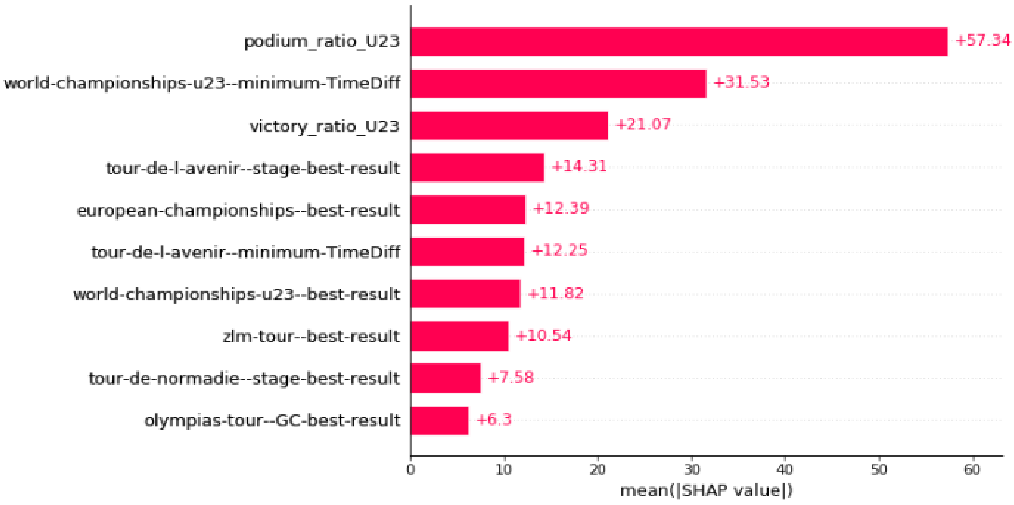


Figure 4: SHAP-based variable importances of KNN imputation – random forest regression model. Model shows a large dependence on consistency, but is also influenced by key races.

The distinctions between the drivers in Table 8 and Figure 4 are remarkable. After contacting a domain expert (i.e., head of development of a WorldTour team), he/she identifies both the specific race-based drivers in Table 8 and Figure 4 as informative. The domain expert, however, anticipates the resulting drivers to reflect slightly different rider types. For having the most diverse set of suggested riders both configurations should be considered and seen as an addition to the current practices, rather than as a complete substitute.

5. Managerial Implications

Table 9: Top 10 predicted top performers according to linear regression model. Several riders have already exhibit good performance at the professional level.

Rider	**Professional Since**
Repa Vojtech	2021
Larsen Niklas	2020
Pidcock Thomas	2021
Colleoni Kevin	2021
Stewart Jake	2020
Meeus Jordi	2021

Hailemichael Mulu Kinfe	2020
Van Gils Maxim	2021
Rodenberg Frederik	2020
Van Wilder Ilan	2020

Both methods are useful depending on the eventual goal of the user. For instance, if a fan wants to know which talents he should keep an eye on, he should simply deploy the linear model and check which riders are suggested as the top performers. The suggested top decile will normally account for 40-50% of the actual top decile. Table 9 depicts the deployment of this technique (linreg_regression) on the riders who turned professional during the years 2020 and 2021. Dependent data from 2019 was not included as the average underperformance of athletes might induce noise in the algorithm.

Of course, the main application lays in its professional usage. The resulting model can be used to both detect new talents, as well as suggest how well current starters will perform. If a rider is heavily underperforming compared to what is predicted by the model, this could indicate that something is going wrong with the talent development of this rider and might lead to changes in how this rider is being deployed in races, and his psychological guidance. Interestingly, we observe several prospects in Table 9 who have already showed some good form at the professional level. For instance, Tom Pidcock already finished in the top-5 of the Strade Bianchi, one of the most important races on the calendar, and Jake Stewart finished second in the Omloop Het Nieuwsblad, another highly prestigious race. The model clearly delivers useful suggestions, but some riders are less performant. Especially riders that started in 2020 can be expected to start showing satisfactory results, such as Hailemichael Mulu Kinfe or Niklas Larsen. A quick inspection shows that Larsen has raced relatively little, probably due to fewer races being hosted due to the global COVID-19 pandemic, yet this discrepancy between performance and expectations might be a good starting point for his team leaders as to ensure if the talent development is going as desired. Hailemichael Mulu Kinfe highlights the overdependence of the model on the two consistency features as he has mainly competed and thrived in some of the lesser competitive youth races.

