ML4QS Assignment 3 - Group 54

Hong Huo[1][2726433], Matteo De Rizzo[2][2749303], and Seth van der Bijl[3][2649279]

[1] Vrije Universiteit Amsterdam, 1081 HV Amsterdam, Netherlands
[2] h2.huo@student.vu.nl
m.de.rizzo@student.vu.nl
seth.vander.bijl@student.vu.nl

1 ABSTRACT

The present research attempts to answer the question to what extent a Random Forest (RF) model outperforms other Machine Learning (ML) models and baselines for regressing the number of push-ups done in the last 10 seconds.[3]. In order to assess the superiority of the RF model on this task, it was compared to a multitude of models and by employing a pipeline of several feature engineering steps, it is concluded that RF outperforms all other ML methods and baseline except Decision Trees (DT).

2 INTRODUCTION

As a result of the technological rise we have seen in the last decades, people now use many more sensory inputs to learn from their environment. A century ago we were mostly limited to our biological sensors but to this date 6.64 billion individuals own a smartphone, which becomes an extension of one's self, adding a wide variety of novel sensory information. The data produced by these phones can be used on a multitude of applications, but this paper will focus on the quantified self[4]. The goal of the quantified self is to be able to process data from daily activities and exercise, to ideally improve mental, physical and emotional performance[10].

Society has become more sedentary and has developed a lack of physical activity, therefore increasing health risks. Feedback is needed to change these behaviours. Physical activity can improve health and well-being, reduce the risk of many diseases and improve the quality of life. A number of studies have been conducted to improve the movement prediction of the quantified self. The employment of machine learning or artificial intelligence has gained popularity in predicting and classifying quantified self properties as well as activities owing to its superiority over conventional means[10] along with some concerns about valid usage of this data[9].

K-nearest neighbour (k-NN) has been shown to be an effective learning algorithm for classification and prediction[3]. K-NN is a non-parametric regression

[3] The accompanying code of this research assignment can be found on GitHub

and classification method developed by Fix and Hodges in the 1950s. It is regarded as one of the simplest forms of supervised machine learning algorithms. Nonetheless, k-NN did not gain considerable attention until the sixties due to the limited computing power available at that time[1].

In a recent study, Pavey et al. also utilised a random forest (RF) classifier to classify different forms of activity (sedentary, stationary+, walking and running) through data obtained via wrist-worn accelerometer[8]. Furthermore, Support Vector Machine (SVM) and Neural Networks (NN) are proposed for use in the prediction of movements to ensure that the information gotten from the system built based on these techniques are reliable[4]. Some studies where able to employ NN and SVM for classifying different fitness activities with an accuracy of 87.5%[7].

In our study, we monitor various data that can be captured via a smartphone and smartwatch and correlate them to a target variable that represents the measure of the number of push-ups. In subsequent sections, we propose a method to convert raw sensor data from a smartphone into categorical features and extract features.

Furthermore, we compare the prediction accuracy between several sets of features to determine whether our proposed method, comprising a combination of features, can successfully predict the amount of push-ups with higher accuracy compared with other machine learning models that has been reported in previous studies.

This literature background and problem statement informs the following research question: *To what extent does an RF model perform significantly better than other ML methods and an average baseline in predicting the number of push-ups in the last 10 seconds from wearable's sensor data?*

3 METHODS

3.1 Data collection & merging

Data was collected by a single person performing a sporting activity where the number of push-ups in the last training set or being inactive (0 push-ups) was annotated. The data was collected using an an iPhone SE 2 using the sensors: accelerometer, gravity, gyroscope, orientation, magnetometer, barometer, location and microphone/ The data was collected simultaneously from the Apple Watch 2 with the sensors for heart rate and wrist motion. This original data resulted in 48 columns for the several axis of the sensors and a time and annotation column.

Most sensors where quantified at a sampling rate of 100Hz for 40 minutes while some sensors where sampled at a rate of 10Hz or 0.1 Hz. Merging the data from these different sensors resulted in an initial dataset with 280000 rows.

3.2 Annotation processing and backfilling

The annotation column was populated extremely sparsely (i.e. annotating an activity only annotates it for the particular millisecond or point in time) however,

it was known that all timepoints before the annotation up until the previous annotation where part of this activity. This was cause to backfilling the annotation column providing every row in the dataset with an annotation for how many push-ups where done in this set. Some exploratory data analysis (EDA) was performed after this step where every sensor was resampled to 100ms, linearly interpolated and plotted to spot preliminary patterns and anomalies which is illustrated in the figure below. The EDA process was interwovenly continued in the operations following. Apprehending visually that certain sensor patterns happened at the intervals of the push-ups positively reinforced expectations for the remained of the inquiry.

