Universiteit Leiden
The Netherlands

Opleiding Informatica

Training Optimization in Professional Road Cycling through Time Series Prediction and Quantification

Thijs Simons

Supervisors:
Arie-Willem de Leeuw & Arno Knobbe

BACHELOR THESIS

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

6/8/2020

Abstract

In sports, the main challenge for a coach is composing training programs that enable athletes to perform optimally in competition. This is a complicated task, since there are many facets that all have their influence. In this thesis, we use Data Mining techniques to investigate the relationship between training and race performances in professional road cycling. We consider a single cyclist of team Jumbo-Visma, from which we have detailed data about the training sessions and race results over a period of two and a half years. After applying Ordinary Least Square regression to fill up the missing heart rate and power output values, we construct a large variety of features that describe aspects of the training sessions. The performance is quantified by comparing the race results of the cyclist with his competitors. We have applied Subgroup Discovery and found multiple significant subgroups. A few of these subgroups indicate that higher variance in training load positively affects performance. Other subgroups show a positive correlation between training at high altitude and performance. The subgroups found, indicating a relatively bad performance were in all cases linked to a form of overtraining. Although not all subgroups provide a concrete improvement point which the coach can apply to their training program, the subgroups might give the coach new insights in what can be improved about the current training program.

Contents

1 Introduction **4**

2 Background **5**

3 Related work **9**
3.1 Modelling performance with training data 9
3.2 Heart rate and power output predictions 9

4 Data **11**

5 Methods **12**
5.1 Overview 12
5.2 Heart rate model 12
5.3 Power output model 14
5.4 Relative result as the target 15
5.5 Quantification framework 17
5.6 Subgroup discovery 19

6 Results **22**
6.1 Heart rate model 22
6.2 Power output model 23
6.3 Subgroup Discovery 25

7 Discussion **29**
7.1 Heart rate model 29
7.2 Power output model 29
7.3 Subgroup discovery 30

8 Conclusion **33**

References **35**

1 Introduction

In sports, athletes aim for optimal performance during a competition. One of the aspects that influences your performance is the training program prior to a competition. The coach and athlete try to create a training program such that the athlete is in optimal shape. However, optimizing a training program is a complex dynamic and consists of many aspects. Overtraining might result in fatigue and too much rest might cause sub-optimal fitness (Calvert et al., 1976). It is important to stimulate the right systems in your body with right amount of training but the exact way to do this is unknown and also might vary a lot between athletes. There are also other factors that could influence your performance, e.g., bad diet, stress, not enough rest or an injury.

In the past, the coaches used their experience to construct the perfect training program according to them. Nowadays the acquisition of data has opened up new possibilities. For example in road cycling, the advancements in cycling computers, heart rate sensors and power meters make it possible for cyclists to capture large amounts of data of their workout sessions. In cycling, metrics are often measured every second during the whole session, resulting in a very detailed descriptions of a training. The data helps the coach to create and analyze a training program with more precision. However it is often unknown what aspect of your training caused you to perform good or bad due to the many variables affecting your performance and for a human it might be infeasible to find a analytical pattern in this data. Therefore, a Data Mining approach might give insights to the coach that had been overlooked before.

In this thesis, we focus on a Data Mining challenge in road cycling. More accurately, we try to answer the following question: *given the training data of a cyclist, can we find patterns in this data that reflect the performance of a cyclist in competitions?* To answer this question, we will look at the training-period[1] prior to the competitions a cyclist has participated in and compare them with each other. In order to compare training-periods, we have to extract meaningful features from the training-periods. Using an algorithm, we can find the features that have had the most influence on the performance of the cyclist. The found features give us information on improvement points and could potentially be fed back to the coach. Furthermore, the given dataset contains sessions where part of the data is missing. In order to answer our main research question we first need to predict these missing values using a constructed model.

Thesis overview

In this thesis we first discuss the different aspects of a training program, the relationship of attributes *during* training and the Data Mining techniques that were used. We then look at previous work on modeling the relationship between training and performance of athletes. In the chapter "data" we give an overview of the available attributes and what data is missing. Next, we discuss our used methods. More specifically, we define our performance measure and our method of quantifying the raw training data. For the prediction of the missing data we experiment with a new attribute that is the convolution of the power output. Finally we discuss our results, in the case of the Subgroup Discovery, for every subgroup individually we discuss their meaning and consequences.

[1] Training-period: is a period of N days before a stage in which the athlete has trained.

2 Background

Aspects of training

In road cycling there are many aspects that come into play when creating a training program. In this section we provide some information on some basic ideas that are important for this thesis. We start by how training sessions are measured/quantified in general. Furthermore, we talk about different aspects that should be considered when designing a training program.

Internal and external load

The load of a training is an important metric for quantifying a training session and is defined as the intensity $\times$ volume. Load is a way to quantify how "tough" a training session was. The way load is measured, is either internal or external.

Internal load represents the way your body perceived an activity. Heart rate [bpm] is a metric that represents the internal load and is often measured during a training session with a heart rate sensor. Many variables can influence your heart rate, e.g., diet, stress, altitude etc. Thus, heart rate might not precisely represent the internal load. Other measures that might represent interal load more objectively e.g. lactate level, often require you to take a blood sample which takes more effort and is invasive to the athlete's body.

External load is not biased towards body condition and truly represents the amount of energy used in an activity. Power output [Watt] is generally used as a metric for external load and can be recorded during activity with a power meter (Allen and Coggan, 2010). Because power output is an objective metric, the power output has to be corrected by the athlete's capacities. For example, a lightweight athlete can perform better than a heavier athlete although the mean power output is less. When designing a training program using power output, this has to be taken into account.

To capture the load of a training, load metrics have been invented. Load metrics are calculated by inputting measurements like heart rate or power output and outputting a single value. Popular load metrics often use the principle that, higher intensity causes exponentially more load to the body. The way your body responds to different amounts of intensity is of a non-linear fashion, outputting more than a certain amount of watt may use a different system in your body and fatigue will occur more rapidly (Allen and Coggan, 2010). For example, you are only able to sustain your current power output for a short amount of time after exceeding your maximal oxygen uptake threshold (VO2max). Popular load metrics that use this principle are e.g., the TRIMP score based on the heart rate and the TSS score based on the power output. Another more obvious and less complex metric that doesn't use this principle is the total amount of energy spent during a training session. The exact consequences of using different load metrics is sometimes unclear (Erp et al., 2018).

Balancing load and intensity

Road cycling is a sport where both the endurance and sprinting abilities of athletes are important (Mujika and Padilla, 2001). An optimal training thus consists of both sprinting and endurance components. In line with this, different power outputs affect different systems, this also causes your

body to adjust different systems at different power output levels. In order to train in an optimal way, all systems necessary for the given task should be stimulated for the right amount of time. Too much load results in fatigue and thus underperformance. Similarly too little load, results in underperformance because the athletes body is sub-optimally adjusted towards exercise. Hence for optimal performance the training sessions have to be balanced accordingly.

Altitude acclimatization

The altitude at which the training was executed is generally also considered as a means to adjust your body towards better performance. Training at higher altitude stimulates your body to increase the hemoglobin concentration and oxygen buffering capacity. Furthermore, an acute increase in altitude has shown to decrease the performance of an athlete significantly (Faria et al., 2005). The most prestigious grand tours, the Tour de France, the Giro and the Vuelta, all contain stages at high altitude. Therefore, altitude acclimatization is an essential aspect of the training if you want to perform in these competitions.

Tapering

Finally, tapering is generally considered to significantly increase the performance of an athlete (Faria et al., 2005). The most optimal tapering method is unknown and can vary a lot between athletes. Faria (2005) concludes exponential reduction of duration or frequency of training is the superior method but the exact span of time and rate of decay is unknown. Tapering can be explained by Banisters (1976) fitness-fatigue (impulse-response) model. This model states that a training session results in fatigue but also an increase in fitness. Fatigue is experienced more quickly after a training session compared to fitness and if training sessions are continued with right amount of dose, fitness can dominate fatigue after a certain amount of time and ultimately increasing performance (Allen and Coggan, 2010).

Relationship of attributes during training

The heart rate, power output and other variables during a training session are sometimes related to each other. For example, power output is an important contributor to your heart rate. If you produce more power your muscles have to be provided with more oxygen by the cardiovascular system, resulting in a faster heart rate (Ludwig et al., 2016; Grazzi et al., 1999). There are of course also other contributing factors. For example, your heart rate increases after about 10 minutes of exercise without the increase of power output, this phenomenon is called cardiovascular drift (Ekelund, 1967). The amount of revolutions your legs make, called the cadence, also influences the heart rate. It has been shown when cycling at a constant power output heart rate is positively related to cadence (Massey et al., 2011). Altitude also seems to be positively related towards heart rate (Reeves et al., 1987). But also factors out of our scope might influence the heart rate, for instance, diet, stress or lack of sleep. Also heart rate dynamics might differ per athlete.

Regression techniques

Ordinary Least Squares regression (OLS) is an algorithm that constructs a model of the form,

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} ... \beta_p x_{ip} + \beta_{p+1} \tag{1}$$

, y_i is the vector of the to be predicted attribute, x_{ij} represents the vectors of the explanatory attributes, p represents the amount of explanatory attributes and the constants β_j represent the weight with which every attribute contributes to y_i. The OLS algorithm optimizes the constants β_j so that the sum of squares of the differences between the observed and the actual value is minimized.