Note that the usage of the linreg_regression model is mostly useful for teams focusing on the top decile, and thus the top tier of riders. Teams who don’t have the budget to attract such riders, or an insufficiently good reputation could repeat a similar analysis for the actual/predicted ranking of their riders using the rf_knn configuration. The main usage of this method is to be situated in talent identification, rather than talent development. By deploying the algorithm on each rider which is present in the youth results without turning professional, one can identify the most promising youth riders. Again, the used data is scraped before the start of the 2021 season, which means that talents that start doing well during this season will not yet be detected by the model. However, they should be automatically picked up by the algorithm when deployed later on.

Two young riders, Rait Ärm and Luca Colnaghi, are some of the algorithm's top suggestions when deployed on the set of current youth riders. Note that the actual list is much longer, but we limit ourselves to these two suggestions as an example of how the methodology can be used. While the suggested list can already be a useful starting point, a scout will typically want to know more about these riders and why they are selected. When deploying the SHAP framework and building the surrogate model, one can derive the individual explanations per prediction by checking the individual SHAP values per feature and checking which ones have the largest absolute value. A positive SHAP value indicates an increasing influence on the predicted value and a negative SHAP value a decreasing effect. One can then generate a plot based on these SHAP values, with absolute value of the SHAP value visualized on the x-axis. The results for these two predictions are displayed in Figures 5 and 6 with prediction-enhancing SHAP values in red, and prediction-reducing values in blue. The actual feature value is depicted in light grey left of the predictor name, as well as the individual predicted value ($f(x)$) above the main figure and the sample average ($E[f(x)]$) below the x-axis.

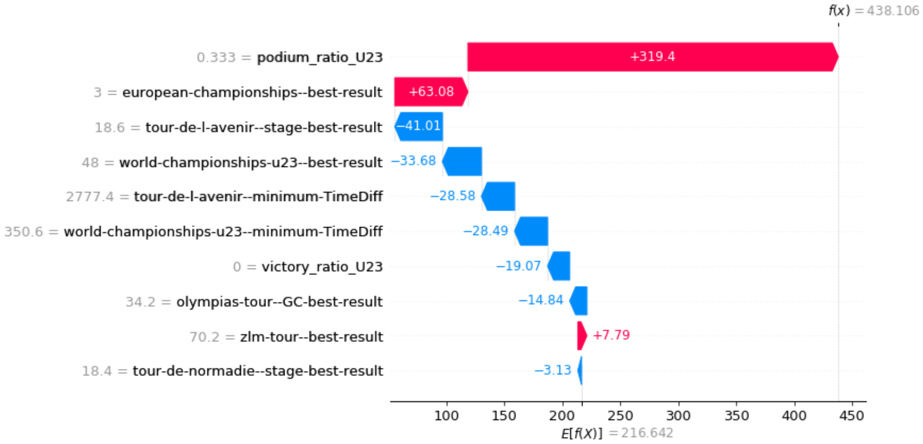


Figure 5: SHAP values Ärm Rait: Positive drivers (red) and negative drivers (blue). Rider is selected due to large number of podiums and European Championship result.