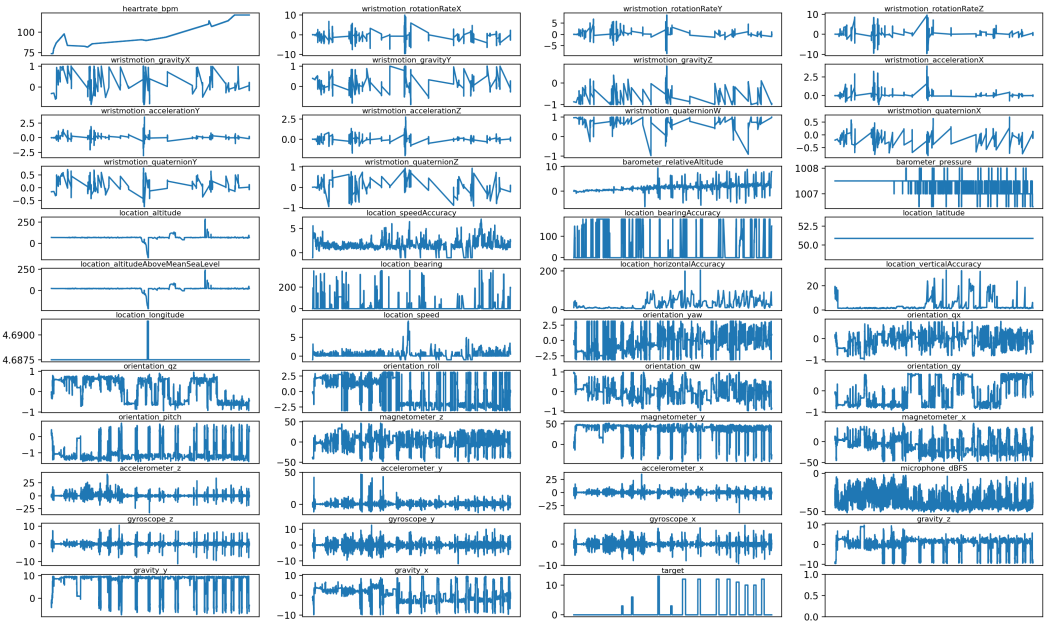

Fig. 1: Interpolated measurements for all sensors

3.3 Outliers

A subset of outliers was present in the data. Some of these outliers where hypothesized to be causes by the phone hitting the floor sometimes when doing push-ups. After analysis all outliers where inferred to be natural outliers and not due to measurement errors. Since it was expected that these outliers stood in a certain relationship with the push-ups themselves the outliers where not removed.

3.4 Missing Value imputation & resampling

Even though the data from different sensors where mostly collected at the same frequency the measurements for different sensors started at slightly different time points causing there intervals to be just off and the be a lot of NaNs for either one sensor or another. A particularly natural way of resolving these missing values was to resample, which was necessary to create a regularly intervalled timescale for the following fourier transformation but also provided a means of 'connecting' the measurements of different sensors being slightly off to each other. In this initial resampling a finegrained window of 200ms was chosen since it was regarded important to preserve the finegrained details for other processes in the data engineering pipeline. This resampling diminished the number of rows to 12564 of which a majority still a NaNs in the form of the location and heartrate being only measured every 10 to 20 seconds. Since all other sensors had no NaNs anymore and only these secondary sensors still caused the majority of rows to still have NaNs it was inferred that linearly intperolating the remaining NaNs was a conceptually valid approximation of the actual values. The last rows which could not be interpolated where dropped yielding a dataset with 11990 rows at 200Hz.

3.5 Frequencies

Fourier transformation The Fourier transform is commonly used to convert a signal in the time spectrum to a frequency spectrum. This transformation was reasoned to be particularly useful for the repeated motions and even vibrations happening when doing push-ups.

A FFT is a trade-off between time information and frequency information. By taking a FFT of a time signal, all time information is lost in return for frequency information. To keep information about time and frequencies in one spectrum, we must make a spectrogram. These are DFT's taken on discrete time windows.

A Fourier Transform will break apart a time signal and will return information about the frequency of all sine waves needed to simulate that time signal. For sequences of evenly spaced values the Discrete Fourier Transform (DFT) is defined as: $X_k = \sum_{n=0}^{N-1} x_n e^{-2\pi i k n / N}$.

