Data Exploration on 4D PC-MRI Dataset

**Goal**

The research aims to find the set of features that allows to differentiate the patients that belongs to different pathologies.

**Dataset**

The dataset used for this experiment is the 4D PC-MRI data. This is a dataset which gives a detailed assessment of blood flow in one heartbeat. The values that are given are in terms of velocity and those are represented as vectors which are dependent on time in a 3D coverage. We have received the records of 42 patients from Universitätklinikum Leipzig. The dataset consists of 4 categories which are “Fallot”, “BAV”, “Healthy” and “Others”. Due to a smaller number of records, we categorized this into two groups: “Healthy” and “Unhealthy”, which resulted in 20 healthy patients and 22 unhealthy patients.

**Implementation**

As an initial step, the data pre-processing step was done to clean the data and combine it into a single dataset.

1. Data Pre-processing

The dataset was in the format of having a heading and its related dimensions. So, the headings had to be removed and the related dimensions like systolic, diastolic was extended with the name of the heading. The measurements which had timestamps and relative values where also processed to another feature in the dataset. After cleaning all the records , all records where combined to a single dataset which had 154 features excluding its pathology.

1. Feature Selection

* Correlation based Feature Selection

This algorithm is mainly based on linear correlation between features and then entropy of that feature. A feature is good if it is highly correlated to the label but not highly correlated to another features. Two advantage of using correlation:

* Remove features which has no correlation with the class
* Remove redundancy between selected features

But linear correlation is restricted to linear relations which should be numeric. So to find out other relations we use Entropy.

The information gain (Entropy) plays between 3 features, ie, if Y is more correlated to X than Z then if IG(X|Y) > IG(Z|Y).

Referred from: <https://www.aaai.org/Papers/ICML/2003/ICML03-111.pdf>

* Information Gain

Information Gain mainly corresponds to how much dependent is a feature to its class. The higher the information gain, higher is the dependence of that feature. It also reduces the noise in the data by avoiding certain features. This highest Information Gain feature is taken as the purest node to split.

* Chi-Square Test

This is also acting similar to Information gain, that is, it selects the features which are highly dependent to the target class. We will find the expected class by calculation and then compare it with the observed class. This is done for each feature. The higher the value, the higher is the dependence to the target. Since IG and chi square are almost same, the features selected are also same.

* Learners fused with feature selection

These are the features that are selected during the training of a particular model. It is integrated as a wrapper method along with model training.

There are 4 are search techniques that is used along with the model training. The wrapper methods are the methods which uses greedy search algorithms as they evaluate all possible combinations of the features and they select the best combination that gives a good result in that particular algorithm based on the classifier performance. These methods are computationally expensive, but this method is said to be one of best algorithm to find features.

The method of searching features are

* Random

Here the specified number of features are randomly drawn with a probability membership. A feature is included in the set and has a probability are Bernoulli distributed. A Bernoulli distribution is the probability distribution of a random variable which takes a value 1 or 0 according to mentioned probability.

* GA

This is a Genetic Algorithm which selects features according to mu and lambda depending on the comma settings. The comma settings are responsible for selecting the new set of size “mu” out of “lambda>mu” offspring. From this “mu” set, the new “lambda” set is selected by randomly choosing pairs of parents and according to the “crossover.rate” which gives the probability of choosing a feature from first parent instead of second parent.

* Sequential

This method consists of forward and backward search. The forward search includes each feature in feature step which model training and feature is selected according to its performance measure when its added to the feature set. The backward search is done by removing each feature and comparing its performance measure.

* Exhaustive

All feature set are searched and the best feature set which gives a good performance measure result is selected. This method is too much time consuming.

1. Model training

We mainly focussed on the random Forest classifier and rpart classifier. The feature selection is done along with the model selection and with the help of “iml” package we represent the feature importance in model.

1. Validation

For validation, we use the Leave one out cross validation because of less number of dataset we have at the moment. After validation, we also calculate the confusion matrix to see which set of features gives us a better accuracy.

