Machine Learning Project 2 - Predicting Energy Expenditure

Tobias Bodenmann, Arthur Cornet, Lucas Guirardel
Referents : Dolaana Khovalyg, Arnab Chatterjee (TEBEL)

***Abstract*—The goal of this study is to develop a model capable of predicting individual energy expenditure from physiological and environmental variables, as easy to measure as possible. Experimental data was collected and has already been used before to train machine learning models: however, we found a problem of overfitting on subjects in this previous work. While proving that claim in this report, we also propose a new data processing pipeline as well as a rectified model, that we could not fully test by lack of data, but that would be easy to enrich with new experiments.**

I. Introduction

In Europe, it is estimated that buildings are responsible for about 40% of the energy consumption, and 36% of all CO2 emissions [1]. Therefore, one of the main challenges of our century consists in optimizing buildings design and operation, which can be done using numerical simulations. Such simulations, in order to be effective, require that we are able to understand and predict how inhabitants will react to environmental changes, e.g. of air temperature or speed. One variable that can be chosen to evaluate people physiological reactions is their energy expenditure (EE), which is measured as a function of oxygen intake and CO2 production.

We would also like to be able to adapt in real time the energy consumption of a building. In order to do that, we must be able to evaluate at every moment the EE of its inhabitants, without of course having to use an oxygen mask. That is why we would like to develop a model that can predict EE using environmental variables, but also physiological characteristics that are more convenient to measure, such as heart rate, that can be easily obtained using a smartwatch.

With this in mind, experiments were carried out on several subjects, following different protocols, and EE was measured, as well as body and environmental variables. Using this data, some predictive models have previously been developed, but we will show that they are not really satisfying, and that the approach used needs to be modified. Therefore, we will try and propose a better model to predict EE using wearable and environmental sensors. As we have seen, we should try to ideally use the least number of sensors possible: therefore, data preprocessing will play a key role in our study, as it is already a way to select relevant variables for our reduced data set.

Data visualization and preprocessing were done using the following Python libraries: NumPy [2], Matplotlib [3], Pandas [4] and Seaborn [5]. We used machine learning algorithms implemented in scikit-learn [6]. Finally, we used scipy [7] and statsmodels [8] for statistics.

II. Data preprocessing

A. Initial data set

The data set consists of time series of different measurements, made for 6 different experimental protocols (1, 2, 3a, 3b, 4a, 4b), on 6 different subjects. Each protocols is about 5 hour-long, and involve modifying experimental conditions over time, e.g. room temperature or subject activity level. Protocols 3 and 4 are separated into 2 variants, option A being a control for option B. All subjects undergo all 6 protocols, with some exceptions (on average, each protocol has data for 5 different subjects).

For each sensor, data is sampled every minute, so there are about 300 data points for a given protocol and subject, on each variable: that amounts to about 1500 rows for each protocol. Initially, there were 54 different features (columns), separated into different categories:

- **COSMED variables** (indirect calorimetry): measure of energy expenditure, from oxygen intake and CO2 production, and of time. These variables will not be used for prediction. Only EE is used, as the value to predict.
- **Body variables**: they are measured using wearable sensors, such as smartwatches, accelerometers, bodypatches, attached buttons... They consists of (non-exhaustive list): heart rate, acceleration in 3 directions, stress level and activity type (from smartwatch), skin temperature. These measurements are rather intrusive (except from the smartwatch), thus it would be preferable to reduce as much as possible the number of measurements to keep, so that the protocol can easily be repeated on regular inhabitants (not subjects), if we wanted to predict their energy expenditure in real time.
- **Environmental variables**: they are measured in the laboratory, and consist mostly of temperatures measured in different ways (air, globe and blackglobe temperatures), at different altitudes of the room (0.1m, 0.6m, 1.1m and 1.7m). They also include air speed, relative humidity and CO2 concentration measurements. We would like to use these variables as much as possible, since they are easier to measure in a real-life case. However, we will see that many environmental variables are highly correlated. Moreover, we suspect that they will have less influence on EE than body variables, such as heart rate for instance, which is more directly related to oxygen intake and CO2 production.
- **Subject characteristics**: the part of the data which does not depend on time. Apart from the code name of the subject, we are given their sex, age, weight and height.

B. Reducing body variables

Many of the body variables have already been reduced before the start of our project. As a result, there were not many highly correlated variables left. Keeping in mind our goal to keep comfortable sensors and reduce more invasive ones altogether without losing information, we further reduced the remaining 18 to just 7 variables.