Window overlap At this stage datapoints where becoming more similar due to resampling, interpolating and having a rolling window for the fast fourier transformations. A similarity window of 90% was allowed to allow preservation of 6000 rows of the dataset whereas the other rows where to similar and filtered out.

3.6 Clustering

Immediately before attempting integration of clustering EDA was resumed where the correlations of each column with the target column was noted. The nine

strongest correlations in respectively growing order where: gravity-y, magnetometer-x-freq-0.0-Hz-ws-50, gravity-z-freq-0.0-Hz-ws-50, gravity-x, orientation-pitch, gravity-x-freq-0.0-Hz-ws-50, gravity-y-freq-0.0-Hz-ws-50, accelerometer-y-freq-0.0-Hz-ws-50, orientation-pitch-freq-0.0-Hz-ws-50 with $0.39 < \rho_{x_1 \ldots x_9} < 0.49$. Noting these strong correlations of the slow frequencies of gravity and orientation meters provoked particular interest towards the remainder of the process. Initial clustering was attempted using the k-means algorithm using all columns but since these results did not intuitively appear satisfactory three clustering operations where performed using sets of these strongest correlating columns. For every clustering operation an appropriate () set of three was used to add three clustering columns to the dataset as illustrated below.

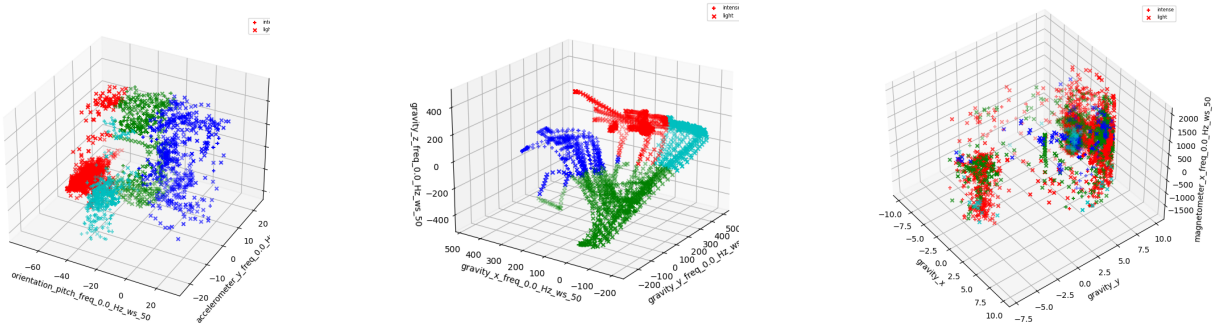

Fig. 2: Three consecutive clustering operations on the most correlated columns

For the purposes of visualization the number of push-ups was divided into a 'light' and 'intense' class representing respectively doing less than or equal to eight push-ups or more than eight push-ups. As can be noted from the clusters[4] the clusters show a tight in-group connectivity and are well-separated from the other clusters. Furthermore, the clusters strongly correspond to either light or intense intensity push-ups. This findings where taken as indication of a possible positive influence of including these clusters as features.

3.7 Random Forest

The main reason for choosing RF as a primary method and other ML algorithms as baselines (as contrasted to a timeseries approach) was the fact that the number of push-ups is not truly a metric that could be encoded nicely in a timeseries. Ascribing a certain point in time a certain amount of push-ups would be unnatural and the nature of the problem was better denoted as a number of push-ups done in a certain timeperiod combined with engineered features for the time dimension.

[4] More detailed versions of these plots can be found at All figures on GitHub.

For purposes of brevity only RF is described of all the ML models. RF was chosen with the rationale that the non-linearity of the data combined with the strong preliminary engineered features would constitute material very suited for employment of RF. Proposed by Ho[5] in 1995 and extended by Breiman[2] RF constitute an ensemble method where a host of weak learner trees is combined in a bagging approach to acquire considerable performance. RF has proven very performant in a number of cases with structural data.

3.8 Experimental set-up

As a preliminary to testing the different Machine Learning (ML) models the data was resampled to 10 seconds for two reasons: First, the temporal dimension of the data was not regularly intervalled anymore after the window overlap processing and, secondly, from a practical point of view it was more interesting the number of push-ups in the last 10 seconds than it is to predict the amount of push-ups in the last 200 milliseconds. This resulted in a dataset of 240 rows.