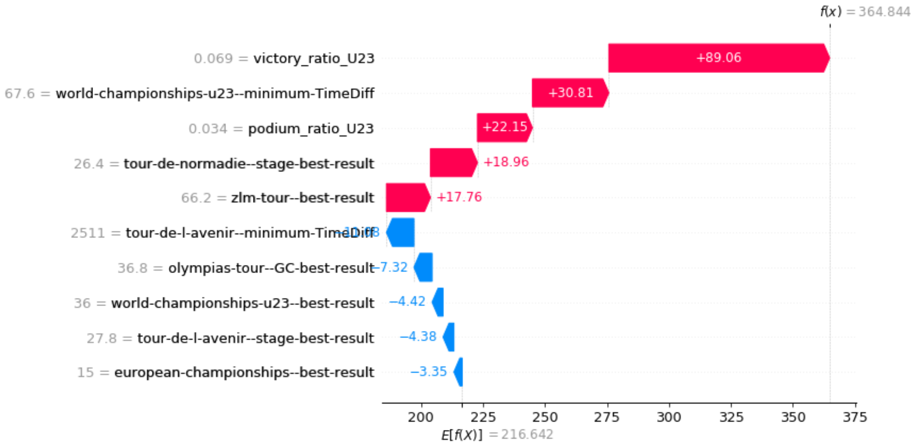


Figure 6: SHAP values Colnaghi Luca: Positive drivers (red) and negative drivers (blue). World Championship driver is imputed: KNN imputation method is capable of imputing the Hilly U23 capabilities into one single feature.

The individual SHAP values explain why the riders receive relatively large scores. For instance, we can observe that Rait Ärm (Figure 5) is selected due to his high number of podium finishes and his good result at the European Championship. His results in other important races and lack of actual wins are however somewhat worrisome. Further inspection of this rider (which can also easily be done by scouts through the procyclingstats website) indicates that he is currently riding for the development team of Groupama-FDJ where he can occasionally ride professional races alongside his youth program. He recently placed inside the top 10 of the GP Monseré, indicating his capabilities against professional riders as well. The methodology could be a push to teams to contact the rider and follow him in other races. His case is a nice example how the method should be used as an additional tool to scouts, initiating further investigation.

Luca Colnaghi (Figure 6) can act as another interesting example. He has several positive indicators, but some of these, such as the World Championship, are imputed. This means that these results are based on his other results. Inspection shows that he indeed is a good performer in (other) hilly classics, which makes this beneficial imputed value feasible. The KNN imputation method is thus capable of imputing the *Hilly U23* capabilities into one single feature, which reduces the importance of many other *Hilly U23* features, which is why relatively few are selected by Boruta-SHAP.

A final usability towards end users should lay in the interpretation of the rider type. Menaspà et al. (2012) identified four rider types: flat terrain, uphill, all terrain, and sprinters. To detect this we apply principal component analysis on the same training data (of riders starting before 2019) after KNN imputation. The derived components are then deployed on the unseen deployment set of non-professional riders. The number of principal components (PC) is set equal to four to derive the rider types as outlined by Menaspà et al. (2012). The results are rather intuitive as they fit very well to four archetypical riders. Only PC4 has to be inversed to reflect climbing rather than non-climbing. We select four riders in the training data set, who are exemplary of the rider types. Filippo Ganna (current ITT world champion) is selected for the flat terrain category, Caleb Ewan (winner of several sprint stages in the Tour) for the sprinter category, Egan Bernal (Tour de France 2019 winner) for uphill specialists, and Michal Kwiatkowski (former world champion) for all terrains. By visualizing the suggested rider in comparison with these riders, the scouts can have a good insight into which type of rider they are dealing with.

These spider plots are displayed in Figure 7 for the above discussed rider Rait Ärm. The rider is visualized as being most similar to the flat terrain specialist Caleb Ewan (i.e., they both have a square-like shape), but however scores lower on all capabilities. This means he is a sprinter, but probably not of world class. Note how our deployment sample only contains riders that are not yet contacted by professional teams and are thus likely lacking the top performers of the 2019 and 2020 seasons in the U23 circuit, who are more likely to be displayed in Table 9.