First of all, one sensor predicts the core body temperature based on the raw heat flux signals at the side of the chest. This sensor yields 7 different variables. Two of them are needed to predict the core body temperature, and therefore, are highly correlated with it. The remaining four contain measurements that are not of use to predict EE.

Secondly, there are two accelerometers, each providing the x, y and z component of the acceleration once measured at the level of the upper leg and once at the level of the chest. Surprisingly

the two sensors were not significantly correlated to each other, but highly correlated within with respect to the three directions. To reduce this correlation but not lose information, we decided to take the norm of the acceleration for each of the two sensors.

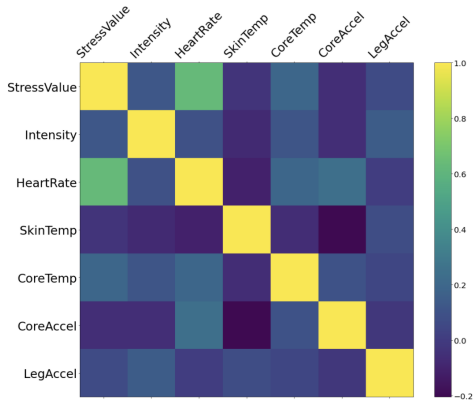

Figure 1. Correlation matrix of the reduced set of body sensor variables (Garmin stress level value, Garmin intensity, Garmin heart rate, skin temperature (iButtons 15l), predicted core temperature, chest acceleration (CORE) and upper leg acceleration (MOX))

The final correlation matrix of the reduced body variables shows almost no remaining correlations (figure 1). The only exception is between the stress level and the heart rate value that provide a non negligible correlation coefficient of 0.8. We nevertheless decided to keep both values, because from a physiological point of view they both seem important to predict EE.

C. Reducing environmental variables

Unlike the body variables presented above, we noticed that many environmental variables were highly correlated, which enabled substantial dimensionality reduction.

First of all, the three different types of temperature (air, globe and blackglobe) had very similar time series, which was confirmed by the fact that each pair of temperatures (averaged on all heights) had a correlation factor superior to 0.99, for all protocols. Therefore, we decided to keep only the air temperature, which is the easiest to measure in a real-life context. Likewise, air temperature measured from sensors placed at different heights were also relatively highly correlated ($R^2 > 0.97$): here again, we decided to reduce the variables and keep only the average of all heights.

Thus, we went from 13 different temperature variables (different types and heights) to only 1 (mean air temperature), which reduces greatly the model complexity as well as the number of measurements required. Likewise, the air speed is measured at different heights, but this does not bring much new information, so we only keep the average in the end. Finally, the two remaining environmental variables are the relative humidity (RH) of the room, and the CO2 concentration. We do not notice any interesting correlation concerning these two variables, which we keep in our final reduced set.

The final reduced set of environmental variables is therefore composed of 4 features. The correlation matrix of this set is shown figure 2, where we also added the energy expenditure. It is worth noticing that there is no strong correlation left between environmental variables, but also that EE cannot be directly and easily predicted from only one of the variables.

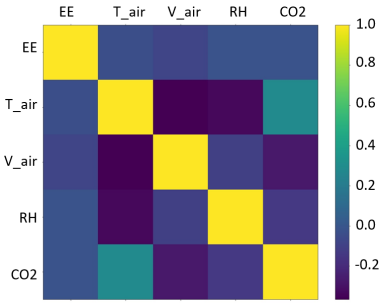

Figure 2. Correlation matrix of the reduced set of environmental variables (air temperature, air velocity, relative humidity, CO2 concentration), plus energy expenditure (EE).

D. Final cleanup and standardization

Before training the models on the data set, we set up a pipeline that takes care of the necessary preprocessing steps. First, we remove outliers, such as zero heart rate values. We also remove all the timestamps, because we do not want the model to learn the experimental patterns from the time information. Then, we apply one-hot encoding to categorical variables such as gender. Lastly, we apply standardization to all variables.

For now, we keep all subject characteristics in the data set, but we will see eventually that we need to treat them differently from the other variables.

III. First models - and why they are wrong

A. The issue with previous work

As explained previously, machine learning models were already developed on this dataset to try and predict EE. These models enabled satisfactory predictions, with high correlation coefficients. However, we argue that these models hide serious overfitting problems, and would not have been able to make good predictions for new subjects outside the train set. In our opinion, this comes from the fact that complex models were used (e.g. Random Forest and Neural Networks), but subject characteristics were kept in the data set: since EE is very subject-dependent, and since there are only 6 subjects, we suspect that the models learned to recognized the subjects, and did not make any generalization based on the variables values.