With the aim of comparing the RF with several other ML models and baselines the experiment was set-up in the following way. An average baseline, linear regression baseline (LR), RF, support vector machine (SVM), neural network (NN), decision tree regressor (DT), gaussian naive bayes (NB) and gradient boosting regressor (GB) where set-up for comparison. Over 18 iterations the training and test set where reshuffled for fair comparison and the mean squared error (MSE), mean absolute error (MAE) and R^2 where recorded. The test-set was always 20% of the data. For most models the default sklearn parameters where selected except for the NN where a logistic activation function proved to deliver better and less varying results than the default reLU.

4 RESULTS

An intuitive representation of the performance of the algorithms is represented in the following boxplot where one notices RF yielding the best performance with a considerable distance to the other algorithms and baselines except BR.

Even though this yields insight it does not explicate the full picture regarding significances yet. The tables below show the significant differences among groups using ANOVA for the different metrics of MSE, defined as:$\sum_{i=1}^{D}(x_i - y_i)^2$, MAE, defined as: $\sum_{i=1}^{D}|x_i - y_i|$ and R^2. R-squared, also known as the coefficient of determination, is the statistical measurement of the correlation between an investment's performance and a specific benchmark index. The R-squared formula is calculated by dividing the sum of the first errors by the sum of the second errors and subtracting the derivation from 1. As can be seen is the difference for every metric among the different methods significant with respective p-values of 0.001, < 0.001 and < 0.001.

The table below shows the significance of the different metrics of the RF model versus the other models. Since this significance is measured by multiple testing a Bonferroni correction is applied conservatively alleviating the

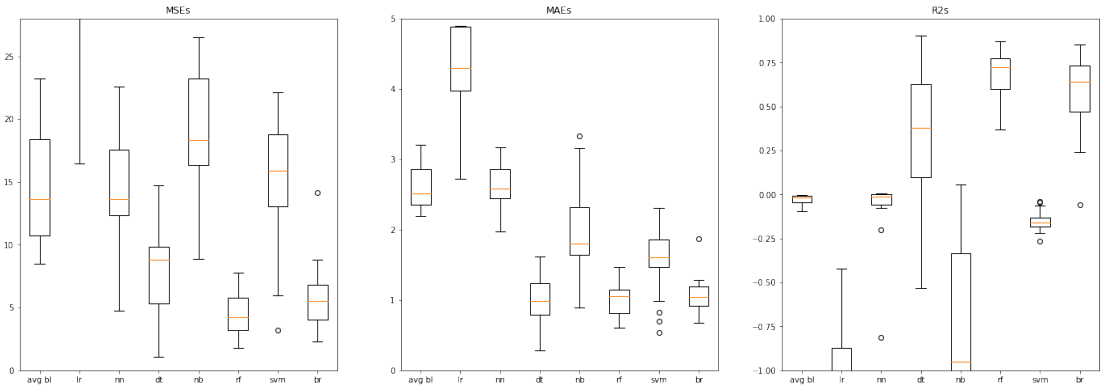

Fig. 3: Performances of baselines and different algorithms on different Metrics

	Source	SS	DF	MS	F	p-unc	np2
mseses	group	3.010826e+06	7	430117.979	3.656	0.001	0.158
	Within	1.600116e+07	136	117655.599			
maeses	group	2.957250e+02	7	42.246	33.035	0.000	0.630
	Within	1.739240e+02	136	1.279			
scoreses	group	1.469330e+04	7	2099.043	4.071	0.000	0.173
	Within	7.012162e+04	136	515.600			

Table 1: ANOVA results for different metrics

multiple testing problem. The RF model has significantly different MSE from all models except LR and BR. The formula for a Bonferroni Correction is $\alpha_{new} = \alpha_{original}/n$.

The model has significantly lower MAE than every model except DT and BR. In a fashion similar to MSE the R^2 is significantly different from every other model except LR and BR.

	avg bl	lr	nn	dt	nb	svm	br
MSE	< 0.001	> 0.999	< 0.001	0.015	< 0.001	< 0.001	> 0.999
MAE	< 0.001	< 0.001	< 0.001	> 0.999	< 0.001	0.005	> 0.999
R2	< 0.001	0.995	< 0.001	0.02	< 0.001	< 0.001	> 0.999

Table 2: RF multiple t-tested against other models with Bonferroni correction

5 DISCUSSION

The RF performs best across all fronts, mostly significantly better than other models but not in all cases. Indeed, the differences with DT are not significant and it is noted that DT actually has a lower MAE. The higher MSE tells us that the DT probably produces more average and safe estimates leading to lower

MAE but much higher MSE for the larger errors it accrues when being more off than RF. Furthermore, is it intriguing to note the extreme variance in R^2 for DT with some models not explaining any of the variance and the MAE still being the best on average over all these models. This is imperative to the statistical hypothesis about the data that even though the DT has a very low bias it has a very high variance since the high scoring model explains only little of the target variance.