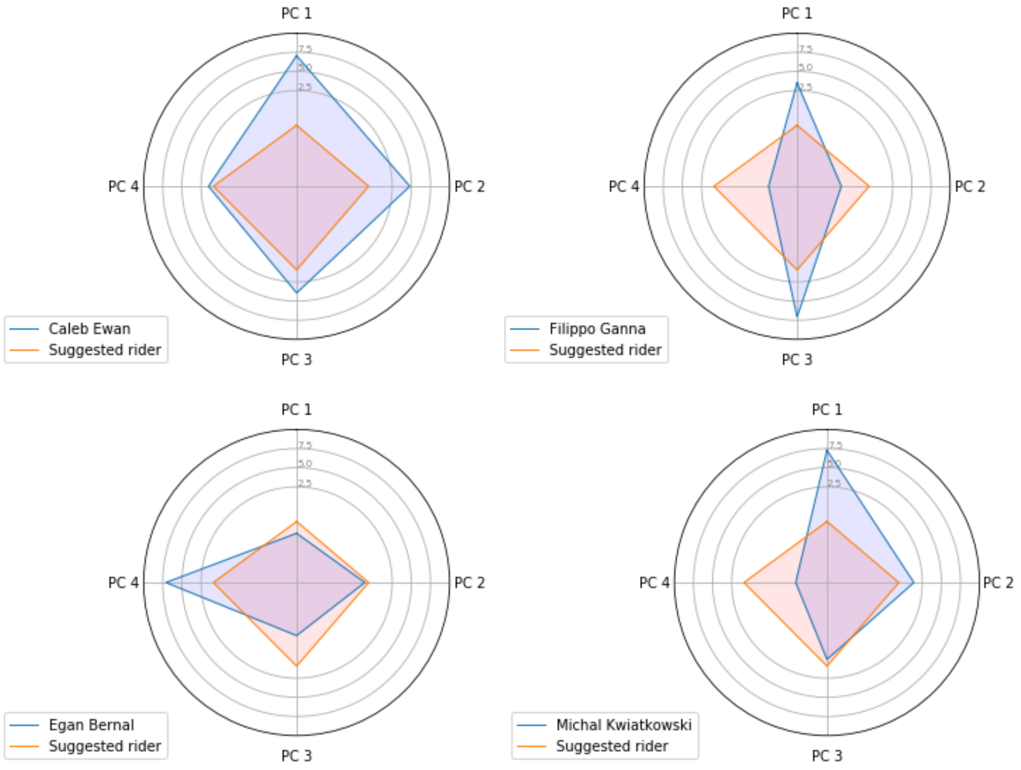


Figure 7: Spider plot Rait Ärm. Four principal components are visualized which represent all terrain (PC1), sprinters (PC2), flat terrain (PC3), and uphill (PC4). Compared with four archetypical professionals. Most similar to sprinter Caleb Ewan.

Finally, one should consider the robustness of the system. A disadvantage of automated applications rises due to the fact that individuals involved could alter their behavior in order to get an incorrect, overly optimistic evaluation. In theory, one could try to peak towards the races whose features are selected by the Boruta-SHAP technique. However, those races are already the most important races on the calendar, where each participant tries to be in top shape. Thus, peaking towards these races would just be similar to what young riders are currently doing. Another strategy could lay in solely participating in races where the athlete expects to perform well. This is, however, already done to a certain extent, explaining the good performance of the grouped KNN imputation technique. Also, riders do not have the sole decision-making power in this regard, as the team coaches must select them for races. On top of this, riders can also be selected for races that they do not desire to participate in. Therefore, we think that , while theoretically feasible, unfair influencing of predicted future performance will not be feasible in practice, as current strategies are already close to rating-optimizing behavior.

6. Conclusions and Future Research

In this paper, we produced an analytical system that retrieves publicly available youth cycling results and uses imputation methods and predictive algorithms to predict future performance of riders who still have to become professional athletes. The results show that the detection of young cycling talents based on youth race results is feasible despite the tendency of the observed data to have many missing values. The resulting algorithms are capable of reaching the most important objective of cycling teams, which is predicting the order of the riders in this points-ranking.

