Predictive Modeling for Sleep Efficiency Using Machine Learning: An Evidence-Based Approach

Quan Tran

Aalto University, School Of Science
Espoo, Finland

1 Introduction

Sleep efficiency is the measure of how effectively you utilize a person's time in bed for sleeping, represented as a percentage of time asleep to time in bed. Maximizing sleep efficiency is important because it indicates better sleep quality, feeling more refreshed during the day, efficient time management, reduced sleep problems, and enhanced physical and mental health [3]. This research aims to predict the value of sleep efficiency as a ratio of the time a person spends asleep to the total amount of time that person spends in bed.

In Section 2, the report will introduce the dataset, put the problem in the context of Machine Learning, and define the project data points, features, and labels. Section 3 will discuss the data preprocessing procedures and the method chosen for the project. Finally, Section 4 will discuss the chosen method based on loss value and evaluate its performance on the test set.

2 Problem Formulation

2.1 Dataset and data points

The dataset we are using is obtained from the website Kaggle [2]. It is part of a study conducted in Morocco by a group of Artificial Intelligent students from the National Higher School for Computer Science and Systems Analysis (ENSIAS), Morocco. The dataset has 452 rows, for 452 different test subjects, and each row records the statistics for one night of sleep. **In this project, each data point also represents one night of sleep for one test subject.**

2.2 Features and Labels

As mentioned before, the goal of this project is to predict the value of sleep efficiency as a ratio of the time a person spends asleep to the total amount of time that person spends in bed, using supervised machine learning.

In this project, the **feature** values of one data point would be the test subject's the number of awakenings during bedtime, alcohol consumption 24 hours prior to bedtime, smoking status, the exercise frequency each week, the time they went to bed and woke up, age, caffeine consumption, and sleep duration.

The **labels** represent the test subject's **logit transformation** of sleep efficiency for that night of sleep. The sleep efficiency value, denoted as θ, is a ratio ranging from 0 to 1. In practice, machine learning models often provide more accurate results when we transform the sleep efficiency θ to the logit scale. This transformation involves using the logit function, which is defined as $\log\left(\frac{\theta}{1-\theta}\right)$.

By performing predictions on $\log\left(\frac{\theta}{1-\theta}\right)$ instead of θ itself, we expand the value space from the range [0, 1] to the entire real number line $(-\infty, \infty)$.

The data used in this project is numeric.

3 Method

3.1 Feature selection process

The feature selection process involves both domain knowledge and data-driven methods. Features known to be associated with sleep efficiency, such as the number of awakenings during bedtime, alcohol consumption 24 hours prior to bedtime, smoking status, the exercise frequency each week, the time they went to bed and woke up, and age, are included based on domain knowledge. Additionally, a correlation analysis is performed to identify other features in the dataset that are strongly associated with the logit-transformed sleep efficiency label. Features with low variance or low correlation with the label are excluded to reduce dimensionality and improve model performance.

By visualizing the data with scatter plots and correlations matrix, alcohol consumption, exercise frequency, awakenings, smoking status, bedtime, wakeup time, and age exhibit more noticeable correlations to the labels than the others.

The dataset is initially static, collected at the start of the project.

3.2 Polynomial Regression

Two different **methods** will be explored in this project: **Polynomial Regression** and **Multi-layer Perceptron**. **Polynomial Regression** is chosen due to its simplicity and efficiency with problems in which there are highly nonlinear relations between features and the label. It assumes a polynomial relationship between the features and the log odds of the outcome, which is a reasonable assumption given the nature of the features in the dataset [1].
For Polynomial Regression method, the hypothesis space is the polynomial predictor maps

$$\boldsymbol{\mathcal{H}}(n)_{\text{poly}} = \Big\{h(w) : \mathbb{R} \to \mathbb{R} : h(w)(x) = \sum_{j=1}^{n} w_j x^{j-1}, \\ \text{with some } \mathbf{w} = (w_1, \ldots, w_n)^T \in \mathbb{R}^n\Big\}$$

as the visualizations (see Appendix) have shown that there are polynomial relations between features and the label. In this project, I will implement polynomial regression by first applying the feature map

$$\phi(x) \mapsto x := (1, x, \ldots, x_{n-1})^T \in \mathbb{R}^n$$

to the scalar feature $x^{(i)}$, resulting in the transformed feature vector

$$x(i) = \phi(x^{(i)}) = (1, x^{(i)}, ..., ((x^{(i)})^{n-1})^T \in \mathbb{R}^n$$

and then applying linear regression to the transformed feature vectors $\boldsymbol{x}^{(i)}$.

3.3 Multi-layer Perceptron

While **Multi-layer Perceptron**, in addition to capturing non-linear relationships in data effectively, can also generalize well to unseen data, reducing the risk of overfitting [1].