10-fold cross validation is a means to account for potential overfitting in the models. This method divides a dataset over 10 random subsets of equal size. The algorithm of choice is executed 10 times. For every execution we alternate the test set, the other 9 subsets form the training set. Resulting in 10 potentially different models, the average of the models' scores is used as a quality measure for the used parameters.

R^2 score is a measure to determine the quality of a constructed model. R^2 can also be applied to OLS. R^2 is defined in the following way:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}, \tag{2}$$

SS_{tot} represents the total sum of squares and SS_{res} represents the residual sum of squares. The R^2 score essentially compares the results of your model to the mean predictor. Because of the squaring of the error, larger errors are penalized more severely.

To judge the significance of the resulting models, the F-test can be used,

$$F = \frac{R^2}{1 - R^2} \cdot \frac{n - p}{p - 1}, \tag{3}$$

R^2 is the explained variance (Eq. 2), n represent the number of datapoints and p represent the number of parameters that were used in the model. With the F-test we can derive a p-value and determine the probability of the dependent and independent variables being correlated by chance.

Subgroup Discovery

Subgroup discovery is a Data Mining method for finding a subset in your data that significantly deviates from the full dataset. A subset is defined using a logical condition. For example, a subgroup's definition might be: "$attribute\_x > y$". Consequently the members of this subgroup are all the datapoints of the full dataset where this condition results to true. The condition with which a subgroup is defined can provide useful information.

Due to the nature of the results of the subgroup discovery, we find thresholds instead of an exact model. The advantage of this is that we can do more of a exploratory analysis on multiple aspects of our data without specifying an exact relationship. The disadvantage of this, depending on what

features we extract from the data is that the results can have multiple interpretations and this still leaves us with missing information. For example, a feature involving the standard deviation can be constructed with different configurations. In that case further analysis has to be performed.

The software package we used for subgroup discovery is called Cortana (Meeng and Knobbe, 2011). Cortana implements an algorithm which tries to find interesting subgroups. The interestingness of a subgroup is determined by a quality measure. Cortana is very versatile in that it supports both numeric and nominal values and implements multiple quality measures suitable for different problems. One of the available quality measures is the z-score (Pieters et al., 2010),

$$\phi(S) = \sqrt{|S|}\frac{\mu_S - \mu_0}{\sigma_0}, \tag{4}$$

μ_s and μ_0 are the mean of the subgroup and the full dataset, respectively. S denotes the subgroup and σ_0 is the standard deviation of the full dataset. The z-score essentially compares the mean of the subgroup with the mean of the full dataset and normalizes it by the standard deviation of the full population. The normalization by the standard deviation of the full population takes into account that a difference between the means of the subgroup and the full dataset should be weighted more if the standard deviation of the full subset is smaller. Another quality measure we used, based on a nominal target, is the Absolute Weighted Relative Accuracy (AbsWRAcc),

$$WRAcc(S,T) = p(ST) - p(S)p(T), \tag{5}$$

where T represents the target, S represents the subgroup and p represents the probability of a provided class. WRAcc is based on the target coverage of the subgroup compensated by the expectedness of finding this subgroup.

When applying Subgroup Discovery some of the found subgroups msy not be significant and occur by chance. If the dataset contains relative many attributes compared to the number of examples, the subgroups found by chance will increase. To account for this phenomenon only subgroups with a quality above a certain threshold are accepted. The threshold represents a level of significance, we can calculate this threshold with a technique called swap-randomization. This technique shuffles the values of the target column to create an incoherence between the target and independent variables. The subgroup discovery algorithm is executed on this shuffled table with the exact same settings as in our experiments. This shuffle and execution are done 100 times and with the results we can calculate an accurate threshold for a certain significance level (Duivesteijn and Knobbe, 2011).

3 Related work

3.1 Modelling performance with training data

The relation between training and performance has been studied many times before (Calvert et al., 1976; Clarke and Skiba, 2013). Calvert and Banister (1976) used a custom load metric designed for swimming in combination with the impulse-response (IR) model to model the relationship between the training and the performance of a swimmer. They conclude that the IR model is a successful model for describing this relationship. Many modifications to the model have been made since. The impulse-response method is generally considered the most superior method currently available to capture the relationship between performance and training load (Clarke and Skiba, 2013). The downside of the IR model is that training data over a long period of time and very precise performance tests need to be available to determine the parameters of the model (Jobson et al., 2009). In regards to road cycling, due to differences in stage profiles and the ability to drift, it is more problematic to get an objective performance metric compared to, for example, swimming where the "stage profile" is always the same and the performance is individual.

Knobbe (2017) extracted features from the training data of 4 athletes to obtain quantified data of the training program of these athletes. Using a subgroup discovery algorithm the features were then analysed for patterns. Significant results were found of which some were used to optimize the training program. This also shows a relative simple model (compared to the IR model) can be successful in modeling the relationship between the training and performance of an athlete.

3.2 Heart rate and power output predictions

Often attributes of sessions are incomplete due to e.g., broken sensors. As a solution to this problem predictive models could fill up this missing data. For example, the heart rate has been modeled before, both analytically and in a machine learning fashion.
Bunc et al. (1988) show that the heart rate response to exercise can be described with a backwards looking exponential kernel applied to the power output: $HR = a - b * e^{ct}$. The experiments were however performed on an ergo-meter in a controlled environment and only at constant power output. The dynamics of swift change in power and endurance cycling, which you find during a training session, are not covered.
Hilmkil et al. (2018) try to predict a full training sessions. A black box machine learning method called Long Short-Term Memory networks was used. This is a recurrent neural network which takes the moments before the moment you would like to predict into account. The main disadvantage of this method is that the model obtained is not interpretable by humans.
Ludwig et al. (2016) propose a model involving an exponential convolution applied to the power output and tries to optimize the parameters of the model. The overarching aspects in these works is that heart rate at a certain moment is largely influenced by the amount of exercise performed in a short period leading up to that moment.

Apart from the heart rate, predicting the power output of a cyclist has also been done before. Martin (1998) proposes a very accurate mathematical model. Although the model accurately describes the output, most of the variables used by the model are very impractical to obtain during a normal

outdoor training session. For example, wind resistance is hard to obtain because this is dependent on, if you are drafting.

4 Data

Session data

We used data of single cyclist of team Jumbo-Visma. The data contains around 800 sessions and spans a period of 2 years and 6 months in which this cyclist has competed in around 160 races. The individual sessions can both be training data as well as data of competitive races. We make no distinction between the two which essentially means we treat a race as part of the training-period, this is often the case in Grand Tours. The raw data is of a time series format with a resolution of 1 sec.

A value for the following attributes is available for every second into a session:

- Power [Watt]
- Cadence [rpm]
- Heart rate [bpm]
- Distance cycled [km]
- Duration cycled [sec]
- Speed [km/h]
- Altitude [m]
- GPS coordinates [Longitude, Latitude]

Part of the heart rate and power data is missing, this can be due to sensors which are not functioning or in case of the heart rate sensor, it is often removed due to discomfort. We are missing 9.8% and 6.7% of the heart rate and power output data respectively.

Stage results data

The cyclist being studied has cycled in about 160 stages during the period the available data spans. These stages were part of Grand Tours but also smaller tours of about a week. We use the results of theses stages as our performance measure and thus our target. The data originates from procyclingstats.com.

The retrieved data is then processed to get a comparable target for the cyclist. This is done by comparing the results of the cyclists with their direct components. This will be further explained in section 5.4.

5 Methods

5.1 Overview

To answer our research question we have designed a pipeline. A visual representation of the pipeline can be found in Figure 1. Here, we briefly discuss the main steps of the pipeline, in the following sections the individual parts will be discussed in more detail.

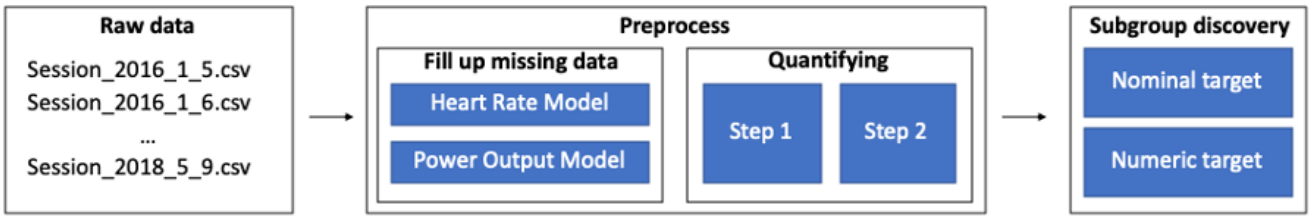

Figure 1: Pipeline of the research, inputting raw session files, outputting subgroups.

Fill up missing data
Since a substantial part of our heart rate and power output data is missing we constructed models for each of these attributes. We use a linear regression approach in order to construct a model for both heart rate and power output. The models predict heart rate and power output values during a training session and are used to fill up the missing data of our dataset.

Quantifying
To analyze the data, we process the raw session data so that we obtain a conventional table of rows representing a single stage with information about the training-period and columns representing a signle feature of the training-period. One of the attributes of the table is the target, in our case the relative time (Section 5.4) of a stage. The other attributes are the constructed features and they describe the training-period before a stage. To generate these features we propose a framework. The framework consists of two quantification steps. The first step quantifies a training session's time series. The second step quantifies over the results of the first step. The resulting table can then be analyzed by the subgroup discovery algorithm.