The baseline methods (i.e., XGBoost without value imputation and a simple consistency-based heuristic) were outperformed across all metrics, which indicates the need for high-quality value imputation and the use of predictive regression techniques over simple heuristics. To cope with the high missing value rate in the data we suggest a method that uses expert knowledge on feature groups to form groups with complete cases and then uses KNN imputation on these groups. Our proposed KNN method in combination with random forest is identified as the best overall performer (rf_knn), while being highly competitive in terms of computation time, and is therefore suggested in situations with extreme missing rates. Users that are interested in identifying solely the top tier rider are recommended to use chained equation regression imputation combined with linear regression (linreg_regression).

As a topic for future research, we could include performance results from outside the traditional youth cycling circuit. Recently, the sport witnessed the emergence of top performers from other fields such as cyclo-cross, football, or even ski-jumping. Future research might focus on how to detect potential targets across all sports fields.

Since the independent data can be regarded as sequential data, another research avenue could be to include models suited for sequential data, such as recurrent neural networks. The number of observations is however limited, and this type of model tends to perform quite weak in cases with a limited number of observations (e.g.. Kataoka & Gray. 2018). This, combined with the even greater problem of missing values when using year-per-year data rather than aggregating into best results, make us confident that our approach is the best possible solution.

Our methodology was only deployed and tested for one specific dependent variable. (i.e., the PCS points in first two years of career). However, it should also be deployable for more specific dependent variables such as whether a rider ever wins a cobbled classic or whether he finishes inside the top-10 of the general classification of a grand tour. This would heavily increase the (moving) dependent period to the entire career, rather than a mere two years. The youth results of riders who are currently at the end of their career are not stored well enough to facilitate such an approach at this point in time.

An application that will remain hard in the following years, is the analytical scouting of talented domestiques (team helpers). While the PCS points system already rewards them in a fairer way than the UCI points system, their work remains undervalued, as no scoring system is in place which values the amount of work they put in. In future research efforts, it might be interesting to evaluate diverse ways of quantifying these efforts, based on race reports or video coverage. Recent developments in text mining and image classification (e.g., De Bock & Verstockt, 2020) might prove useful in determining such metrics.

Finally, as our method was only developed for the male professionals as the youth results of female athletes are less well-kept track of. This, combined with the still changing female youth calendar, made our methodology less suited for the female circuit. However, it might be interesting to check if the results are similar when the methodology is applied on female prospects once the desired amount of data is available.

In order to comply with privacy regulations, no personal data was used which was not available to the general public.

References

Andreff. W. (2016). The Tour de France: a success story in spite of competitive imbalance and doping. In *The economics of professional road cycling* (pp. 233-255). Springer. Cham.

Anshel. M. H.. & Lidor. R. (2012). Talent detection programs in sport: The questionable use of psychological measures. *Journal of Sport Behavior*. *35*(3). 239.

Baesens. B. (2014). *Analytics in a big data world: The essential guide to data science and its applications*. John Wiley & Sons.

Blom. C. (2019). *Using Data Analytics to Make the Scouting and Training of Sports Talents More Effective* (Master's thesis).

Bogaert. M.. Ballings. M.. & Van den Poel. D. (2018). Evaluating the importance of different communication types in romantic tie prediction on social media. *Annals of Operations Research*. *263*(1). 501-527.

Breiman. L. (2001). Random forests. *Machine learning*. *45*(1). 5-32.

Breiman. L.. Friedman. J.. Stone. C. J.. & Olshen. R. A. (1984). *Classification and regression trees*. CRC press.

Chen. T.. & Guestrin. C. (2016. August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Dancey. C. P.. & Reidy. J. (2007). *Statistics without maths for psychology*. Pearson education.

De Bock. J.. & Verstockt. S. (2020). GPS driven camera selection in cyclocross races for automatic rider story generation. *In icSPORTS 2020. the 8th International Conference on Sport Sciences Research and Technology Support* (pp. 67-74).