In order to prove that claim, we reproduce a similar approach, keeping the subjects characteristics in our training set and treating them as any other variable. Using this method, we implement models of various complexities (linear regression, random forest, neural networks), and then demonstrate that the obtained models are prone to overfitting, and can not be generalized to new subjects.

B. Linear regression

The first model tested is a linear regression, with a polynomial feature expansion. To prevent overfitting as much as possible, a regularized version is used (ridge regression), along with a 5-fold cross validation (cv). In order to choose the best hyperparameters (polynomial expansion degree and ridge regularization parameter), we train a model for several pairs of hyperparameters, and note the average correlation coefficient on the 5 different cv test sets. This coefficient is computed over all shuffled data points, from all subjects and protocols mixed. The result is showed figure 3.

This baseline result turns out to be already pretty satisfying: indeed, if we look at some time predictions of EE made with

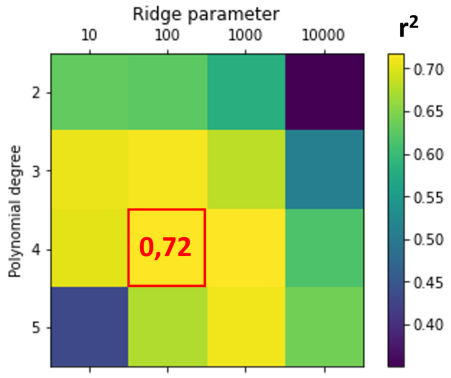

Figure 3. Hyperparameter tuning for ridge regression. For each pair of polynomial expansion degree and regularizer value, the correlation coefficient R^2 is plotted. The maximum value (0.72) is highlighted in red.

this model (figure 4), they follow the same trend as the real value.

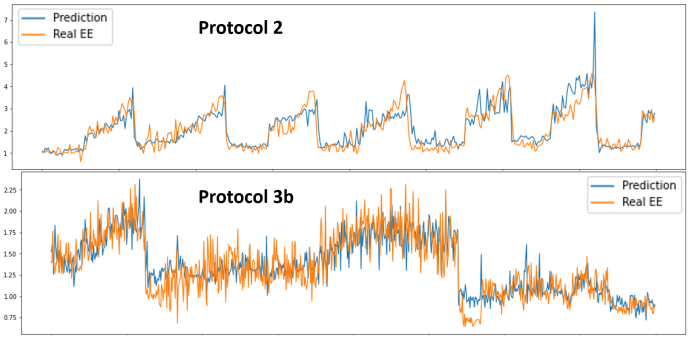

Figure 4. Time evolution of real (orange) and predicted EE using linear regression model (blue), for two different protocols.

However, the achieved accuracy, which is similar to the one obtained with previous models, is misleading. To demonstrate that, let us use the same model with the same hyperparameters, but instead of scoring it with 5-fold cross-validation, we use leave-one-out cross-validation: pick a subject S_0, train on all the other subjects, and finally test on S_0. This method allows us to verify that the model does not simply "recognize" the subject to make its prediction, and is capable of generalizing to new subjects. However, the results show that this is not the case, as summarized in table I. The first row corresponds to the classical 5-fold CV method, while the second one results from performing leave-one-out CV. By looking at the correlation coefficient between true and predicted EE for each subject, it is clear that while the model performs well on subjects seen in the training set, it does not on new subjects. This proves the fact that the first method was able to recognize the subject, and did not really "understand" the general link between body/environmental variables and EE.

Subject	WT8	HL7	QR5	TS3	PS2	DF5
5-fold cv	0.49	0.80	0.65	0.43	0.67	0.76
Subject cv	-1.1	-1.2	-0.0041	-3.4	-0.80	-1.3

Table I. Correlation coefficient R^2 on test sets for all 6 subjects.

C. Random forest

Random Forest regression is a common supervised learning algorithm based on ensemble learning, consisting of multiple decision trees. The predictions across all trees are then averaged to yield a more accurate prediction. Random forest models are usually very accurate, robust and correct for the overfitting of simple decision trees.

Scoring this model on the entire data set using 5-fold CV yielded an R^2 coefficient of 0.84, which was reached with 105 trees. On the dataset without subject characteristics (age, weight, etc.), a R^2 value of 0.80 was reached, this time with 115 trees.