Subgroup Discovery
In this step we apply a subgroup discovery algorithm to the dataset we constructed in the quantification step. We experiment with both a nominal target and a numeric target.

5.2 Heart rate model

Predicting the heart rate presents us with a regression task. Out of the available attributes we have chosen the following:

- Riding Time [sec]
- Pedal Power [W]
- Cadence [rpm]

- Distance [km]
- Altitude [m]

In the most simple model, the *baseline*, we only use the untouched attributes. Moreover, we also consider another model that includes the attribute "Pedal Power Convoluted [W]", this attribute captures the period before the moment we want to predict by means of convolution. The definition of the attribute is as follows,

$$p_t^s = \{p_0, p_1, ..., p_l\},$$

where p is the pedal power, l is the length of the session and s is the session selector.

$$C(t,s) = \sum_{i=0}^{w} h(-i) * p_{t-i}^s \ , \tag{6}$$

$$h(x) = e^{x/\tau}, \tag{7}$$

$C(t,s)$ defines the convoluted time series and w is the length of convolution window. $h(x)$ is the kernel that was used, in our case a backwards looking exponential kernel. A visualization of the kernel can be found in Figure 2. A higher τ makes the period summarized longer, a lower τ shorter. Moments further away from the moment t contribute less towards the convoluted value and consequently, moments closer to the moment t contribute more towards the convoluted value. The kernel is inspired by Bunc (1988) and Ludwig (2016) who showed that a backwards looking exponential kernel can successfully mimic the response of a body to load in a controlled environment.

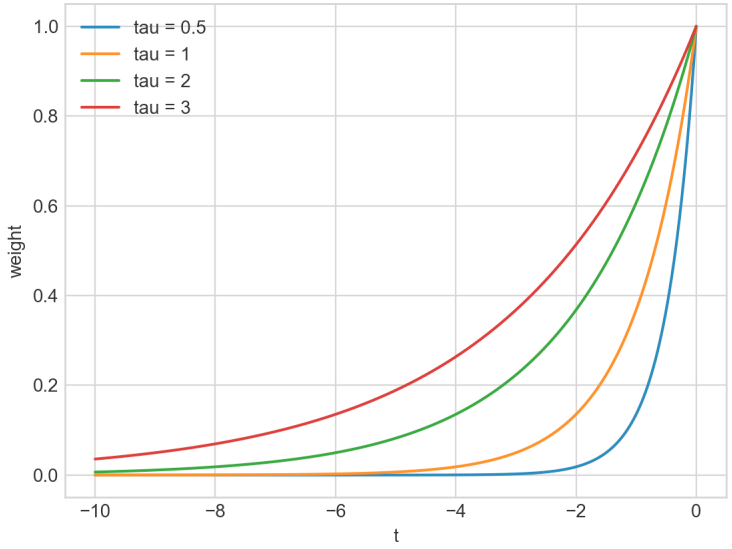

Figure 2: Plot of the kernel (Eq. 7). $t = 0$ is the moment to be predicted, a lower t is weighted exponentially less towards the convoluted value. The τ values for the red, black, blue and purple line are 0.5, 1, 2 and 3 respectively. A higher τ makes the period summarized longer, a lower τ shorter.

We used linear OLS regression to build the predictive model. We first applied OLS to the time series without the convoluted power output to get a baseline model. We then applied OLS to the

time series with the convoluted power output and optimized the parameters w and τ. For every w in 30, 50, 100, 200 we find the optimal τ. A shorter window length is preferred because it makes the model depend on fewer variables and allows us to predict more values closer to the beginning of a session. For example, a window length of 100 would allow us to start predicting values from t=101 on-wards.

The final model that was obtained is able to predict the time series of the heart rate attribute in full detail, meaning in our case we can predict a value for the heart rate for every second during a session provided that the attributes that the model is dependent on are available. However, only aggregated values of our heart rate are used in the subgroup discovery analysis. Thus we have to evaluate the accuracy of the constructed model on these aggregated values by determining the R^2 of the predicted aggregates. Therefore, we first calculated all the actual aggregates for every individual training session of which there is no data missing. We then predicted the aggregates using our constructed model. We could then compare them with the actual aggregates and determine the corresponding R^2.

5.3 Power output model

To construct the power output model we take a similar approach as with the heart rate model. The main difference is that we predict the convoluted value of the power output at a certain moment t and then apply deconvolution to this convoluted value to obtain our watts. The formula for the deconvolution can be obtained in the following way:

$$w_p = \{w_0, w_1, ..., w_l - 1\}, \qquad v_t = \{v_0, v_1, ..., v_m - 1\ \},$$

w is our *convolution* window and contains the weights, the window can be constructed with $h(x)$ (Eq. 7) and l is the length of our window. Because of the exponential kernel we use for our window, $w_p > w_{p+1}$ and $w_0 = 1$ (Figure 2), v contains the convoluted power output, m is the length of the time series. Assuming the function $D(t)$ represents the deconvoluted value at moment t (that is the normal power output measured by the sensor during a cycling session). To derive $D(t)$, we rewrite the formula of the convolution from Eq. 6 to include $D(t)$,

$$v_t = \sum_{i=0}^{l-1} w_i * D(t-i),$$

v_t is our convoluted value and can be easily obtained from the power output values ($D(t)$) and the convolution window. We are interested in $D(t)$ because this is our deconvoluted value hence we extract $D(t)$ from the summation,

$$v_t = w_0 * D(t) + \sum_{i=1}^{l-1} w_i * D(t-i), \qquad w_0 = 1,$$

and invert the formula to get the definition of $D(t)$.

$$D(t) = \begin{cases} v_t - \sum_{i=1}^{l-1} w_i * D(t-i), & t \geq 0 \\ 0, & t < 0 \end{cases}, \tag{8}$$

This leaves us with a recursive definition because $D(t)$ is dependent on itself. Also note, depending on the window size, values too close to beginning of the time series will result in very inaccurate results for two reasons. First of all $D(t) = v_t$ for values too close to the beginning of the time series. Secondly there is no training data available for the convoluted values at the beginning of a training session because no such data exists. To test for any potential error in the deconvolution method we compare the R^2 score of the deconvolution model to the model that predicts the convoluted values of the power output.

For the regression the same attributes are used as in the heart rate model except for the cadence, the reason for this is that in reality, often when the power output is missing, the cadence is also missing because it is recorded by the same sensor.

In summary these are the steps we take:

1. Apply OLS to the data and find optimal values for w and τ. The resulting model is capable of predicting the convoluted power output for a given moment t.
2. Predict convoluted power output using our constructed model on our dataset.
3. Deconvolute the predicted convoluted power output.
4. Apply a rolling average to the deconvoluted power output, this results in the usable values.
5. Test accuracy of the model after deconvolution is applied.
6. Test accuracy of model in predicting aggregates for individual training sessions.

5.4 Relative result as the target

The quality of every training-period before a stage is judged by the result of that stage. In order to judge the performance we determine the time of our cyclist and compare this time to the top 10 in the General Classification (GC) of the corresponding competition after the final stage. We call this measure of performance the relative time. If it is a competition which only consists of one stage, we compare the results of the cyclist with the 10 fastest cyclists of that stage. Applying this method, we often obtain a result of 0 (Figure 3), this happens in the case of a mass finish. Results higher than 300, meaning you lost more than 300 seconds compared to the top 10 in the GC, are clipped to 300. The 300 threshold is fairly arbitrary, but was put in place so that outliers do not skew the average towards the right. Also the outliers often happen because the cyclist broke down due to, for example, illness or falling of the bike. This might not accurately represent the relation between training and performance.

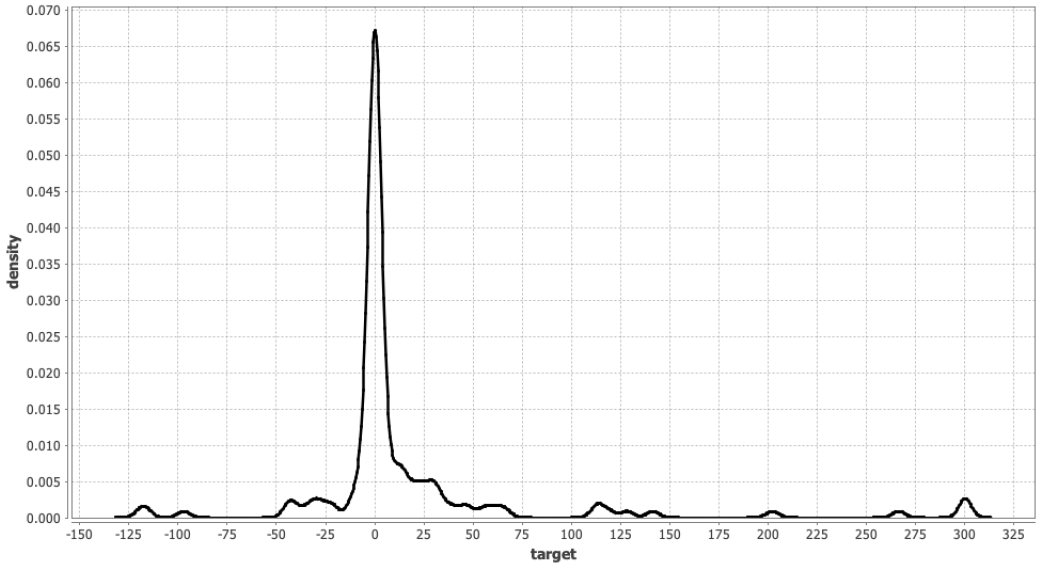

Figure 3: Distribution of the target variable of our cyclist. Note the bump at 300 because of the clipping that was applied. y-axis values are probabilities.