**RESULTS**

We tried both Rpart and RandomForest classifiers after the benchmark predictions. But after validation, Random Forest classifier gave us good results compared to rpart, which made us to move forward with RF classifier. Clearly it shows that the rpart in first 3 methods, that performance measure is not as good as compared to Random Forest

**Rpart results**

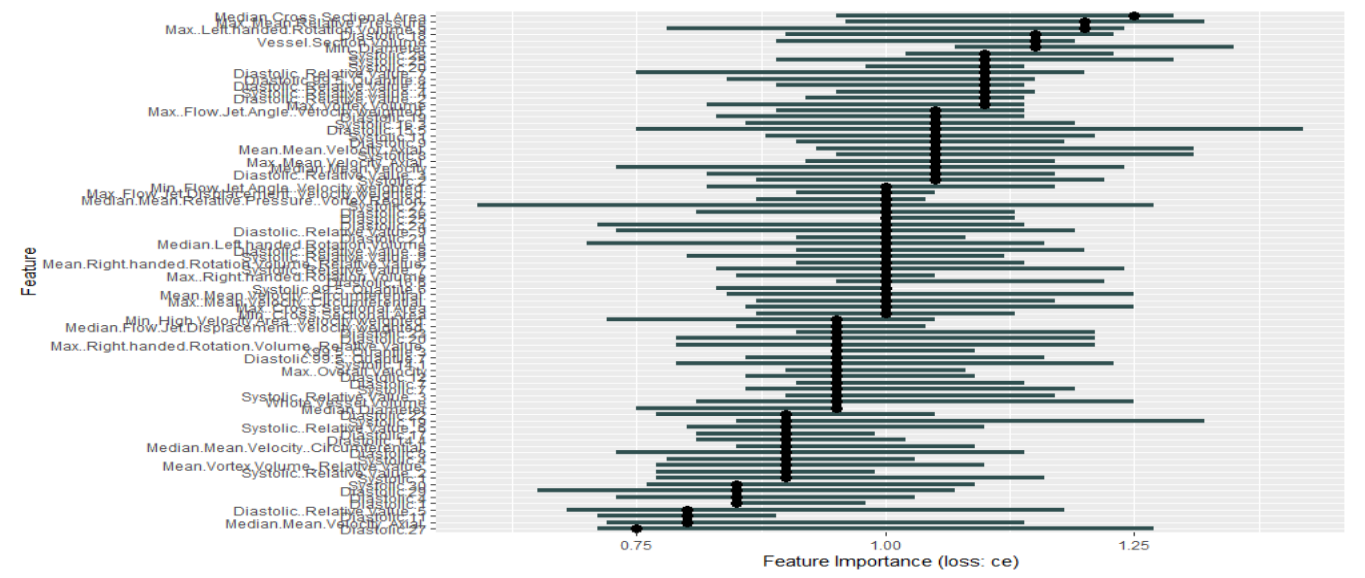
|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Whole Features | 0.7857 | 0.5635 | 154 |
| CFS | 0.8095 | 0.6129 | 8 |
| Information Gain | 0.8095 | 0.6129 | 8 |

Table 1

**Random Forest Classifier**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** | **Time Taken**  **(sec)** | **Standard deviation** |
| Whole Features | 0.8333 | 0.6682 | 154 | 3.11699 | 0.357 |
| CFS | 0.881 | 0.763 | 8 | 0.747998 | 0.357 |
| Information Gain | 0.881 | 0.7619 | 81 | 1.38434 |  |
| Chi Square | 0.8095 | 0.6182 | 81 | 1.45106 |  |
| Random | 0.8333 | 0.6697 | 74 | 1808.75 |  |
| GA | 0.881 | 0.763 | 82 | 6077.59 |  |

Table 2

Now, we fixed our classifier as Random forest. The Random and GA feature set is too large than we expected. So, we decided to plot the graph of features with their importance value. One example is shown as below: 

So according to this method, we selected the features with importance > 1 and tried validating the model. The results are:

**Importance > 1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Random FS | 0.8571 | 0.7136 | 38 |
| GA Feature Selection | 0.8095 | 0.6199 | 27 |

Table 3

After this, we thought we will try reducing the set further by **doing correlation in the selected feature** set irrespective of importance value.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** | **threshold** |
| Information Gain | 0.9524 | 0.9049 | 20 | 0.9 |
| Chi Square | 0.9524 | 0.9049 | 20 | 0.9 |
| Random FS | 0.8881 | 0.7619 | 24 | 0.99 |
| GA Feature Selection | 0.8333 | 0.6667 | 26 | 0.99 |

