Protein optimization in ER patients

Almog Amiga Dvir Sadon Oz Klingel

August 2021

1 Abstract

Nowadays, ER patients get between 1 to 2 grams of protein per kilogram per day, without considering their various factors like their medical past and the amount of days they stayed in ER. We aim to find a way to get the optimal amount of proteins for each individual considering those parameters and more. Our goal is to help find the optimal amount of proteins for patients and thus reduce the length of stay in the ER and most importantly, death rate.

2 Introduction

ER patients are often in critical condition and are in need of sustenance while in a coma or coma-like state. This necessity is provided in the way of protein and other essentials. In this paper we will focus on protein because this seems to be the most crucial one of them all. Presently, the amount of protein given to the patient is specified according to factors like BMI, and age of the patient, but is otherwise set for all patients. We tried to change that in this work using numerous data science methods to study the previous cases and learn from more factors the optimal protein amount for the specific patient. Specifically, we set out to differentiate between cases where the patient has a different type of condition.

3 Analyzing the data

When analyzing the data we went trough a few phases. Firstly, a dataset of possible conditions and their designation into groups was organized in a way where every condition was designated to a specific group of conditions. The data was separated into 5 groups including trauma, surgical and more. This data was then included in other datasets that were used in order to help the models to learn better. This approach was much easier for the models to learn from then specifying for each patient, the specific condition he came to the ER with. Its important to note that many patients had a number of conditions, some from different groups and that these groups were also divided into conditions that the patient had previous to coming to the ER and ones he came in with.

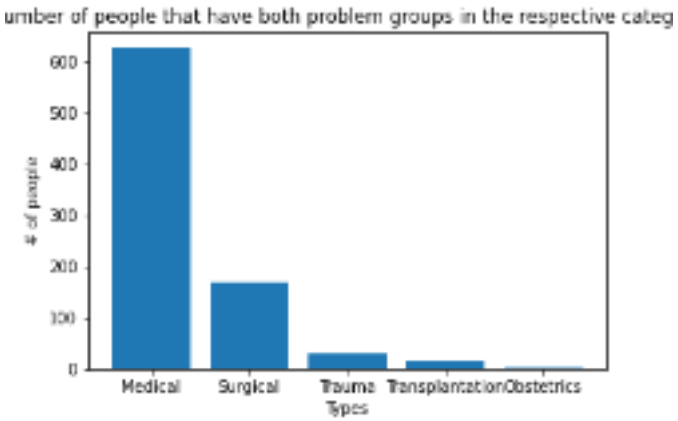

Figure 1: This figure shows the number of people in each of the categories described above.

The second part of data analyzing was trying to understand and how to work best with the main dataset used to train the models. This dataset contains, for each patient, the patient's BMI, age and feeding procedure among 67 other features that we were able to make use of. This includes features added by our team. We added a few features that represent the ratio between different features. This helps the model learn more easily. The most common and important example of this was a ratio between certain features and the patient's weight. We also tried to visualise the data in various ways in order to understand which features are more beneficial and whether we need to add more features and if so, in what manner.

Using our understanding of the data and PCA we also tried to reduce the number of dimensions our models work with. This improved the models and we found that when the number of components is 40 we get the picture for all the variables.

Our data is very imbalanced, roughly 80,20. Recall score is a useful measure of success of prediction when the classes are very imbalanced.Therefore we will look on recall with confusion matrix instead of accuracy.

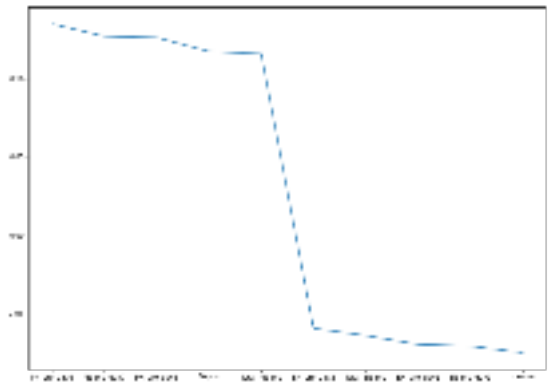

Figure 2: This figure shows the correlations of the most correlated features to the death variable in the dataset.

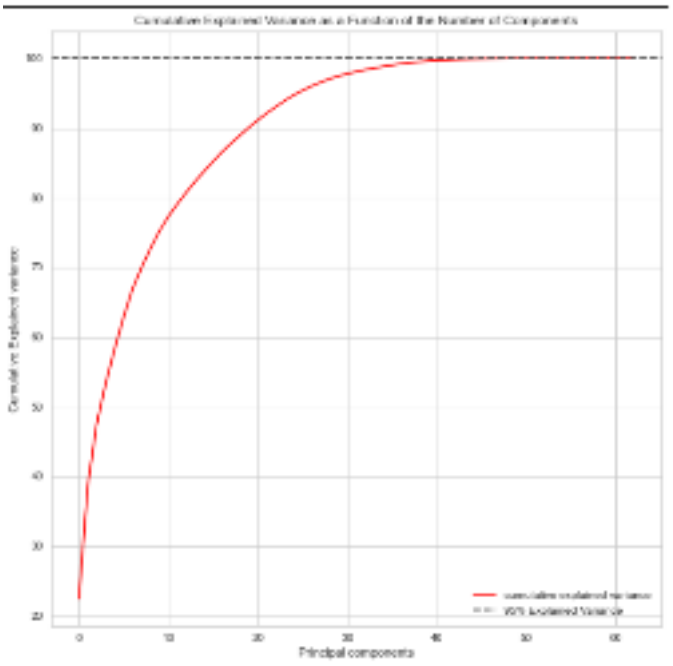

Figure 3: This figure shows that when reaching 40 features we get to 98% of the variance or more. so to avoid overfitting, we reduce the data to that amount of dimensions.