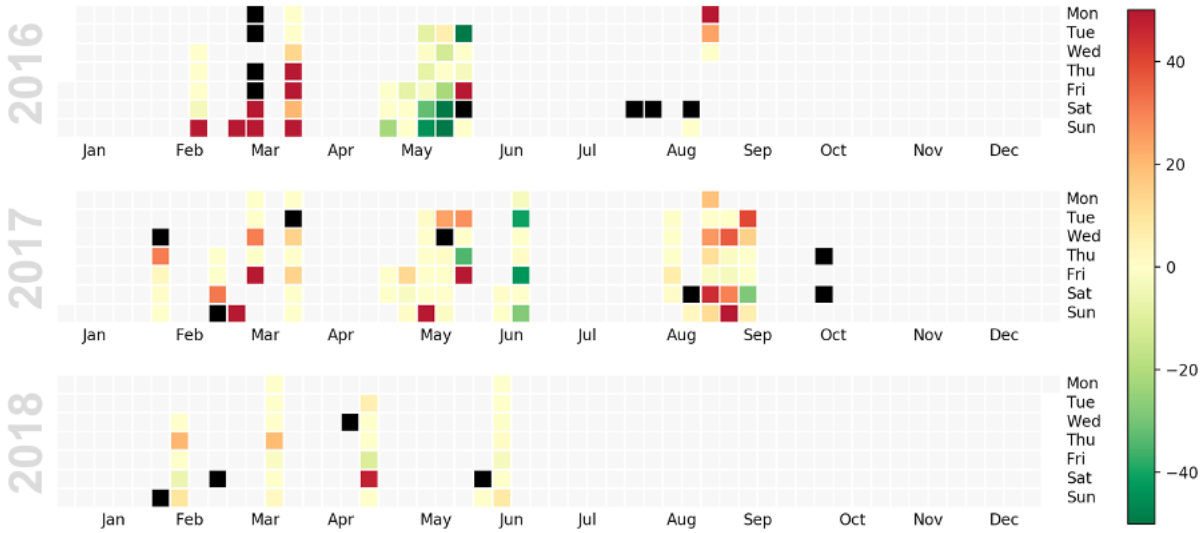

Figure 4: Overview of performances of our cyclist. Color represents relative results in seconds, black means result was unusable, results are clipped in the calendar between -50 and 50 for the purpose of visualization. Green is time gain compared to top 10 in GC, red is time loss compared to top 10 in GC.

For the concerned cyclist we calculated the relative time. An overview of this can be found in Figure 4. All the days marked black indicate that the result of the stage that day, was not used in our research. In total 141 of 160 races were used. Stages were not used if the stage was a team time trial, this variant is very different from mass starts and individual time trials because it is much more a team result and for our research we need a quality measure for the individual cyclists. Stages where the cyclist did not start or did not finish were also not being taken into account

because the relative time could not be calculated.

Apart from the numeric relative result shown in Figure 3, we also created a nominal version of the relative result. Stages with a negative relative time are classified as "WIN", stages with a negative time are classified as "LOSE" and stages with a relative time of 0 are classified as "NEUTRAL". By introducing the nominal target you are essentially only focused on winning or losing time compared to your direct opponents or in the case of a relative result of 0, you are neither losing nor winning.

5.5 Quantification framework

In this paper, we propose a framework in which we can process our raw time series into features that describe a training period. To obtain these features, two iterations of quantification are applied to the time series data. The first quantification results in a description of an individual training. The second iteration of quantification aggregates over the results of the first quantification, resulting in a description of a training-period.

Part 1

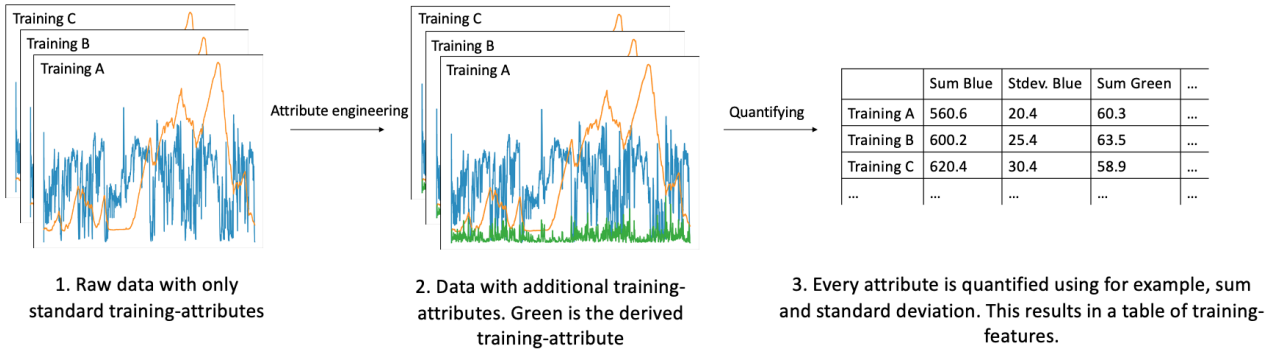

Figure 5: Overview of part 1 of the quantification. In part 1 the individual sessions are quantified.

A single training is built up of multiple training-attributes. For example, heart rate, power output or duration are training-attributes. During every second of the training, a value is known for every training-attribute. New training-attributes can be derived by for example, taking the rolling average of an existing training-attribute. A training is then quantified using an attribute-aggregate. An attribute-aggregate function inputs a training-attribute in the form of a time series and outputs a single value. For every training-attribute we specify what aggregate functions must be applied because not every aggregate might make sense for every training-attribute. An aggregate function might for instance calculate your training load or volume. After the aggregate functions are applied to all training sessions, we obtain a table with training-features. This table is essentially representing multiple aspects of every training session. The training-features are further processed in part 2 of the quantification.

Part 2

In this part we use the obtained values from part 1 of the quantification to create descriptive features of the training-period prior to every competition. Figure 6 shows an overview of part 2 of the quantification.

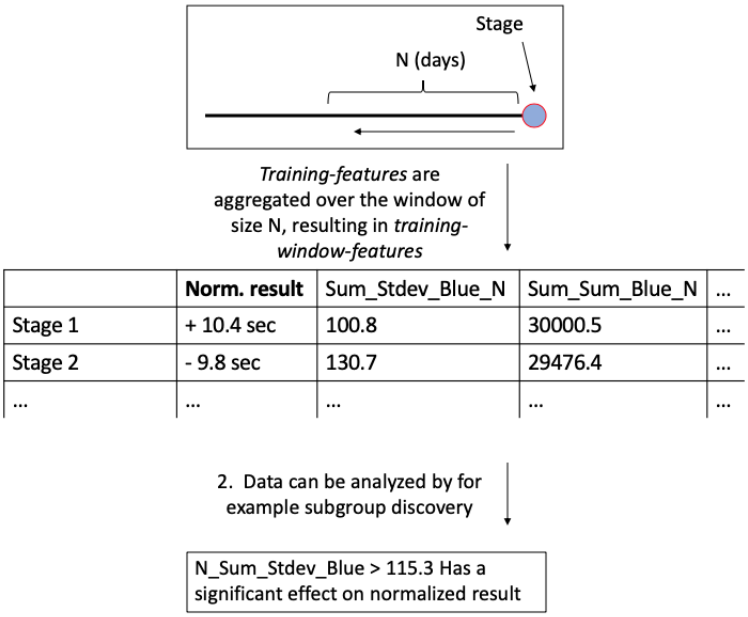

Figure 6: Overview of part 2 of the quantification. In this part of the quantification we describe a period of training by processing the individual quantified training sessions.

A training-window is a period of N days prior to a stage up until the day before a stage. Multiple window sizes can be created for a single stage in order to study the various time spans before a stage. For every training-window we calculate a set of training-window-features. First we specify which window-aggregates must be applied to which training-features. We then obtain the training-window-features by applying these aggregates on the training-features that fall within the scope of the window. As a result we obtain a table that contains information of stages and its corresponding training-period.

In Figure 7 you can find an example of a constructed feature. The first number of the string, in the example a "4", is the size of the window, "max" is the window-aggregate and "avg" is the attribute-aggregate. This particular string can be described as follows: What was your highest average heart rate within a four day period before a stage.

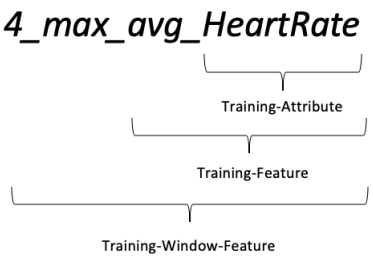

Figure 7: Stripped down example of a final training-window-feature.

For simplicity, in case every window-aggregate is applied to every training-feature, the amount of training-window-features can be calculated in the following way:

$$\#\_features = \#\_windows \times \#\_window\_aggregates \times \#\_training\_features$$

The number of training-features is dependent on the number of aggregates that are applied to the training-attributes. Note that the total number of features in the final table can increase very fast.