Table 4

After this, we considered the correlation of Random and GA Feature Selection with its **importance value > 1.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Random FS | 0.8333 | 0.6682 | 10 |
| GA Feature Selection | 0.7619 | 0.5249 | 15 |

Table 5

Now let’s consider **grouping the features** with same importance value during model training.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Information Gain | 0.8537 | 0.7078 | 11 |
| Chi Square | 0.8537 | 0.7078 | 10 |
| Random FS | 0.878 | 0.7568 | 10 |
| GA Feature Selection | 0.8537 | 0.7085 | 8 |

Table 6

On considering the dataset, we found that almost 30 features are the relative values of certain features. So, we thought of considering only the relative values and then repeat all the experiments we have done before.

Removing the original value, and considering the relative value only, our dataset ended up with 124 features.

So, with **Relative Values only**, the performance measures are

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **No.of Features** |
| Whole Features | 0.881 | 0.763 | 123 |
| CFS | 0.9048 | 0.81 | 10 |
| Information Gain | 0.878 | 0.7562 | 48 |
| Chi Square | 0.878 | 0.7562 | 48 |
| Random FS | 0.8571 | 0.7162 | 59 |
| GA Feature Selection | 0.7857 | 0.5753 | 63 |

Table 7

The Feature Set of CFS are:

1. "Max..Diameter"
2. "Mean.Diameter"
3. "Max..Cross.Sectional.Area"
4. "Whole.Vessel.Volume"
5. "Systolic Median.Vortex.Volume "
6. "Systolic Mean.Mean.Velocity Circumferential"
7. "Systolic Median.Right.handed.Rotation.Volume"
8. "Diastolic Median.Left.handed.Rotation.Volume"
9. "Max..Mean.Relative.Pressure..Vortex.Region"
10. "Median.High.Velocity.Area..Velocity.weighted."

Here as we done earlier,

we thought of considering the selected features of wrapper methods by **importance >1** after model training.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **No. of Features** |
| Information Gain | 0.878 | 0.7562 | 20 |
| Chi Square | 0.8537 | 0.7078 | 20 |
| Random FS | 0.8571 | 0.7162 | 22 |
| GA Feature Selection | 0.9048 | 0.81 | 18 |

Table 8

We **grouped the features** according to the importance value such that only one representative from each set of features for random and GA selection methods. The features are selected according to sql command.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Information gain | 0.8049 | 0.6077 | 8 |
| Chi square | 0.7317 | 0.4637 | 8 |
| Random FS | 0.8571 | 0.7149 | 10 |
| GA Feature Selection | 0.8333 | 0.6667 | 11 |

Table 9

Then, we again looked forward for the **correlation** between the selected features before consideration of importance value.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Information Gain | 0.8333 | 0.6636 | 15 |
| Chi Square | 0.8333 | 0.6636 | 15 |
| Random FS | 0.8333 | 0.6682 | 24 |
| GA Feature Selection | 0.8571 | 0.7149 | 24 |

Table 10

So, till now, we have followed the feature selection method first, then done the correlation in those selected features. This made us to think about considering doing the correlation in whole feature set, then do the feature selection. After this, compare the performance measures.

When **correlation** was performed in original dataset with 154 features, we ended with 44 features as final set with cut-off of 0.9.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Information Gain | 0.881 | 0.763 | 17 |
| Chi Square | 0.881 | 0.763 | 18 |
| Random FS | 0.8095 | 0.6199 | 23 |
| GA Feature Selection | 0.7615 | 0.527 | 24 |

Table 11

Then we considered **only relative value, and correlation** in the dataset of 124 features, which gave us 45 features as final set with cut-off of 0.9.

|  |  |  |  |
| --- | --- | --- | --- |
| **Methods** | **Accuracy** | **Kappa** | **Features count** |
| Information Gain | 0.878 | 0.7568 | 16 |
| Chi Square | 0.878 | 0.7568 | 16 |
| Random FS | 0.7857 | 0.5734 | 24 |
| GA Feature Selection | 0.8095 | 0.6216 | 16 |

Table 12

Task for next meeting

1. IG FS with correlation – 10min
2. Gender age and weight as feat – 20min
3. Standard deviation also – 30 min
4. Again documentation – 30 min