2021

AI in Enterprise Systems – Final Project Report

TITLE: HEART DISEASE PREDICTOR

AUTHORS: RAMON VILARINS

UMUTCAN ASUTLU

1. **Introduction**

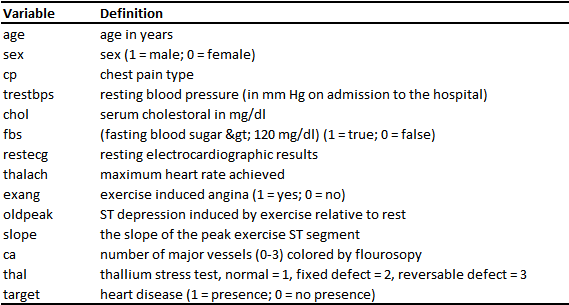
Heart disease refers to several types of heart conditions, varying from coronary artery disease to a heart that does not pump well, for instance. Nowadays, it is common sense that smoking, unhealthy diets and lack of physical exercises increase the risk for having cardiac diseases. With regard to its signs and symptoms, most of the time, it can be suspected through conditions such as: chest pain, shortness of breath and pain in the arms.

According to the Centers for Disease Control and Prevention, about 610,000 people die of heart disease in the United States every year. Moreover, it estimates that heart diseases cost $219 billion to the country annually. This includes spending with health care services, medicines and lost productivity due to deaths. In Canada, the situation is not different, and every hour 12 Canadian adults age 20+ with diagnosed heart disease die.

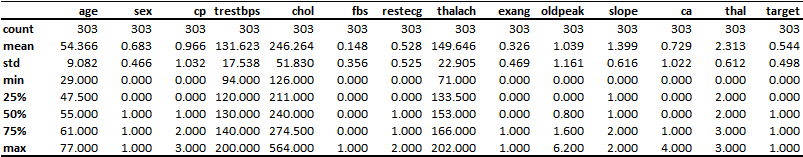
Using a dataset provided by the Cleveland Heart Disease, this work aims to apply machine learning techniques to detect hidden patterns and to predict the presence of heart disease. By early detecting this abnormality, it will be possible to act preventively, reducing the risk of heart attacks and heart failures in the future.

1. **Dataset**

Our dataset is composed of 303 entries and includes a total of 14 attributes. In Table 1 we present the definition of all 13 features and the target variable. Except from the feature *oldpeak,* which is a floating-point type, all the remaining ones are integers values. It is worth noting that there are no missing values in the dataset.

**Table 1 - Variables Definition**

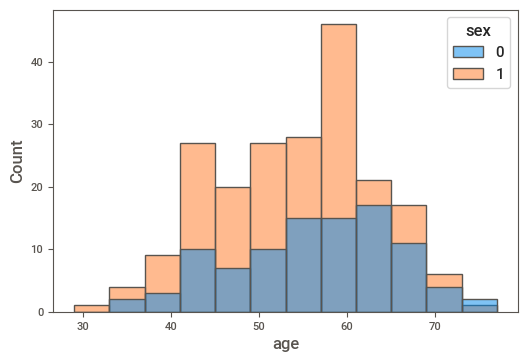
In Table 2 we have a statistical description of the dataset. It can be seen that the sample is aged between 29 and 77. Moreover, we observe that, on average, the resting blood pressure is 131.623 and chest pain is 0.966.

**Table 2 - Dataset Description**

1. **Exploratory Data Analysis**

Observing Figure – 1, we can verify that the data contains mores male than females. That is, from a total of 303 entries, 96 are females and 207 males.

**Figure 1 - Age Distribution by Sex**

****

From Figure 2, we note that the number of patients who has heart disease (target = 1) is 165, being 72 females and 93 males. Therefore, the percentage of females with cardiac condition is much higher than the percentage of males.