5.6 Subgroup discovery

Subgroup Discovery was the technique used in this study to find a pattern in our data. To use the Cortana Subgroup Discovery algorithm our data first has to be transformed in to a conventional table, to achieve this we use the proposed quantification framework. After the quantification we discuss the exact settings that were used with the Subgroup Discovery algorithm.

Quantification

The quantification framework requires us to specify how the data should be processed. We start by specifying the way the individual training sessions should be processed. The training-aggregates that were applied to every training-attribute can be found in Table 1.

Table 1: Training-attributes and the aggregates that were applied to the training-attributes.

Training-attribute	**Applied aggregate functions**
Power Output 30-s Rolling Average	sum, 85th percentile 85, 95th percentile
Heart Rate	avg, std, 25th percentile, 50th percentile, 75th percentile, 85th percentile, 95th percentile
Constant 1	sum
Altitude	avg

A 30 second rolling average version of the power output is used instead of the regular power output for several reasons. First of all, a rolling average of 30 seconds more closely represents your bodies response towards exercise, this is inspired by Coggan's (2010) TSS load metric. Secondly, the rolling average removes possible outliers that can occur when large amounts of force are exerted on the pedals for a small amount of time. Finally, it makes it easier to predict the power output. Because the model uses a deconvolution approach, errors in the prediction can also cause the deconvoluted

values to be highly fluctuating, by applying a smoothing we solve this problem.

It is not strictly necessary to explain the reasons why these aggregates were applied to the training-attributes because this introduces a exploratory effect to the analysis. However, for some configurations, we can explain what the quantified result represents.
The sum aggregate applied to the power output represents the amount of energy used in the training session and is our main load metric. In this study we have decided to use as our main load metric, the total energy used during a session for reasons that are mainly practical, but there are also strong indications that the relationship between total energy used and metrics such as TSS and TRIMP are highly correlated making our decision less important (Erp et al., 2018). Additionaly, the exploratory aspect of our Data Mining method makes that if significant results are found for total energy spent as a load metric, further analysis could be done with different load metrics to finetune the results.
The sum applied to the training-attribute "Constant 1" returns the duration of the training session and represents the training volume.
The average applied to the altitude is used to research altitude acclimatization.
The average of the heart rate is a representation of the intensity of the training session. The standard deviation of the heart rate gives an indication of variation of the training session. The percentiles are applied to the power output and heart rate in order to get a depiction of the distribution of the training session. The percentiles 25, 50 and 75 were not applied to the power output because in our experiment we determined they could not be predicted accurately enough.

For the second step of the quantification we require some more specification, especially, about the period prior to a stage. The window-aggregates and to what training-features they are applied to can be found in Table 2. The sum aggregate captures the volume of training. The standard deviation captures the variety of sessions during a period. The max aggregate can capture a day with above average exercise, indicating either overtraining or positive adaption to high load depending on a positive or negative average of the subgroup's target. The min aggregate is capable to capture the a rest day if the value is low, if the value is relatively high it shows that during that period the training intensity was high. Finally, the average gives an indication of the intensity of training over a period.

Table 2: Window-aggregates and to what type of training-features they will be applied to.

Window-aggregates	**Applied to training-attributes with**
sum	sum
std	sum, avg, 25th percentile, 50th percentile, 75th percentile, 85th percentile, 95th percentile
max	sum, avg
min	sum, avg
avg	sum, avg, std, 25th percentile, 50th percentile, 75th percentile, 85th percentile, 95th percentile

The window sizes over which we aggregate are: 1, 4, 7, 14, 21 and 28 days. This gives us a wide variety of training-periods, with the emphasis on the first week before a competition. In the end

we obtained 167 features, taken into account that not all window aggregates make sense for the window size of 1. For example, the standard deviation of a period with a single training session is always 0 or the minimum of a period with a single training session is the same as the maximum.

Settings

The quality measure we used to score a subgroup for the *nominal* target is absWRAcc. We try to find a high quality subgroup of losing stages or, because we use the absolute variant of WRAcc, we try to find a complement subgroup consisting of mostly winning or neutral stages.

Compared to the nominal target, for the *numeric* target we weight the values of the relative results in determining the quality of a subgroup. We use both the inverse z-score and the z-score as a quality measure because we are interested in both good and bad performance. Subgroups of bad performance indicate what you should avoid in your training and subgroups of good performance indicate what aspects of training you should emphasize, both ultimately increasing performance.

For all targets and quality measures we use the same parameters in Cortana. We only consider subgroups with a size between 10% and 90% of the entire data set to ignore the more insignificant distinguishes in the dataset. The search strategy is a beam search with a width of 100, meaning only the best 100 candidates are used to find additional conditions. Since we only search at a depth of 1 a search width of 100 means only the best 100 subgroups are returned. For the numeric strategy we chose the "best" option. The numeric strategy determines where in the range of a particular attribute it is split to form a condition. In the case of best, no potential splits resulting in higher quality are skipped. We used swap-randomization to compute a threshold for a significance level of 95% and subgroups below this threshold are discarded.

6 Results

6.1 Heart rate model

We first applied OLS to a set of untouched attributes to get a *baseline*. The baseline model was then used to measure the potential improvement of our second model, the model with convolution applied to the power output attribute. Note, all results are +/- std whenever 10-fold cross validation was applied. Using these models we were able to predict 87% of the missing heart rate data. Not all data could be predicted because sometimes both the heart rate and the data the predictive model is dependent on was missing.

The baseline experiment resulted in a R^2 score of 0.39 +/- 0.00. We then applied OLS to the data with the additional convoluted attribute. The results of the experiment where convolution was applied to the power output attribute, can be found in table 3. The R^2 of $w = 50, 100, 200$ only differ 0.01 at maximum, $w = 30$ differs 0.04 at maximum. We decided to use a window size of 50 in our final model because this uses significantly fewer variables compared to $w = 100$ or $w = 200$. The model with $w = 50$ and $\tau = 37$ shows an increase in R^2 of 0.39. Because the model with convolution performed better compared to the baseline model we only show the coefficients of the model with convolution, these can be found in Table 4. The coefficients are normalized by the standard deviation.

Table 3: The optimal parameters we found for our model *with* convolution.

w	τ	R^2 ***+/- std***
30	39	*0.75 +/- 0.00*
50	*37*	*0.78 +/- 0.00*
100	*32*	*0.79 +/- 0.00*
200	*33*	*0.79 +/- 0.00*

Table 4: Normalized coefficients of the heart rate model with convolution. Note, the intercept is not normalized.

Riding Time [sec]	***Pedal Power [W]***	***Pedal Power Conv. [W]***	***Cadence [rpm]***	***Distance [km]***	***Altitude [m]***	***Intercept***
-0.245 +/- 0.00	*-0.058 +/- 0.00*	*0.893 +/- 0.00*	*-0.025 +/- 0.00*	*0.397 +/- 0.00*	*0.044 +/- 0.00*	*72.39 +/- 0.00*

Note that all the standard deviations are 0.00, this means the different models generated by the 10-fold cross validated were all very similar. Consequently, all the 10 splits in the dataset were saturated with a good variation of datapoints.

The R^2 score of our model on the individual training session aggregates can be found in Table 5. Some of the model's scores on predicting aggregates are significantly lower compared to the R^2 score in Table 3. The model performed similarly on Std, 85th Per and 95th Per. The lower percentile aggregates scored lower compared to the higher percentile aggregates. The score of Avg was surprisingly low.

Table 5: R^2 scores of the model on the individual training sessions $\tau = 37, w = 50$.

Avg	***Std***	***25th Per***	***50th Per***	***75th Per***	***85th Per***	***95th Per***
0.59	*0.75*	*0.53*	*0.58*	*0.68*	*0.70*	*0.74*

For all regression models that predict the heart rate we can calculate the R^2 threshold with a confidence interval of 99%, using the F-test (Eq. 3). H_0: There is no correlation between the dependent variables and the heart rate. For every found R^2 above a certain threshold we can reject the null hypotheses. For a 99% confidence interval with $n = 6198913$ and $p = 6$, we obtain a R^2 threshold of $2.457 \cdot 10^{-6}$. Thus we can reject the H_0 and conclude our heart rate models are statistically significant. Note, for the model in Table 4, $p = 7$ thus even a lower R^2 threshold is found.

6.2 Power output model

Similar to the heart rate model we first constructed a *baseline* model by applying OLS to a set of untouched attributes. The baseline model was then used to measure the potential improvement of our second model *with* convolution. The baseline model predicting the 30-s rolling average of the power output resulted in a score of: $R^2 = 0.61$ +/- 0.0. In the case when both the power output and the data the predictive model is dependent on are missing we could not predict all missing values. For that reason we were able to predict 76% of the missing heart rate data instead of the full 100%.

The parameters and the corresponding R^2 of the model predicting the convoluted value of power output can be found in Table 6. For the same reasons as for the heart rate we chose to use the parameters $w = 50$ and $\tau = 37$ because a lower w makes the model dependent on fewer variables. The normalized coefficients of this model can be found in Figure 7.

Table 6: Model for the prediction of the *convoluted* value of power output.

w	τ	R^2 ***+/- std***
30	43	*0.73 +/- 0.00*
50	*37*	*0.76 +/- 0.00*
100	*38*	*0.77 +/- 0.00*
200	*35*	*0.78 +/- 0.00*

Using the model in Table 7 we predicted the 30-s rolling average of the power output by applying a deconvolution to the result of the model. This resulted in a score of $R^2 = 0.75$, only a decrease of 0.01 compared to the R^2 score prior to the deconvolution (Table 6). Compared to the baseline model the R^2 score has largely increased by 0.14, but this is a lower increase compared to the heart rate model.