Finally, checking this model with leave-one-out validation, we obtain an average R^2 of -0.20, which means that our model, on average, performs worse on new subjects than a model that would just output the average EE on that subject. Although the values of R^2 vary according to which subject is left out, and can go as high as 0.5, this still shows that the Random Forest model overfits.

D. Neural network

To back up our last claim, we apply the leave-one-out method to neural networks. By doing so, we ensure that the model can not use average values to overfit, as it did with random forest by simply removing subject characteristics.

A multi-layer perceptron with 3 hidden layers of 40 neurons each is used, with $lr = 10^{-3}$ and ReLU activation. The predicted EE versus the true EE is displayed on figure 5, for both the 5-fold split (80% train set, 20% test set) and the leave-one-out split. The first split gives as planned a nicely correlated prediction with the true EE, while the second split is pretty bad : the neural network also overfits. Therefore, we need to develop a more advanced model if we still want to make good predictions, while not overfitting on specific subjects.

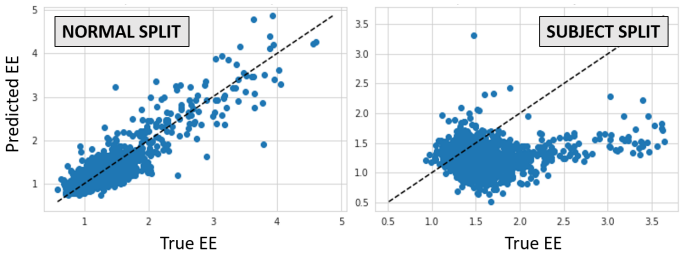

Figure 5. Predicted EE versus true EE. On the left, the neural net was trained on 80% of the data and tested on the remaining 20%. On the right, the neural net was trained on five subjects and tested on the remaining one.

IV. Proposition of a rectified model

A. Motivation

Our overall goal is to predict EE for different subjects. In the previous part, we have shown that classical machine learning algorithms can easily overfit on subject characteristics, leading to non-generalizable models. Moreover, even if we remove these dimensions from the data set, sophisticated models like random forest can still recognize a specific subject from its body variables average, as they depend on the individual. However, we can not really do without these "subject identifiers" (age, weight, average heart rate at rest...), because it seems that EE is highly dependent on them.

In order to develop a model that is not too prone to overfitting, but still takes into account that EE is subject-dependent, we made the following assumption: EE for subject S can be written as $EE_S(t) = \overline{EE}_S + \widetilde{EE}(t)$, where:

- $\overline{EE}_S$ is an average EE that depends only on the subject's characteristics at rest (individual base level of S)

- $\widetilde{EE}(t)$ is a variation over time of EE that can be linked to environmental and body changes, independently of the subject (of course this is an approximation, in reality the variations would also depend on the subject)

This assumption seems to make physical sense when looking at some EE profiles, like in figure 6. On this graph, we observe that for two different subjects, the average EE ($\overline{EE}_S$) is different, however the variations due to environmental changes ($\widetilde{EE}(t)$) are very similar. We will see in the next section how this assumption is useful for developing a better model.

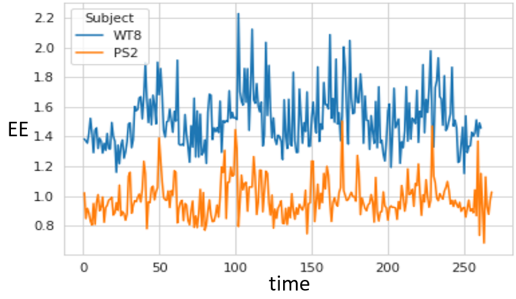

Figure 6. Energy expenditure for two different subjects, on the same protocol.