**Figure 2 – Heart Disease by Sex**

Checking the relationship between age, sex and heart disease in Figure 3, we found that females with positive cardiac condition are on average older than males. That is, according to our data, while most part of the men with heart disease are around between 40 and 60 years, women are between 40 and 65 years old.

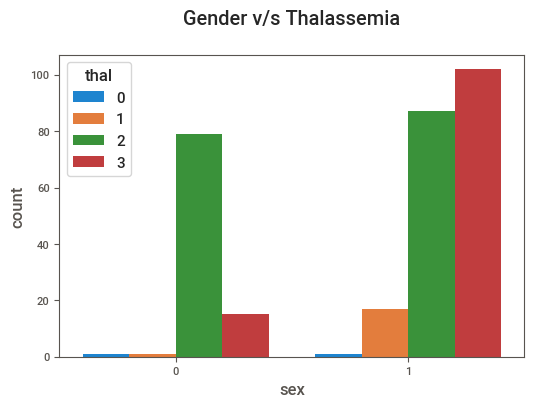
**Figure 3 - Heart Disease by sex and age**

From Figure - 4, one can see that there seems to be a negative correlation between maximum heart rate and age. In addition, there are indications that heart condition is associated with higher heart rates.

**Figure 4 - Maximum Heart Rate by Age**

People with thalassemia can get too much iron in their bodies, either from the disease or from frequent blood transfusions. By its turn, too much iron can result in damage to the heart. So, from Figure – 5, we observe that the number of males with Thalassemia (reversable defect, thal = 3) is greater than the number of females.

**Figure 5 - Gender Versus Thalassemia**

****

A typical symptom of heart attack is chest pain. While examining Figure – 6, we have clear indications that confirm this association. That is, correlation coefficient between chest pain and the target variable is 0.43. Additional examining indicates that maximum heart rate has also a very strong positive correlation with heart condition, reaching 0.42.

**Figure 6 - Correlation CoefficientsInterface gráfica do usuário, Aplicativo

Descrição gerada automaticamente**

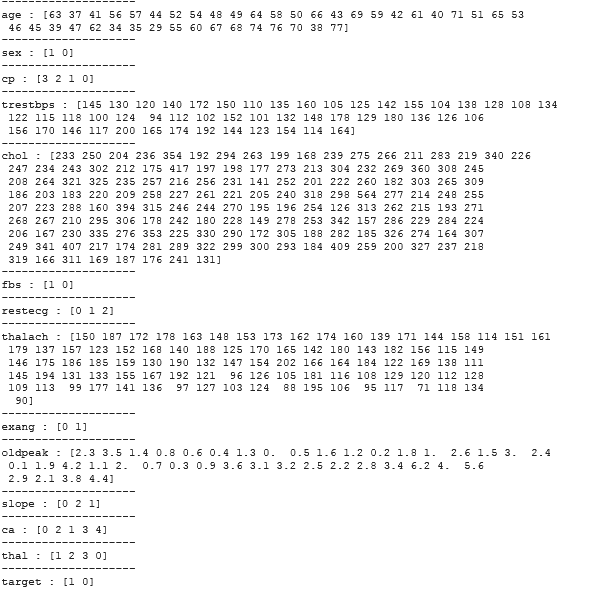
1. **Modelling**

As there is a process of training our machine learning model on a labelled dataset, that is, a dataset in which the target variable is known, it is a supervised model. Moreover, considering that the goal of the model is to predict discrete values, we have a classification problem. So, logistic regression, decision tree and KNN are initially good options of algorithms to train the model.

The features in our data set are at varying scales. For example, Table 3 shows that variables *chol* and *oldpeak* are on completely different scales. Thus, depending on the machine learning model being used, this situation requires some form of data transformation, such as normalization or standardization. Normally, this is necessary for algorithms that calculate the distance between data, like KNN and PCA.