Table 7: The found coefficients of the baseline that tries to predict the 30-s rolling average of the power output. Note, the intercept is not normalized. $w = 50, \tau = 37$.

Riding Time [sec]	***Heart Rate [W]***	***Distance [km]***	***Altitude [m]***	***Intercept***
0.156 +/- 0.00	*0.904 +/- 0.00*	*-0.280 +/- 0.00*	*-0.020 +/- 0.00*	*-6643.433 +/- 0.00*

Similar to the heart rate model we evaluated the found model for its predictive capacities on the individual training session aggregates. Results of this test can be found in Table 8.

Table 8: R^2 scores of the model on the individual training sessions $\tau = 37, w = 50$.

Avg	***Sum***	***Std***	***25th Per***	***50th Per***	***75th Per***	***85th Per***	***95th Per***
0.16	*0.93*	*0.29*	*0.17*	*0.31*	*0.46*	*0.54*	*0.67*

The model shows a significant drop in R^2 on all aggregates except for the sum aggregate, which shows an increase in R^2. Similarly, to the heart rate, the lower percentile aggregates scored lower compared to the higher percentile aggregates.

For all regression models that predict the power output we can calculate the R^2 threshold with a confidence interval of 99% using the F-test (see Eq. 3). H_0: There is no correlation between the dependent variables and the power output. For every found R^2 above a certain threshold we can reject the null hypotheses. For a 99% confidence interval with $n = 6198913$ and $p = 5$ we obtain a R^2 threshold of $2.687 \cdot 10^{-6}$. Thus we can reject the H_0 and conclude our power output models are statistically significant.

6.3 Subgroup Discovery

We applied a subgroup discovery algorithm to the dataset we obtained by applying the proposed framework. The dataset contains a nominal variant and a numeric variant of the target. For both variants we applied subgroup discovery in a slightly different way.

Nominal target

For the nominal target the subgroup discovery was run with the specified parameters and absWRAcc as the quality measure. The results above the significance threshold of 95% can be found in Table 9.

Table 9: The results of the subgroup discovery on the nominal target with the AbsWRAcc quality measure. Threshold value corresponding to the 95% significance level: 0.0779.

Nr.	*Coverage*	*AbsWRAcc*	*Probability*	*Losing*	*Winning*	*Neutral*	*Conditions*
1.1	*61*	*0.0836*	*0.2*	*12*	*22*	*27*	*14_std_sum_PedalPowerRolling[W] >= 1362154.8*
1.2	*52*	*0.0800*	*0.17*	*9*	*19*	*24*	*21_std_sum_PedalPowerRolling[W] >= 1402560.1*
1.3	*75*	*0.0798*	*0.24*	*18*	*23*	*30*	*28_std_sum_PedalPowerRolling[W] >= 1354601.4*

Three similar subgroups are found. All 3 subgroups use the same aggregates on the same training-attribute, the only difference is the length of the window. A higher standard deviation in your training volume has had a significantly lower probability of losing time compared to your opponents. More specifically, compared to the entire dataset the probability of losing time dropped from 0.39% to 0.2% when looking at the highest scoring subgroup (nr. 1.1). Because the 3 subgroups are largely overlapping we only visualized the most significant one in Figure 8.

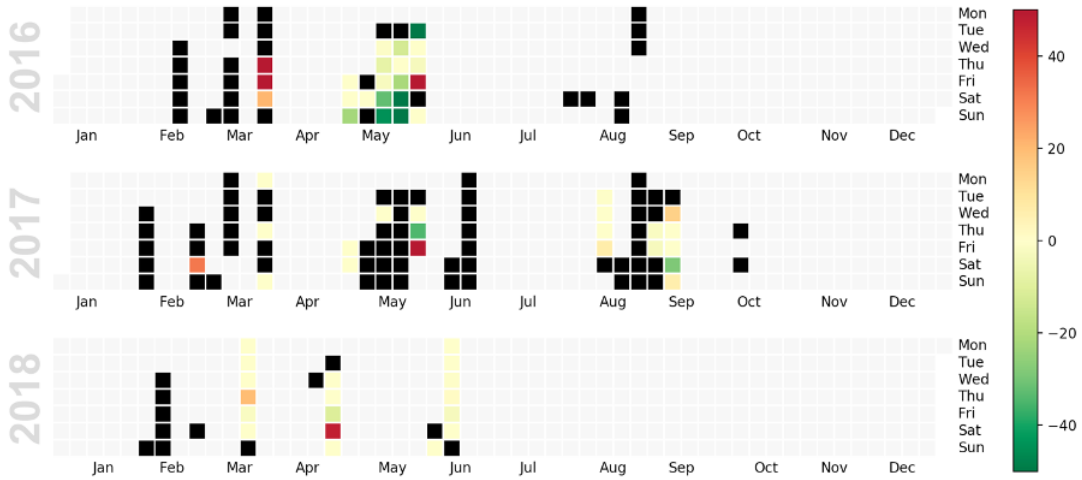

Figure 8: Subgroup nr. 1.1 14_std_sum_pedalPowerRolling[W] >= 1362154.8, black are all stages that are not in the subgroup or not used. Relative results are clipped to -50 and 50 for visualization purposes.

Numeric target

The results above the significance threshold of 95% for the z-score as the quality measure can be found in Table 10.

Table 10: Found subgroups for quality measure z-score. Threshold corresponding to the 95% significance level: 4.21.

Nr.	*Coverage*	*z-score*	*Average*	*St. Dev.*	*Conditions*
2.1	*16*	*4.89*	*88.32*	*113.04*	*28_min_avg_HeartRate[bpm] >= 100.05824*
2.2	*20*	*4.74*	*78.64*	*117.01*	*4_avg_per95_PedalPowerRolling[W] >= 380.88977*

Due to the use of the z-score we only found subgroups with a positive average relative result meaning the cyclist has lost time compared to its opponents. Thus the conditions of the subgroups that were found could tell us what potentially caused the cyclist to perform badly. As a comparison, the average relative time on the numeric target is 15.17 seconds.
The subgroup sizes are a lot smaller compared to the nominal target and less spread across the observed period of time, you can see this well in the visualization of the subgroups (Figures 9 and 10) Looking at subgroup nr. 2.1 (Figure 9), all but one of the subgroups occurred in the beginning of the 2016 season. Subgroup nr. 2.2 (Figure 10) is more spread over the observed period, although there is only 1 stage in 2018 and all of the stages except 3 occur in the beginning of the season.

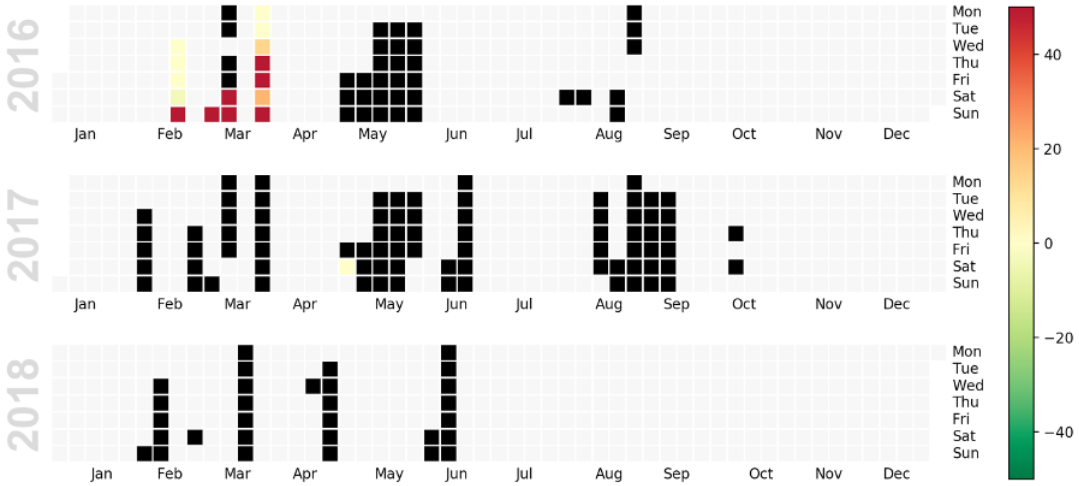

Figure 9: Subgroup nr. 2.1 28_min_avg_HeartRate[bpm] >= 100.05824, black are all stages that are not in the subgroup or not used. Relative results are clipped to -50 and 50 for visualization purposes. Note, all but one of the stages in the subgroup are in the beginning of the 2016 season.