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, *7*, 1-30.

De Spiegeleer. E (2019). Predicting cycling results using machine learning.

Dolatsara. H. A.. Chen. Y. J.. Evans. C.. Gupta. A.. & Megahed. F. M. (2020). A two-stage machine learning framework to predict heart transplantation survival probabilities over time with a monotonic probability constraint. *Decision Support Systems*. *137*. 113363.

Dreiseitl. S.. & Ohno-Machado. L. (2002). Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*. *35*(5-6). 352-359.

Fernández-Delgado. M.. Cernadas. E.. Barro. S.. & Amorim. D. (2014). Do we need hundreds of classifiers to solve real world classification problems?. *The journal of machine learning research*. *15*(1). 3133-3181.

Gaudette. L.. & Japkowicz. N. (2009. May). Evaluation methods for ordinal classification. In *Canadian conference on artificial intelligence* (pp. 207-210). Springer. Berlin. Heidelberg.

Gauthier. T. D. (2001). Detecting trends using Spearman's rank correlation coefficient. *Environmental forensics*. *2*(4). 359-362.

Gujarati. D. N.. & Porter. D. C. (2009). *Basic econometrics*. Tata McGraw-Hill Education. Vol. 3. pp. 55-97

Guyon. I.. & Elisseeff. A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*. *3*(Mar). 1157-1182.

Hilmkil. A.. Ivarsson. O.. Johansson. M.. Kuylenstierna. D.. & van Erp. T. (2018). Towards machine learning on data from professional cyclists. *CoRR abs/1808.00198.*

Jadhav. A.. Pramod. D.. & Ramanathan. K. (2019). Comparison of performance of data imputation methods for numeric dataset. *Applied Artificial Intelligence*. 33(10). 913-933.

Jain. A. K.. Duin. R. P. W.. & Mao. J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*. *22*(1). 4-37.

Johnston. K.. Wattie. N.. Schorer. J.. & Baker. J. (2018). Talent identification in sport: a systematic review. *Sports Medicine*. *48*(1). 97-109.

Karetnikov. A. (2019). Application of Data-Driven Analytics on Sport Data from a Professional Bicycle Racing Team. *Eindhoven University of Technology. The Netherlands*.

Kataoka. Y.. & Gray. P. (2018. September). Real-time power performance prediction in tour de France. In *International Workshop on Machine Learning and Data Mining for Sports Analytics* (pp. 121-130). Springer. Cham.)

Kholkine. L.. De Schepper. T.. Verdonck. T.. & Latré. S. (2020. September). A Machine Learning Approach for Road Cycling Race Performance Prediction. In *International Workshop on Machine Learning and Data Mining for Sports Analytics* (pp. 103-112). Springer. Cham.

Kocheturov. A.. Pardalos. P. M.. & Karakitsiou. A. (2019). Massive datasets and machine learning for computational biomedicine: trends and challenges. *Annals of Operations Research*. *276*(1). 5-34.

Koseler. K.. & Stephan. M. (2017). Machine learning applications in baseball: A systematic literature review. *Applied Artificial Intelligence*. *31*(9-10). 745-763.

Kowarik. A.. & Templ. M. (2016). Imputation with the R Package VIM. *Journal of Statistical Software*. *74*(7). 1-16.

Kumar. A.. Nguyen. V. A.. & Teo. K. M. (2016). Commuter cycling policy in Singapore: a farecard data analytics based approach. *Annals of Operations Research*. *236*(1). 57-73.

Kursa. M. B.. & Rudnicki. W. R. (2010). Feature selection with the Boruta package. *Journal of Statistical Software*. *36*(11). 1-13.

Larson. D. J.. & Maxcy. J. G. (2016). Human capital development in professional cycling. In *The Economics of Professional Road Cycling* (pp. 129-145). Springer. Cham.