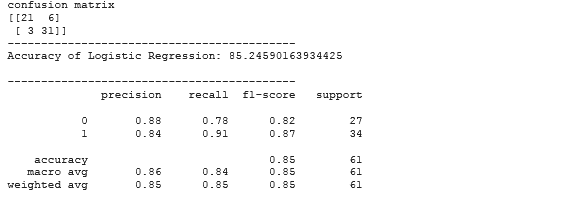
Nonetheless, standardization is not required for logistic regression nor for decision tree or random forests. As these algorithms depend on rules and not on distance, they are not affected by any monotonic transformations of the variables. As we are using KNN, Logistic Regression and Decision Tree, we opted for standardizing our variables.

**Table 3 - Feature Engineering**

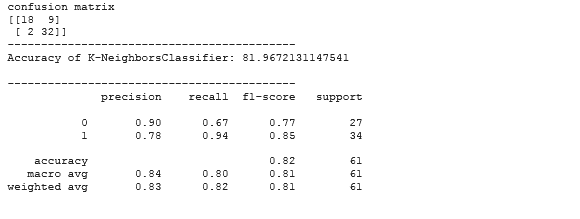


Logistic regression is a linear model and is considered one of the simplest classification models. Thus, it will be the starting point and will set a low bar for future models to surpass. To put it simply, it computes the probability of an event occurrence utilizing logit function where logit = log (p/(1 − p)) = log (probability of occurrence/probability of event not occurrence). In Table 4 we have the classification report of the logistic regression model. It can be seen that the accuracy level of the model is 85%.

**Table 4 - Logistic Regression Model Classification Report**

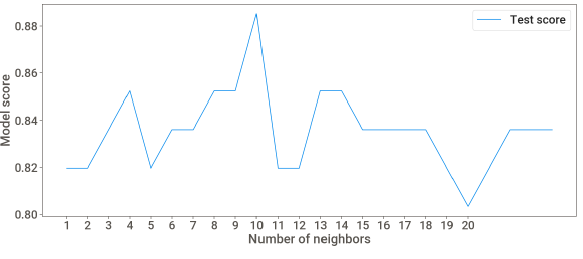


In Table 5 we have the classification report of the KNN model. It can be seen that the accuracy level of the model is 82%.

**Table 5 - KNN Model Classification Report**

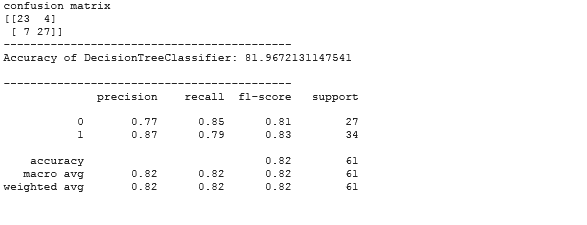
In Figure 7, we show the level of accuracy of the KNN model for different numbers of neighbors. The highest accuracy is reached when the number of neighbors is equal to 10.

**Figure 7 - KNN Model Score**



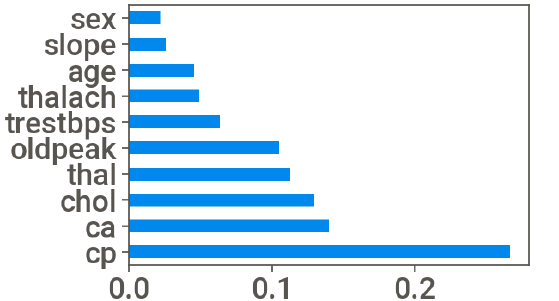
In Table 6 we have the classification report of the Decision Tree model. It can be seen that the accuracy level of the model is 82%.

**Table 6 – Decision Tree Model Classification Report**



In Figure 8, we can visualize the feature importances of the decision tree model. It can be seen that *cp*, *ca* and *chol* are the three most important features.