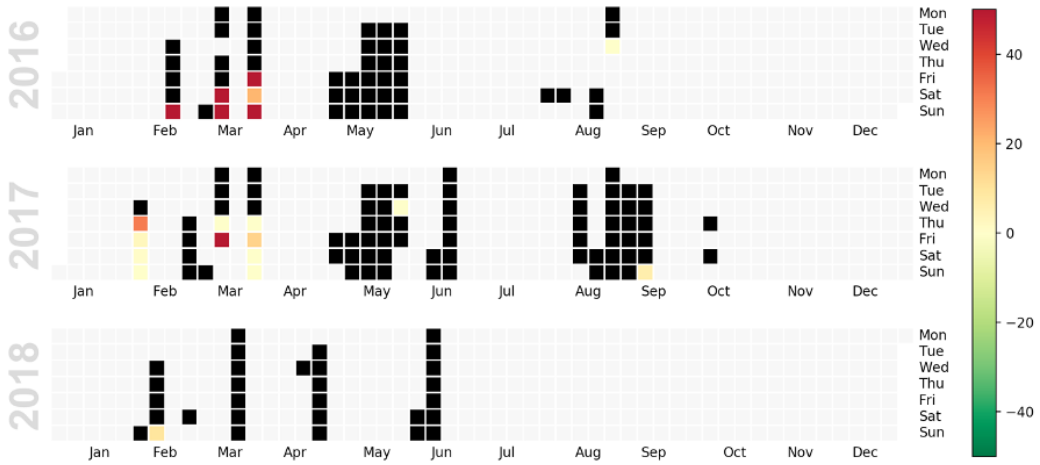

Figure 10: Subgroup nr. 2.2 4_avg_per95_PedalPowerRolling[W] >= 380.88977. Black are all stages that are not in the subgroup or not used. Relative results are clipped to -50 and 50 for visualization purposes. Note, most of the members of the subgroup appear at the end of a competition

The results above the significance threshold of 95% for the inverse z-score as the quality measure can be found in table 11.

Table 11: The found subgroups using the inverse z-score quality measure. Found threshold for the 95% significance level was 2.53.

Nr.	***Coverage***	***Inv z-score***	***Average***	***St. Dev.***	***Conditions***
3.1	*54*	*2.868*	*-8.18*	*31.88*	*21_std_sum_Constant1[sec] >= 6243.813*
3.2	*69*	*2.865*	*-5.46*	*27.46*	*28_avg_avg_Altitude[m] >= 360.03903*
3.3	*40*	*2.826*	*-11.57*	*36.7*	*28_min_sum_Constant1[sec] <= 865.0*
3.4	*39*	*2.822*	*-11.87*	*37.12*	*21_min_sum_Constant1[sec] <= 865.0*
3.4	*75*	*2.809*	*-4.24*	*32.18*	*28_std_sum_PedalPowerRolling[W] >= 1354601.4*
3.5	*43*	*2.636*	*-8.89*	*31.6*	*28_max_avg_Altitude[m] >= 2157.3584*
3.6	*65*	*2.618*	*-4.26*	*28.99*	*28_std_avg_Altitude[m] >= 398.19525*
3.7	*54*	*2.613*	*-6.1*	*31.9*	*28_std_sum_Constant1[sec] >= 6426.9575*
3.8	*75*	*2.56*	*-2.51*	*33.11*	*28_max_sum_PedalPowerRolling[W] >= 4875393.5*

Due to the nature of the inverse z-score as the quality measure, the averages of the relative results of the subgroups that were found are negative, meaning the cyclist performed well on average. Similarly to the result of the nominal target we find subgroups based on the standard deviation (nr. 3.1, 3.6 and 3.7). Now also previously unseen subgroups based on altitude were found (nr. 3.2, 3.5 and 3.6). Figure 11 shows an overview of subgroup nr. 3.3. The subgroup's stages are very clustered together, only 4 competitions are included.

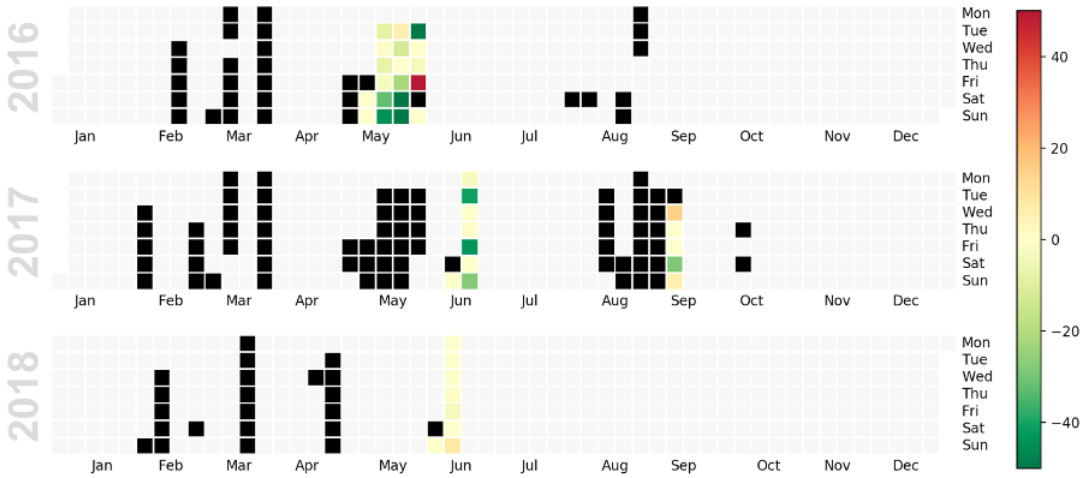

Figure 11: Subgroup nr. 3.3 28_min_sum_Constant1[sec] <= 865.0. Black are all stages that are not in the subgroup or not used. Relative results are clipped to -50 and 50 for visualization purposes.

7 Discussion

7.1 Heart rate model

Compared to the baseline model, the model with the convolution applied to the power output (Table 3) performs significantly better with an increase in R^2 of 0.4. A window size of 50 turned out to be sufficiently long enough to capture how your body is responding to load.

We then evaluated how the model performed on predicting the aggregates that were finally used in the Subgroup Discovery, the final model performed worse on the aggregates per training, especially on the average and the lower percentiles. The average is surprising because it is inherent to OLS regression that the mean of all the errors is 0. Consequently, the OLS should be especially good in predicting the average. The only difference is that the model was optimized for *all* the training sessions combined, for the evaluation of the aggregates we calculate the mean for every *individual* training session. Therefore, we can conclude that there is a lot of variance between the mean error between training sessions, the model is either overestimating or underestimating the heart rate in a training session. It could be interesting to optimize a heart rate model for every training individually and compare the obtained models and training sessions with each other to potentially find other variables affecting the heart rate, and ultimately improve the performance of the model. It could also be the case that the variables that are responsible for the differences in behavior between training are not easily measurable, for instance, illness, diet or stress.

The advantage of the current model is that it tries to predict a training in detail and could therefore be used to also predict every possible aggregate. The disadvantage of this method is that we might predict information that we do not use, making it unnecessarily complicated. Therefore, another method might give better results. One way would be to predict the aggregates of an attribute by constructing features of other attributes. For example, how does the standard deviation of the power output relate to the standard deviation of the heart rate. The disadvantage of this method is that for every aggregate a custom model has to be constructed.

7.2 Power output model

The application of convolution (Table 6) resulted in a significant increase in R^2 of 0.14 compared to the baseline model. However, this increase is a lot lower compared to the increase we observed in the heart rate models. This is because the baseline power output model (R^2= 0.61) performed significantly better than the baseline heart rate model (R^2= 0.39). The difference might be explainable by the fact that we are predicting a rolling average in the case of the power output essentially already applying a convolution, but a less effective one.
The deconvolution method we proposed turned out to successfully work, there was only a 0.01 decrease in R^2 between the model before applying a deconvolution and the model after the deconvolution.

In case of the aggregate evaluation we find similar results between the power output model and the heart rate model. Lower percentiles perform less compared to higher percentiles. The power output model performed even worse in predicting the average. The power output model was also a

lot worse in predicting the standard deviation. The model performed well on the sum aggregate (R^2= 0.93), it performed even better on the sum than the task it was trained on. This could be explained by the nature of the R^2 score, which uses the mean model as a comparison. The mean model would in this case be a bad predictor because there is a large variance in training volume between training sessions. Hence our model performs relatively well.

7.3 Subgroup discovery

All references in this section to subgroup numbers can be found in Table 9, 10 and 11. Independent of the precise settings in all our experiments, there are similarities between the subgroups that were found. More precisely, we found 5 similar subgroups in our dataset, 1.1, 1.2, 1.3/3.4, 3.1 and 3.7. They all describe the variation of training volume. All 5 subgroups share the same characteristic that a variation in training volume above a certain threshold shows a significant lower average in relative result. The period of high variation is between 14 and 28 days prior to the stage. The training volume is either described by the sum of the power output or the duration ("sum_Constant1[sec]"). Because the duration is not dependent on any predictions it makes it more plausible that the subgroups based on the sum of the power output do not exist by chance due to the error in our model. The issue of the standard deviation is that it only gives information about the variation. Because multiple configurations can result in the same value for the standard deviation, this does not give us an exact improvement point. The detection of variation as a positive training impulse, could potentially be described with tapering. With tapering you decrease your training volume and/or frequency before a competition resulting in higher variation between training sessions' volume also the length of conventional tapering periods is similar to the found subgroups.
To get a more concrete understanding of these subgroups, the stages should be studied case by case or more specific features should be introduced. Because presumably tapering causes these subgroups to exist, new window-training-features could be introduced that more accurately represents tapering. These new window-training-features should have the property to be time conscious, thus distinguishing between training sessions closer and further to the stage. All the current window-training-features do not have this property. An example of an interesting feature that could be introduced is the average rate of decay of training volume per week.