B. Description of the method

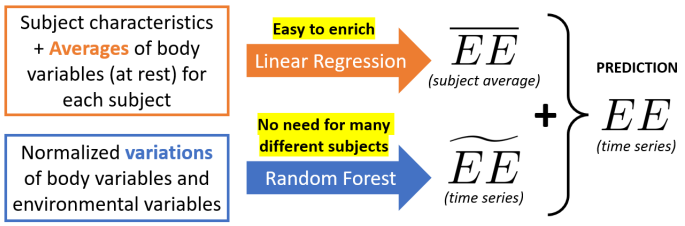

Figure 7. Functional schematic of the rectified method

Given the previous decomposition, we aim at predicting the two terms separately: the average $\overline{EE}_S$ with a simple model, using the subject's characteristics, and the variation $\widetilde{EE}(t)$ with a more complex model (e.g. random forest), using the variables variations. The advantage compared with the previous method is that we now isolate the subject-dependent part that used to overfit. Therefore, even though it does not fix the issue of having only 6 different subjects, it simplifies the way it can be solved, because this lack of data only impacts the prediction of $\overline{EE}$. Now, to enrich the new model, simple measurements can be made on more subjects (evaluating their EE base level in function of simple characteristics), which does not require complex protocols (just resting). This data can thus be conveniently collected on a large scale, and is fast to pass on the model (simple linear regression). On the other hand, the random forest part that predicts $\widetilde{EE}$ does not require many subjects, but only many data points from more complex protocols. This is already the case of our current data set, so this part would theoretically not require further training.

C. Implementation

First, we need to compute the references for the body variables for each subject, To that end, we use as references the protocols 3a and 4a, which cover all subjects and include no physical effort nor temperature changes. For each subject, we compute the average over these protocols, and we substract it over the whole dataset. : $\tilde{X} = X - \bar{X}_{ref}$. The environment variable are standardized as before. Then, we split the data into two parts :

- the 'subject' dataset, with one data point per subject, comprising the subject characteristics and the average of the body variables for each subject,
- and the 'series' dataset, with one data point per subject, protocol and timepoint, comprising the standardized environment variables and the body variables with mean removed.

We then fit a ridge regression on the subject dataset with $\overline{EE}$, and a random forest with 115 trees on the series dataset with $\widetilde{EE}$.

D. Results

The prediction of $\overline{EE}$ is difficult to evaluate, as we only have 6 subjects. We can still report that the average relative variation $\frac{|\overline{EE}_S - \widehat{\overline{EE}_S}|}{\overline{EE}_S}$ over subjects is of $9 \pm 3\%$ (95% confidence interval).

We score the 'series' model in two ways. In the leave-one-out procedure, for each subject, we train the model on the other subjects and score it on the selected subject. In the mixed procedure, we use a usual 5-fold split, but for each split we score the model separately on each subject in the test set and take the average as the score of the model on the split. This procedure allows us to meaningfully compare a model trained on all subjects and a model trained on a subject it has not seen before. The average R^2 is of 0.39 in the leave-one-out procedure, and of 0.78 in the mixed procedure.

This means that the series model still overfits, as it has a lower score when tested on a subject it has not seen before, which means that subtracting the average of the body variable is not enough to obfuscate the identity of the subject to the model. We also tried other forms of normalization of the body variables ($\tilde{X} = \frac{X - \bar{X}_{ref}}{\sigma(X_{ref})}$, $\tilde{X} = \frac{X - \min_{ref}(X)}{\max_{ref}(X) - \min_{ref}(X)}$) but the model still overfitted and its R^2 on leave-one-out protocol was worse.

Finally, when combining the two models, we obtain an average $R^2 = 0.30$ on the leave-one-out protocol. This is quite bad, but still much better than the negative R^2 we obtained without the rectified two-parts model.

V. Conclusion

We have shown that, with a reduced set of 15 variables, different machine learning algorithms were able to accurately predict energy expenditure for subjects present in the dataset.

However, we also have shown that, with only the 6 subjects present in the experiments, even simple algorithms such as a linear regression overfit and are unable to predict energy expenditure on new subjects.

We then devised a pipeline to predict the energy expenditure while decreasing overfitting, by separataing the data into subject-dependent and time-dependent data, and fitting a different model on each part. We show that this method reduces overfitting, although its performance is still quite low on new subjects.

To improve this model, we could use new reference data from simple experiments on numerous people, so as to improve the subject-dependent model. We also could improve the method used to decorrelate the data from the corresponding subject, as well as extend the model of dependency of energy expenditure to subject, which only takes into account its average.

References

[1] "Energy performance of buildings directive factsheet". In: *European Commission website* (2019). URL: https://ec.europa.eu/energy/content/factsheet-energy-performance-buildings-directive_en.

Used external Python librairies

[2] Charles R. Harris et al. "Array programming with NumPy". In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2.

[3] J. D. Hunter. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[4] The pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134.

[5] Michael L. Waskom. "seaborn: statistical data visualization". In: *Journal of Open Source Software* 6.60 (2021), p. 3021. DOI: 10.21105/joss.03021. URL: https://doi.org/10.21105/joss.03021.

[6] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[7] Pauli Virtanen et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[8] Skipper Seabold and Josef Perktold. "statsmodels: Econometric and statistical modeling with python". In: *9th Python in Science Conference*. 2010.