**Figure 8 - Features Importance**



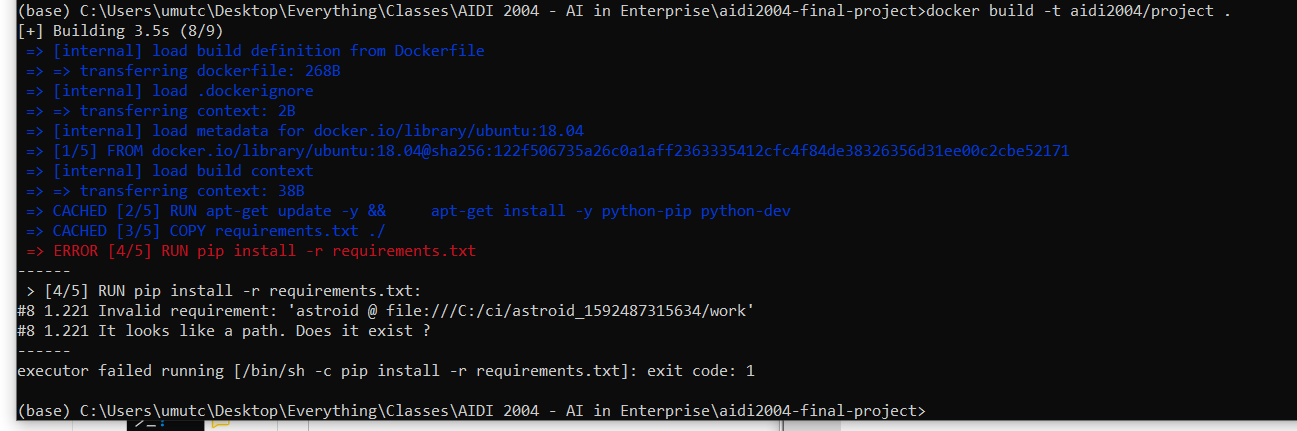
To conclude, we observe that the Logistic Regression model has the best level of accuracy, totaling 85.25%. So, it should be our first choice. However, it is also worth saying that the recall of the Logistic Regression model is 91% while the recall of the KNN model is 94%. Therefore, as in this case it is particularly important to minimize the number of patients with cardiac condition who are classified as a healthy person, we should consider doing further analysis with the KNN model as well.

1. **Deployment**

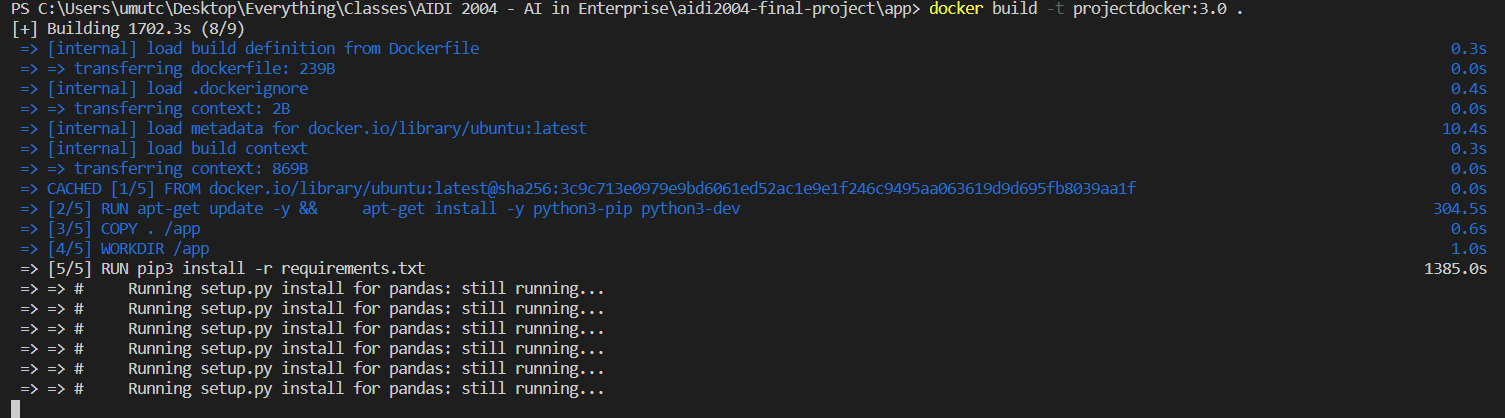
Deployment is done in two phases. The first one is GitHub deployment. Throughout the project GitHub is used effective to maintain communication and development process between team members. The GitHub link for the project:

<https://github.com/umutcanasutlu/aidi2004-final-project>

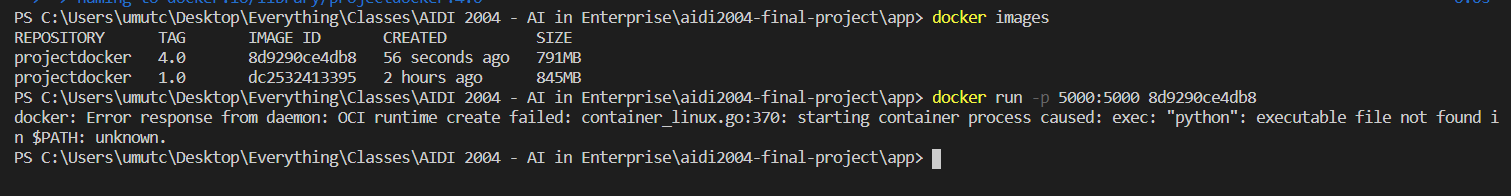
For the docker deployment, it was not as smooth as expected. We created the requirements file using “pip freeze > requirement.txt” but it did not work.



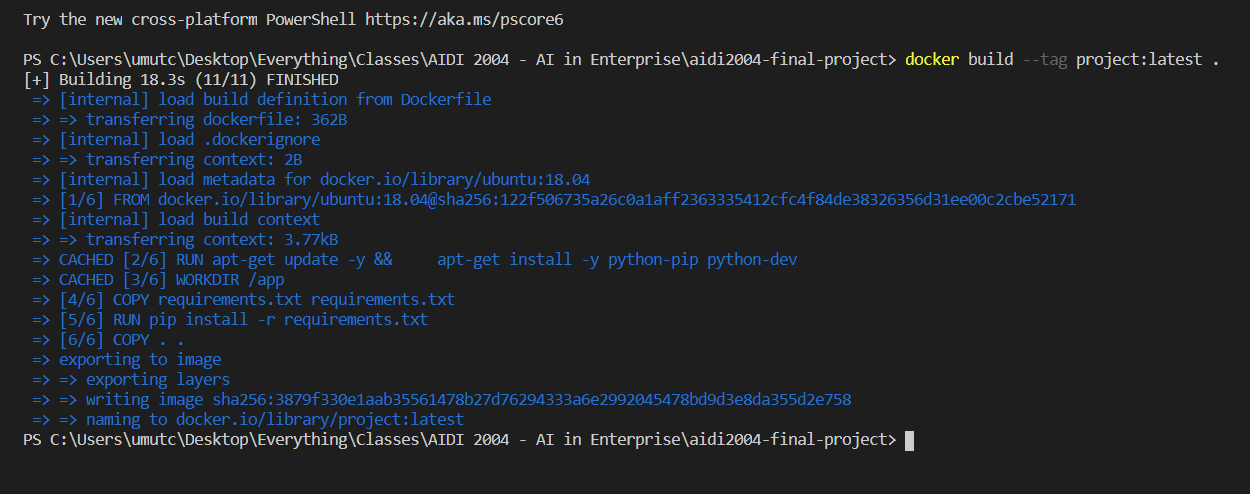
Afterwards we had to prepare al the requirements manually, it was an easy task but as we learned from our research as well, Pandas is not very integrable with docker to be used. Due to its inner files, it takes around 20 minutes for docker to be deployed. As in the figure below you can see it was still running for pandas after 1385 seonds.