Little. R. J.. & Rubin. D. B. (2019). *Statistical analysis with missing data* (Vol. 793). John Wiley & Sons.

Liu. Y.. Schulte. O.. & Li. C. (2018). Model Trees for Identifying Exceptional Players in the NHL and NBA Drafts. In *International Workshop on Machine Learning and Data Mining for Sports Analytics* (pp. 93-105). Springer. Cham.

Lundberg. S. M.. & Lee. S. I. (2017). A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*. 30. 4765-4774.

Luo. Y.. Cai. X.. Zhang. Y.. Xu. J.. & Yuan. X. (2018. December). Multivariate time series imputation with generative adversarial networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (pp. 1603-1614).

Menaspà. P.. Rampinini. E.. Bosio. A.. Carlomagno. D.. Riggio. M.. & Sassi. A. (2012). Physiological and anthropometric characteristics of junior cyclists of different specialties and performance levels. *Scandinavian Journal of medicine & science in sports*. *22*(3). 392-398.

Menaspà. P.. Sassi. A.. & Impellizzeri. F. M. (2010). Aerobic fitness variables do not predict the professional career of young cyclists. *Medicine and science in sports and exercise*. *42*(4). 805-812.

Miller. J.. & Susa. K. (2018). Comparison of anthropometric characteristics between world tour and professional continental cyclists. *Journal of Science and Cycling*. *7*(3). 3-6.

Molnar. C. (2020). *Interpretable machine learning*. Lulu.com.

Nevill. A. M.. Jobson. S. A.. Palmer. G. S.. & Olds. T. S. (2005). Scaling maximal oxygen uptake to predict cycling time-trial performance in the field: a non-linear approach. *European journal of applied physiology*. *94*(5). 705-710.

Persson. T. L.. Kozlica. H.. Carlsson. N.. & Lambrix. P. (2020). Prediction of tiers in the ranking of ice hockey players. In *International Workshop on Machine Learning and Data Mining for Sports Analytics* (pp. 89-100). Springer. Cham.)

Piri. S. (2020). Missing care: A framework to address the issue of frequent missing values; The case of a clinical decision support system for Parkinson's disease. *Decision Support Systems*. *136*. 113339.

Schetgen, L., Bogaert, M., & Van den Poel, D. (2021). Predicting donation behavior: Acquisition modeling in the nonprofit sector using Facebook data. *Decision Support Systems*, *141*, 113446.

Tingling. P. M (2016). Educated Guesswork: Drafting in the National Hockey League. In Albert J.. Glickman M. E.. Swartz T.B.. and Koning R.H. (Ed.). *Handbook of Statistical Methods and Analyses in Sports* (pp. 327-339). Boca Raton. FL: CRC Press LLC.

Troyanskaya. O.. Cantor. M.. Sherlock. G.. Brown. P.. Hastie. T.. Tibshirani. R.. ... & Altman. R. B. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics*. *17*(6). 520-525.

Vaeyens. R.. Lenoir. M.. Williams. A. M.. & Philippaerts. R. M. (2008). Talent identification and development programmes in sport. *Sports Medicine*. *38*(9). 703-714.

Van Buuren. S.. & Groothuis-Oudshoorn. K. (2010). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*. 1-68.

van Erp. T.. Sanders. D.. & Lamberts. R. P. (2021). Maintaining Power Output with Accumulating Levels of Work Done Is a Key Determinant for Success in Professional Cycling. *Medicine and Science in Sports and Exercise*.

Van Reeth. D. (2016). Globalization in professional road cycling. In *The Economics of Professional Road Cycling* (pp. 165-205). Springer. Cham.

Van Reeth. D. (2019). Forecasting Tour de France TV audiences: A multi-country analysis. *International Journal of Forecasting*. *35*(2). 810-821.

Verbeke. W.. Dejaeger. K.. Martens. D.. Hur. J.. & Baesens. B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European journal of operational research*. *218*(1). 211-229.