For the Multi-layer Perceptron (MLP) method, the hypothesis space corresponds to the signal-flow representation of an Artificial Neural Network (ANN) which maps the feature vector $\boldsymbol{x} = (x_1, x_2)^T$ to a predicted label $h^{(\boldsymbol{w})}(\boldsymbol{x})$ achieved from all feasible options for the weights $\boldsymbol{w} = (w_1, ..., w_9)^T$. The chosen activation function used in the ANN is rectified linear unit (ReLU) $\sigma(z) = \max(0, z)$ as this is one of the popular choices. The figure of the ANN representation can be found on page 101 of the course book http://mlbook.aalto.fi [1].

3.4 Loss function

The loss function for both models will be the Mean squared error (MSE) loss function. This loss function is suitable for Polynomial Regression and Multi-layer Perceptron problems like this ones, as it penalizes larger errors more heavily, which is often desirable when minimizing the overall deviation between predicted and true labels [1]. Plus, the dataset does not contain outliers, which makes the MSE a reasonable choice for the problem.

3.5 The training and validation set construction

The data is first splitted into training-validation (80%) and testing (20%) subsets. Then the training-validation set are used to train the model by using K-Fold cross-validation, with K = 5. This design choice is made to provide a balance between overfitting the training set and the reliability of the validation error when the amount of dataset is relatively limited.

4 Result

4.1 Training and validation error

The next step is to choose a hypothesis map from each hypothesis space discussed based on the approximation error.

Polynomial Regression By using the Mean squared error loss function, we can compare the performances of various predictor maps from the hypothesis space. Based on the training and validation errors (Fig. 2), the best degree of choice for polynomial regression is 2^{nd} degree as it inccured the smallest validation error while having a decent training error.

	poly degree	linear train errors	linear val errors
0	1	0.018142	0.019227
1	2	0.011365	0.018887
2	3	0.006159	0.119795
3	4	0.000011	332.192935

Fig. 1. Polynomial Regression training and validation loss

Multi-layer Perceptron By using the same loss function, we can also compare the performances various ANNs with different numbers of hidden layers within the same hypothesis space. Based on the training and validation errors, the most suitable number of hidden layers is 17.

	num hidden layers	mlp train errors	mlp val errors
0	6	0.062957	0.059371
1	7	1.835022	1.954107
2	8	0.059866	0.059730
3	9	0.065475	0.063908
4	10	2.261453	2.359468
5	11	0.110157	0.112533
6	12	0.059813	0.061742
7	13	0.054722	0.056702
8	14	0.064332	0.069349
9	15	0.057885	0.058499
10	16	0.055602	0.057207
11	17	0.053440	0.056502
12	18	0.054191	0.056637

Fig. 2. Multi-layer Perceptron training and validation loss

4.2 Methods comparison and test error

The validations errors of the 2^{nd} Polynomial Regression model (0.014330) and the 17-hidden-layer Multi-layer Pereptron (0.057) suggest that the Polynomial Regression is performing better. Hence, it is chosen as the final model.

To understand the accuracy of the final model, we have to retrain the selected model on the entire training dataset (including the validation set) with the chosen hyperparameters. Then, we assess the performance of the selected model on the separate test set to get an unbiased estimate of its generalization performance using selected loss function (MSE). The test set is constructed using the function train_test_split from the scikit-learn library as already mentioned.
After retraining the model, the calculated test error of the selected model is 0.027.

5 Conclusion

In conclusion, we also need to apply the inverse logit-transformation to the label to revert it back to the [0,1] scale in the last step.

For our Machine Learning problem, it turned out that the Polynomial Regression model with 2^{nd} degree performing better than the Multi-layer Perceptron. However, there is a considerable gap between the validation error (0.014330) and the test error (0.027), as the test error almost doubles the validation error. This suggests the possibility overfitting in the model of the final model.

Looking forward, we acknowledge that there is room for improvement. Future directions may involve the collection of more extensive training data to address overfitting concerns and exploring alternative features or models to enhance predictive accuracy. Furthermore, it is also beneficial to collect sample from test subjects of various ethnics as each ethnic may have a different internal biological clock. These considerations pave the way for further advancements in the field of sleep efficiency prediction, ultimately contributing to improved physical and mental well-being for individuals.

References

1. Jung, A., 2022. Machine Learning: The Basics. Singapore: Springer
2. EQUILIBRIUMM. (n.d.). Sleep Efficiency Dataset. Kaggle. Retrieved September 21, 2023, from https://www.kaggle.com/datasets/equilibriumm/sleep-efficiency
3. Reed, D.L. and Sacco, W.P. (2016) Measuring sleep efficiency: What should the denominator be?, Journal of clinical sleep medicine: JCSM: official publication of the American Academy of Sleep Medicine. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4751425/ (Accessed: 09 October 2023).