In addition to the common results, we have also obtained subgroups that are specific to the experimental settings. The subgroups found for the numeric target with z-score as the quality measure (Table 10) both indicate the cyclist has overtrained and therefore performs worse. However the window size of the two subgroups differ by more than 3 weeks, indicating two different forms of fatigue. Subgroup 2.2,

$$4\_avg\_per95\_PedalPowerRolling[W] >= 380.88977 \qquad \text{(nr. 2.2)}$$

indicates a form of short term fatigue. The subgroup can be explained as the last four days your average highest 5% of intensity of your sessions were at a power output of 380.89 or above. Which is, depending on the exact capacity of the cyclist, above lactate or VO2max threshold and therefore can cause considerable fatigue (Allen and Coggan, 2010). Figure 10 shows that most of the time this condition only occurs after 3 to 4 days of competition, indicating that this fatigue is due to high intensities during competitions. When tactics allow it, the cyclist should consider to keep the

95th percentile of the power output below a certain threshold. Subgroup 2.1,

$$28\_min\_avg\_HeartRate[bpm] >= 100.05824 \qquad \text{(nr. 2.1)}$$

indicates overtraining over a longer period of time. In the past 28 days there was no training with an average heart rate below 100.1. From Figure 9 we can see that all but one of the cases were in the beginning of the 2016 season. It could be this systemic training at high loads contributed to the relatively bad start of the 2016 season and that they learned from this mistake.

Subgroup 3.2, 3.5 and 3.6 all share altitude as a training-attribute and window size of 28. Interestingly, all stages in subgroup 3.5 are also in 3.2. Also, all but 2 stages in subgroup 3.6 are also in 3.2. This suggests there are multiple features describing the altitude acclimatization that causes improved performance. These 3 subgroups indicate that training at high altitude 28 days prior to a stage improves performance. The exact altitude to train at is unknown.

Subgroups 3.3 and 3.4 are almost exactly the same except 3.3 has one additional stage.

$$28\_min\_sum\_Constant1[sec] <= 865.0 \qquad \text{(nr. 3.3)}$$

Subgroup nr. 3.3 suggests a at least one training of low duration during a training-period will improve performance. There are a few issues with this description of the subgroup. Firstly, moving the threshold from 865 to 1208, only increasing your training duration with about 5 minutes, the average relative time of the subgroup will move from -11.57 to 5.03. Secondly, looking at Figure 11 we see that all stages in the subgroup are only divided over 4 competitions and because the window size of the subgroup's feature is 28 days long, it means only 4 different training sessions cause this subgroup to exist. This makes it more sensitive to errors. The general problem with min aggregates in the form of $\min < x$ where x is relatively small, is that it only provides information on a single low volume training session. Whereas the min aggregates in the form of $\min > x$ where x is relatively high, it can provide you with information about lack of resting during the whole period.

Subgroup 3.8,

$$28\_max\_sum\_PedalPowerRolling[W] >= 4875393.5 \qquad \text{(nr. 3.8)}$$

suggests a single training of high volume during a 28 day training period increases performance. Considering the subgroup is barely significant and similar to subgroups 3.3 and 3.4, the condition only provides information on a single training and thus large amounts of effects are more improbable. However this subgroup might be better explainable using the literature. High amounts of load might lead to adaption of different parts of your body, thus a single training might already give a noticeable stimulus to your body.

For all our subgroups we used relative results as our target/performance measure. A general disadvantage of using relative results is that it does not measure how much of the cyclist reserves were used. Two races with the same relative result do not mean it was done with the same amount of ease. For example, in the case of a relative result of 0, a mass finish, because the cyclist could draft a large amount of time it could be it still took a lot of effort in order not to lose the group

or maybe the cyclist was saving energy for other stages still to come. In addition, relative result does not represent the objective capacity of the cyclist's body, this could happen in multi-day competitions. For instance, if a cyclist got a better relative result on the 10th day of a competition compared to the 1st day, it does not mean the cyclist was more fit on the 10th day, in fact the cyclist was probably more fit the 1st day because the cyclist had not used up its reserves yet. The relative result thus only measures your strength compared to your opponents.

8 Conclusion

In this thesis we studied the response of performance on aspects of training. To answer the research question we first developed models to predict the power output and heart rate to fill up missing data. As expected the application of a backwards looking exponential convolution to the power output has shown to significantly improve the performance of the heart rate and power output model. We were able to construct models with an accuracy of $R^2 = 0.78$ and $R^2 = 0.75$ for the heart rate and power output, respectively. However, the R^2 accuracy in predicting individual training session aggregates was lower and ranged from 0.16 to 0.75, with the exception of the sum aggregate applied to the power output, which had an R^2 accuracy of 0.93.

Once the missing data was filled up we could create the features of the training-periods. We used the proposed quantification framework to obtain a dataset describing the training-period on multiple aspects over multiple time-frames. We then applied subgroup discovery on the obtained dataset with multiple settings.

The subgroup discovery resulted in multiple findings. The most recurring features were the features that represented the variety between sessions' volume, either represented by amount of energy used or duration. Higher variation between the volume of the sessions showed a significant increase in performance for both the nominal and numeric target. The variation should occur in a window of size 14 to 28, but precise improvement points could not be concluded because of the many possible different configurations that results in high variation between your training sessions. A possible explanation of these type of subgroups could be tapering, because tapering implies a higher variation in your training session's volume.
Subgroup nr. 2.2 indicated a significantly worse performance when the 4 days prior to the stage had been relatively intense. Mostly this occurred because of past stages and thus the cyclist might not always have a choice to alter their behavior. The cyclist thus should consider, when tactics allow it, to not use up all of their resources.
Multiple subgroups showed that altitude positively affects the performance significantly and the cyclist should consider training at high altitude at least 4 weeks prior to the stage.
Other subgroups that were found showed less conclusive results. Either because the conditions of the subgroup seem improbable (Subgroups nr. 3.3 and 3.4), the stages were not well spread over the period of available data (Subgroup nr. 2.1) or because it was barely significant (Subgroup nr. 3.8).

Thus, we can conclude we were able to find patterns in the training data that reflect the performance of the cyclist in a competition. Although this gives first insights in the relationship between training and race performances, for more concrete improvement points the most promising subgroups should be further analyzed. A way to further analyze these subgroups could be a per stage case study or new types of a more specific features could give new insights. Also, more recent data of the cyclist being studied could give extra confirmation on our results and would also show if our results are still applicable to cyclist today.

References

Allen, H. and Coggan, A. (2010). *Training and Racing with a Power Meter*. Boulder Colorado, third edition.

Bunc, V., Heller, J., and Leso, J. (1988). Kinetics of heart rate responses to exercise. *J Sports Sci.*, 6(1):39–48.

Calvert, T. W., Banister, E. W., Savage, M. V., and Bach, T. (1976). A systems model of the effects of training on physical performance. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(2):94–102.

Clarke, D. and Skiba, P. (2013). Rationale and resources for teaching the mathematical modeling of athletic training and performance. *Advances in physiology education*, 37:134–152.

Duivesteijn, W. and Knobbe, A. (2011). Exploiting false discoveries – statistical validation of patterns and quality measures in subgroup discovery. pages 151–160.

Ekelund, L.-G. (1967). Circulatory and respiratory adaptation during prolonged exercise of moderate intensity in the sitting position. *Acta Physiologica Scandinavica*, 69(4):327–340.

Erp, T., Foster, C., and de Koning, J. (2018). Relationship between various training-load measures in elite cyclists during training, road races, and time trials. *International Journal of Sports Physiology and Performance*, 14:1–25.

Faria, E., Parker, D., and Faria, I. (2005). The science of cycling: physiology and training - part 1. *Sports medicine (Auckland, N.Z.)*, 35:285–312.

Grazzi, G., Alfieri, N., Borsetto, C., Casoni, I., Manfredini, F., Mazzoni, G., and Francesco, C. (1999). The power output/heart rate relationship in cycling: Test standardization and repeatability. *Medicine and science in sports and exercise*, 31:1478–83.

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

Jobson, S., Passfield, L., Atkinson, G., Barton, G., and Scarf, P. (2009). The analysis and utilization of cycling training data. *Sports medicine (Auckland, N.Z.)*, 39:833–44.

Knobbe, A., Orie, J., Hofman, N., Burgh, B., and Cachucho, R. (2017). Sports analytics for professional speed skating. *Data Min. Knowl. Discov.*, 31(6):1872–1902.

Ludwig, M., Grohganz, H., and Asteroth, A. (2016). A convolution model for heart rate prediction in physical exercise.

Martin, J., Milliken, D., Cobb, J., McFadden, K., and Coggan, A. (1998). Validation of a mathematical model for road cycling power. *Journal of Applied Biomechanics*, 14:276–291.

Massey, H., Corbett, J., Barwood, M., and Tipton, M. (2011). Cycling cadence affects heart rate variability. *Physiological measurement*, 32:1133–45.

Meeng, M. and Knobbe, A. (2011). Flexible enrichment with cortana – software demo.

Mujika, I. and Padilla, S. (2001). Physiological and performance characteristics of male professional road cyclists. *Sports medicine (Auckland, N.Z.)*, 31:479–87.

Pieters, B., Knobbe, A., and Džeroski, S. (2010). Subgroup discovery in ranked data, with an application to gene set enrichment.

Reeves, J. T., Groves, B. M., Sutton, J. R., Wagner, P. D., Cymerman, A., Malconian, M. K., Rock, P. B., Young, P. M., and Houston, C. S. (1987). Operation everest ii: preservation of cardiac function at extreme altitude. *Journal of Applied Physiology*, 63(2):531–539. PMID: 3654411.