Similar low MAE combined with low R^2 is found for the other models indicating again that they might not be very biased but do have high variance. This is further reinforced in inspecting the MSEs where only NB, DT and BR perform considerably better than average baseline and other equal or worse. As such, most models tend to make large mistakes even though having a low MAE, so not being very off in general.

The single best performance was for a DT with an MAE of only 0.28 and an R^2 of 0.89. This was a singular performance and not an average performance and could be well caused by the particular configuration of the train and test set in that case.

An secondary peculiarity is the significantly different MSE from all models except LR and BR for RF. A strong candidate hypothesis for this peculiar phenomenon might be that the BR model performance strongly overlaps while the LR model performance has such extreme variance (several thousand magnitudes higher than the other models) that it statistically cannot be stated as significantly different.

Comparison against previous research in quantitative terms is rendered hard by the controlled collecting of the present dataset and the different research aims of other research, focusing on activity classification rather than push-up regression. Nevertheless, does the present research extend the previously successful activity classification[7] or push-up posture accuracy[6] to intensity regression.

Future research might involve itself with a more representative, multi-person, multi-setting dataset yielding a general algorithm for quantifying the number of push-ups in the last time period instead in a general setting instead of the uniform dataset collected in this case. Furthermore, since the clustering in this research already returned such accurate results, unsupervised approaches in itself might be employed to advance quantification of sport measurements.

6 CONCLUSION

A RF model is able to achieve strong results in quantifying the number of push-ups in the last ten seconds in a single setting for a single recorded over prolonged time. RF performs significantly better than baseline and several other ML methods but not significantly better than DT. The extent of this performance is hypothesized to be caused by the controlled recording of the dataset and several preprocessing steps strongly correlating to the target before applying ML.

References

[1] David Adedayo Adeniyi, Zhaoqiang Wei, and Yang Yongquan. "Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method". In: *Applied Computing and Informatics* 12.1 (2016), pp. 90–108.

[2] Leo Breiman. "Random Forests". en. In: *Machine Learning* 45.1 (Oct. 2001), pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324 (visited on 06/25/2022).

[3] Padraig Cunningham and Sarah Jane Delany. "K-nearest neighbour classifiers-a tutorial". In: *ACM Computing Surveys (CSUR)* 54.6 (2021), pp. 1–25.

[4] Edwin A Fleishman. "The structure and measurement of physical fitness." In: (1964).

[5] Tin Kam Ho. "Random decision forests". In: *Proceedings of 3rd International Conference on Document Analysis and Recognition*. Vol. 1. Aug. 1995, 278–282 vol.1. DOI: 10.1109/ICDAR.1995.598994.

[6] Junseok Lee, Donghan Oh, and Kyung-Il Ahn. "Measurement of Push-up Accuracy Using Image Learning by CNN". kor. In: *Journal of Korea Multimedia Society* 24.6 (2021). Publisher: Korea Multimedia Society, pp. 805–814. ISSN: 1229-7771. DOI: 10.9717/kmms.2021.24.6.805. URL: https://www.koreascience.or.kr/article/JAKO202119759357803.page (visited on 06/25/2022).

[7] Mårten Nilsson and Herman Wilén. *Push-up Tracking through Smartphone Sensors*. eng. 2016. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-201035 (visited on 06/25/2022).

[8] Toby G Pavey et al. "Field evaluation of a random forest activity classifier for wrist-worn accelerometer data". In: *Journal of science and medicine in sport* 20.1 (2017), pp. 75–80.

[9] Thomas Reichherzer et al. "Using machine learning techniques to track individuals & their fitness activities". en. In: *CATA 2017*. Accepted: 2017-06-18T22:00:11Z. ISCA, 2017, pp. 119–124. ISBN: 978-1-5108-3666-2. URL: https://researchcommons.waikato.ac.nz/handle/10289/11113 (visited on 06/25/2022).

[10] S Robertson. "Improving load/injury predictive modelling in sport: The role of data analytics". In: *Journal of Science and Medicine in Sport* 18 (2014), e25–e26.