4 Methods

4.1 Decision trees

Decision Trees are considered to be one of the most popular approaches for representing classifiers. Researchers from various disciplines such as statistics, machine learning, pattern recognition, and Data Mining have dealt with the issue of growing a decision tree from available data. Therefore, our first model was Decision Tree. We used 'gini'. In order to avoid overfitting and poorly generalizing to new samples, we used pruning and found that tree depth 16 gives us better results.

4.2 KNN

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems.We tried the under-sampling approach using the EDITED NEAREST NEIGHBOUR method where we remove the samples close to the decision limit. The results were that out of 69 test samples, in 60 samples he answered correctly and only 9 were wrong in predicting the state of the patient. Although these results are good, the disadvantage of this method is that downloading samples that are close to the border are important samples that we want to study and know how to classify them. Despite this, given the fact that many records in a dataset are given irrational values (a protein-irrational pattern pattern like 3 days the patient did not eat), it seems that the results of this method should be considered because perhaps it implies many errors in the data processing phase. To reach reliable information.

4.3 SVM

support-vector machines (SVM) is a technique to find a hyperplane separating the points with large margin.For our model we put C= 0.01 and gamma= 1. When C is high it will classify all the data points correctly, also there is a chance to overfit. When gamma is higher, nearby points will have high influence, low gamma means far away points also be considered to get the decision boundary.

4.4 Neural network

Neural networks reflect the behavior of the human brain, allowing computer programs to recognize patterns and solve common problems in the fields of AI, machine learning, and deep learning. Our Neural network contained 2 hidden layers, the first contained 8 neurons and the second contained 3 neurons, both used RELU activation function. we used binary cross entropy as our loss function and adam optimizer. We also used under sampling due to our data being imbalanced.

5 Results

In this section we examine and show the results of the models that were run. We specify the recall mainly because the most important point of the model is to try to indicate whether a patient died and try to find those that did. This is reflected in the recall. Finding the dead patients is a tougher task because the data is overwhelmingly skewed toward alive patients. The accuracy is important as well but is less so and is easier to obtain. For this reason we show in this section the confusion matrix for each model's results.

5.1 Decision tree

The Decision tree Resulted in 93% recall.

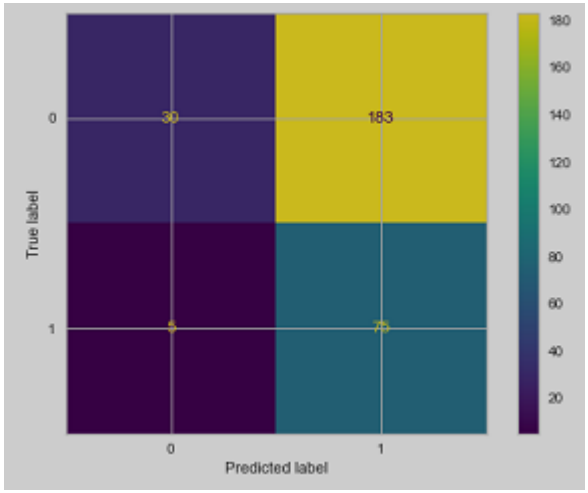

Figure 4: confusion matrix for our Decision tree model.

As we can see, this model is very safe, it has the least false negative.

5.2 KNN

The KNN model resulted in 76% recall.

```
[[33  0]
 [ 9 27]]
```

Figure 5: confusion matrix for our SVM model.

5.3 SVM

The SVM model resulted in 57% recall.

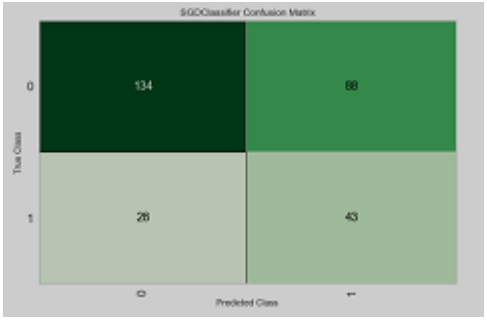

Figure 6: confusion matrix for our SVM model.

5.4 Neural network

The Neural network resulted in 60% recall.

```
array([[44, 17],
       [19, 38]], dtype=int64)
```

Figure 7: confusion matrix for our neural network model.

6 Conclusion

In this work we explore our attempts to improve the feeding policy for ER patients. We find that when trying to determine which patient dies, we get some good results. This work should be continued by other works using more data to train and test on, and use the methods that described in this paper and try to achieve better results and also try to give off a better form of output that specifies what is the exact amount of protein to feed the patient. Also, trying to do the same with other vital resources like sugars and alike. We also encourage other papers to experiment with neural networks with more depth and optimisation than done in this paper, as these may have better results.