

**Course:**

**Predictive and Descriptive Learning**

Half-term Activities Report

**Developing Machine Learning models**

# CONTENTS

[1. Type of machine learning 5](#_Toc85874100)

[1.1. **Self-Supervised learning** 5](#_Toc85874101)

[1.1.1. **What is Self-Supervised learning** 5](#_Toc85874102)

[1.1.2. **How it is used** 5](#_Toc85874103)

[2. DEVELOPING Machine learning models 6](#_Toc85874104)

[1.1. Problem description 6](#_Toc85874105)

[1.2. Machine Learning approach 6](#_Toc85874106)

[1.2.1. Regression 6](#_Toc85874107)

[1.2.2. Classification 7](#_Toc85874108)

[3. Data description 8](#_Toc85874109)

[4. EXPLORATORY data analysis (EDA) 9](#_Toc85874110)

[5. Experimental setup 10](#_Toc85874111)

[6. machine learning models 11](#_Toc85874112)

[7. RESULTS 12](#_Toc85874113)

[8. CONCLUSIONS 13](#_Toc85874114)

[9. REFERENCES 14](#_Toc85874115)

# LIST OF FIGURES

Figure 1 5

# Type of machine learning

## Self-Supervised learning

### What is Self-Supervised learning

Self-supervised learning is useful when having a lot of labels data is not possible. For example, it’s very used for NLP (natural language processing) It can be related to unsupervised learning because data are not labeled but the difference is that we don’t want the model to do cauterization. Indeed, it’s not possible to have all sentences that can exist labeled because there are too many combinations that we can make with words. So self-supervised learning is used to make the model understand the mechanism of language such as correlation between words. A model trained like this with a big amount of data can then be used or trained to complete a sentence with a missing word or a text with a complete sentence. It’s also can complete image with pixel or video with frame.

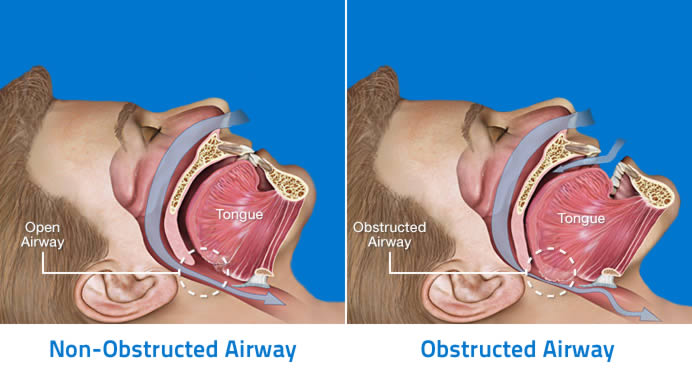
### How it is used

The huge improvement in NLP AI area is dues to this way of learning. A real application of this type of training is the algorithm of NLP called "Bert" and released in 2018 which has beat all other algorithms.

# DEVELOPING Machine learning models

## Problem description

**Obstructive Sleep Apnea (OSA)** is a disorder in which breathing is repetitively interrupted during sleep due to collapse of the upper airway. it is the most common sleep-disordered breathing.



OSA illustration 1

Figure 1 – OSA Illustration

**AHI (Apnea-Hypopnea Index)** measures the severity of the disease. it represents the number of times the patient stops breathing per hour.

The disease is prevalent as 100 million individuals worldwide have OSA but only 10%-20% are diagnosed because of the difficulty of its diagnosis. In fact, it implies to sleep one night at hospital with a lot of sensors and the waiting list for this can be very long depending on the country (1 year in Spain).

Because this disease is dangerous and can lead to death, it can be relevant to predict the AHI of a patient with data that are easy to get from an application (weight, height, age, gender, cervical perimeter, speech recordings, facial image or mouth images). This will be the aim of the project

## Machine Learning approach

We will try to solve the problem using both regression and classification approaches. The aim of the regression approach will be to estimate the AHI of a patient using the data that we collect from an imagined app (Actually the data we will use has been collected by a medical center). The classification approach will be ordered in the same way as the regression on but the prediction we expected will not be the same. Instead of estimating the exact AHI, we want our ML model to put a patient into two classes which will be normal (AHI < 10) or severe (AHI > 30).

### EDA

We will first focus on the clinical data which are easier to use and can be treated by simple ML algorithms instead of a deep neural network which will be required to deals with data as image or speech recordings.

For both of those approach the first things to do in ML is to get the data, analyze it and transform it because it is important, even more in our medical case) to understand our data and find out which features can be relevant to predict the AHI. It avoids constructing a model as a black box that we can’t explain.

For this step I first want to have just one dataset instead of the two givens so the analyze of transformation will be easier. After this I also want to have only feature in my dataset because ML algorithm can only deal with number as word or letter doesn’t mean anything in term of mathematics. After having a clean dataset, I tried to find out which feature can be useful to predict the AHI of a person while using statistical tool and information I can get about this disease.

### Train, test method

Next step was to construct, train and test different model. Because we don’t want our model to do overfitting (try to fit perfectly to the dataset instead of getting a good understanding of it which can be used to other data), it is highly recommended to separate our dataset into 2 which will be used as a training and as a testing dataset.

To test my different models, I used different metrics depending on the approach. Indeed, since the output are not the same, we have to us different method to test our models. I used the MSE and the R2\_value to test my regression models where I used confusion matrix and metrics such as accuracy, sensitivity and specificity to classification models.