Vomfell. L.. Härdle. W. K.. & Lessmann. S. (2018). Improving crime count forecasts using Twitter and taxi data. *Decision Support Systems*. *113*. 73-85.

Weissbock. J. (2015). Draft analytics: unveiling the prospect cohort success model. *Retrieved March*. *17*. 2021.

Appendix

Table A1: Selected Youth Races

Race	Category	Country
Gent-Wevelgem Kattenkoers	Cobbles	Belgium
Ronde van Vlaanderen Beloften	Cobbles	Belgium
Liège - Bastogne - Liège U23	Hilly U23	Belgium
Tour de l'Avenir	Big Tour	France
Coppa della Pace - Trofeo F.lli Anelli	Hilly U23	Italy
G.P. Palio del Recioto	Hilly U23	Italy
Giro Ciclistico d'Italia	Big Tour	Italy
Giro Ciclistico della Valle d'Aosta - Mont Blanc	Big Tour	Italy
Gp Capodarco Comunita Di Capodarco	Hilly U23	Italy
Gran Premio della Liberazione	Hilly U23	Italy
Gran Premio Industrie del Marmo	Hilly U23	Italy
Gran Premio Sportivi di Poggiana	Hilly U23	Italy
Il Piccolo Lombardia	Hilly U23	Italy
Le Triptyque des Monts et Châteaux	Rest	Belgium
Paris-Roubaix Espoirs	Cobbles	France
Ronde de l'Isard	Stage Race Climb	France
Ruota d'Oro - GP Festa del Perdono	Hilly U23	Italy
Tr. Città di S. Vendemiano	Hilly U23	Italy
Trofeo Piva	Hilly U23	Italy
Circuito Belvedere	Hilly U23	Italy
Eschborn Frankfurt U23	Rest	Germany
European Chamionships U23 ITT	ITT	Varying
European Continental Championships U23	Hilly U23	Varying
World Champioships ITT U23	ITT	Varying
World Championships U23	Hilly U23	Varying
Omloop der Vlaamse Gewesten	Cobbles	Belgium
Tour du Valromey	Stage Race Junior	France
Liège la Geize	Stage Race Junior	Belgium
Bernaudeau Junior	One Day Junior	France
Course de la Paix Junior	Stage Race Junior	Czech Republic
GP General Patton	One Day Junior	Luxembourg
Grand Prix Rüebliland	Stage Race Junior	Switzerland
GP dell Arno	One Day Junior	Italy
Keizer der Juniores	Stage Race Junior	Belgium
La Coupe du President de la ville Grudziadz	Stage Race Junior	Poland
Trofeo Karlsberg	Stage Race Junior	Germany
Paris Roubaix Juniors	Cobbles	France
Ronde van Vlaanderen voor junioren	Cobbles	Belgium
Sint-Martinusprijs Kontich	Stage Race Junior	Belgium
Driedaagse van Axel	Stage Race Junior	Netherlands
Tour de l’Abitibi	Stage Race Junior	Canada
Tour du Pays de Vaud	Stage Race Junior	France
Trofeo Buffoni	One Day Junior	Italy
Trofeo Commune di Vertova	One Day Junior	Italy
Trofeo Emilio Paganesi	One Day Junior	Italy

Le Trophee Centre Morhiban	One Day Junior	France
Chrono des Nations Junior	ITT	France
Giro Internazionalle della Lunigiana	Stage Race Junior	Italy
Internationales Junioren Rundfahrt Niedersachsen	Stage Race Junior	Germany
European Chamionships Junior	One Day Junior	Varying
UCI World Championships ITT Junior	ITT	Varying
UCI World Championships Junior	One Day Junior	Varying
Olympia Tour	Rest	Netherlands
Tour d'Alsace	Stage Race Climb	France
Tour de Normandie	Rest	France
Ster ZLM Tour	Rest	Netherlands
Paris-Arras	Rest	France
Paris-Tours U23	Rest	France
Tour de Berlin	Rest	Germany
Tour des Pays de Savoie	Big Tour	France