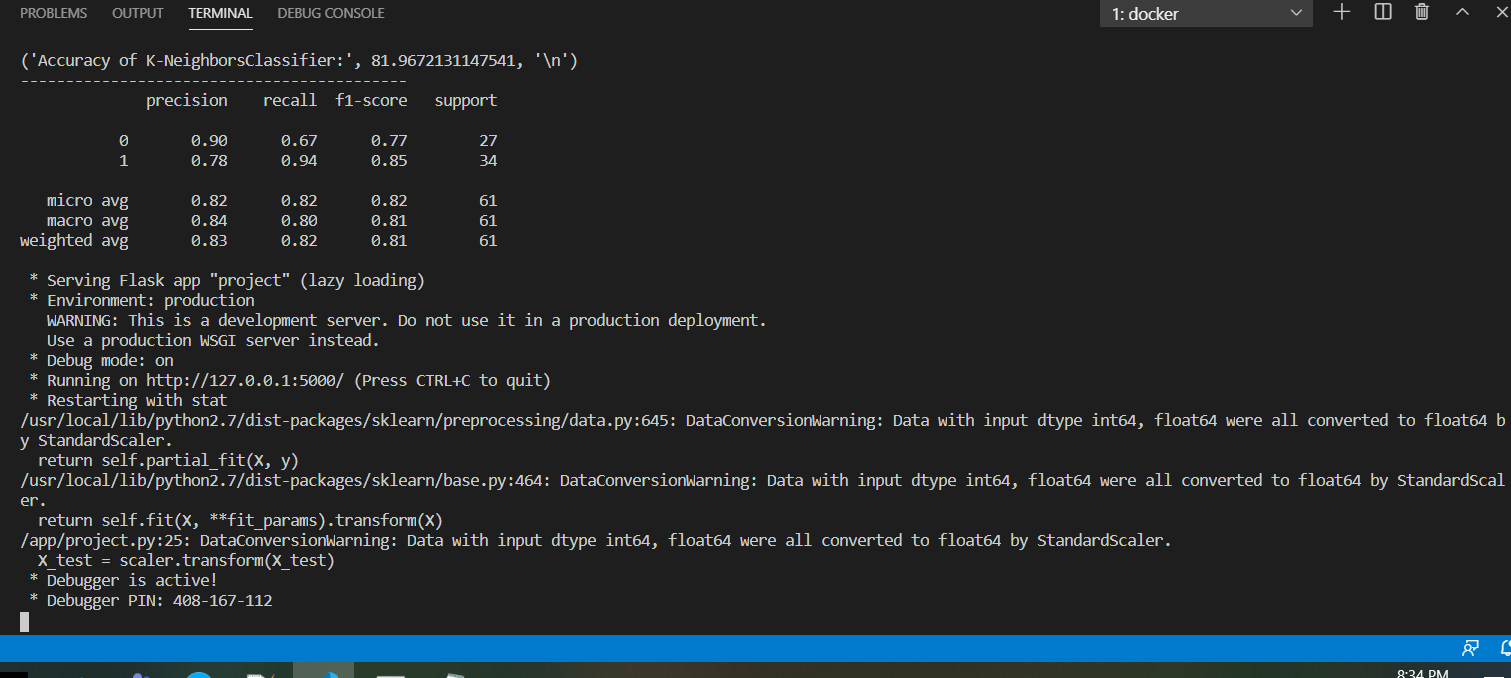


After many attempts with docker with 20 minutes build time it was obvious that we had not enough time to solve it with pandas. Then we had to eliminate the ‘pandas’ library. To do that we had to use ‘joblib’ library from Sklearn. After that we implemented a Numpy solution which required minimum amount of external libraries, however; it was giving this error after building the image.



Finally we managed to build proper image that can be run without any further problems.





However; after this step, we got stuck with web app view.