### Machine Learning Models

For dealing with the regression approach, I first choose to use the simplest model which are the linear the regression

# Data description

The clinical data that we have access to are split into two CVS file:

Une image contenant table

Description générée automatiquement

Figure 2 - CSV 1: Clinical Data

Une image contenant table

Description générée automatiquement

Figure 3 - CSV 2: Gender and Age

Therefore, the first thing to do after import those into python is to fusion them into one panda data frame. To process this operation, we first must remove the incomplete row of our data so we can fusion them correctly.

We now have our only data frame having the columns “AHI, Age, Weight, Height, Cervical, Gender”.

Une image contenant table

Description générée automatiquement

Figure 4 – Joined Data frame

We are now able to analyze them in order to now which features will be relevant to predict the AHI or classify the patient.

# EXPLORATORY DATA ANALYSES (EDA)

The first thing to do when we have data like this is to think if we can create some new one, more relevant, from them. This procedure is called feature engineering and it have several benefits. The first is that we reduce the number of columns we want our model to use and therefore we are simplifying our ML problem and will be more able to understand what our model are doing. The second thing is that as human, we know the signification of our data and the impact they might have on the problem, but the model doesn’t. It’s good to help him by giving to him the features that are more susceptible to be helpful because we know they have a real impact on what we are wanting to predict. For example, according to our context and the info we have about the disease, we know that the BMI is an index more relevant than having the weight and the height because what matter here is a metric that represent the obesity of a people.

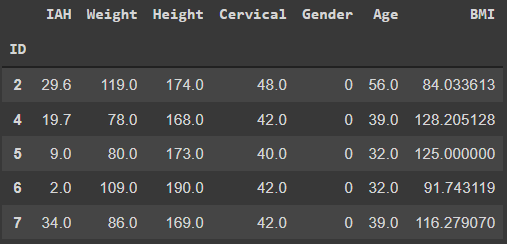


Figure 5 – Adding of BMI

After we did all, we can do to clean our data and try to improve them using our knowledge, it’s time to use statistical tools which are depending on the approach we use.

For built my regression model, I used scatter plot to see if I can find some obvious relation between data and correlation matrix too detect how features have an impact on the AHI. As it is a complex problem, I didn’t find any evidence, but I can at least know which data are more relevant to predict the output.

In order to treat the classification problem, the tools I used was bow plot, confusion matrix and p value.

Using those statistical methods, I found in both case that the more correlated data are the BMI and the age, but I decided to use all of them at the beginning and then test if I can improve the accuracy by removing some useless data.

# Experimental setup

After splitting our data, we can train our model and test it by doing prediction, but we are still not able to evaluate our model. Indeed, we still have any way to evaluate if predictions have been and how far have, they been good or not. The is why we need to use some metrics but as predictions are not from the same type if we are doing regression or classification, we must use different metrics depending on the approach.

To evaluate the predictions of my regression models, I used MSE and the R2 value which compare the MSE with the one we would have if we predicted the average of AHI for every input (simplest way to treat to problem).

To evaluate the predictions of my classification models, I used the accuracy and confusion matrix. Accuracy tells us in what proportion my models get right on the classification while the confusions matrix shows also the false positive and true negative. Confusion matrix is more complete than accuracy to evaluate predictions because we don’t want our model to predict a “severe” AHI on healthy people.

It is now possible to train, test and evaluate the different models we want to try to solve the problems of regression and classification.

Once I get a good understood of my data, I was now able create my models and try them in order to get the best result. But the question is how do I evaluate my model? Like I said before, to test a model it is important to make him doing prediction from other data that those used to train it. It avoids having good result and think our model is good whereas it is just fitting our training data but can’t predict well with others. A model like this is useless because we want something that we can use in general, with any input. A thing to do is to split our dataset into to smaller one which will be use respectively for training and testing. In this case, I choose the proportion 80-20, because I want to keep enough data to train my model correctly

# machine learning models

## Regression:

linear regression (simplest, will help to evaluate more complex model)

SVM : Because we have many features and therefore many dimension (worst result)

Decision tree: non-linear problem , still understandable (same result)

Random forest : very non-linear problem (get better result)

## Classification:

logistic regression : simplest

KK-N : don’t need to find a function, just using data

Random forest : very non-linear problem (get better result)

# RESULTS

As expected, our models, even the most complex, didn’t get very good result. It’s explicable by the fact that the data we are using are not very correlated to the AHI. We might could have better result using more complex data such as speech recording or facial images and more complex as a deep neural network.

# CONCLUSIONS

During this project we saw how to implement a procedure to deals with a ML problem.

We first saw how to treat and analyse the data in order to use feature as most relevant as we can so that our models can train on data which have a real impact on what we are trying to predict or describe.

In a second order we saw how exploit our data to train and test our models in the best way. We also saw some examples of metrics that can be useful to know how good our model is doing.

During the third part I tried to explain how we choose an algorithm based on the type of problem we have (the basic model is often a good point of start before testing more complex one). We could have a good illustration of the black box problematics consisting in the fact that we something are given more importance in understand what our ML model is doing than having best result (especially in medical case for instance). Therefore, it’s not so easy to find the best compromise between the complexity and the efficiency of our model).

Finally, we saw that ML is not magic and even is we use the best model we won’t be able to get good result if we are not using the appropriate data.

# REFERENCES