Assigned Project Report

Ankur Garg
agarg12@ncsu.edu

Sanket Shahane
svshahan@ncsu.edu

Abstract

The methods used for feature selection were Principal Component Analysis, Mixed Factor Analysis. Feature Subset Selection for selecting the best subset for MDP Process. Discretization was done using various binning techniques like Clustering, equal width binning etc. Greedy discretization for finding the optimal number of bins for discretization. Neural networks for feature compression.

1 Plan of Action

The data set provided contained a total of 130 features out of which 6 were fixed. So, out of the remaining 124, we identified the continuous and categorical variables. This was done by deciding that any variable with more than 10 unique values will be categorized as categorical variable. In case of any confusion, the feature description file was referred to understand the semantics of that feature. After this step, we identified 100 continuous and 24 categorical variables. We decided to handle these separately.

2 Processing Continuous Features

2.1 Correlation and PCA

First step towards processing continuous variables was to check the data for correlation and see how this affects MDP algorithm. We gave few highly correlated features as input to the MDP to see how it performs. It did not perform very well, as expected. So, we decided to remove the correlation from the data. Figure 1a and 1b, show the correlation matrix of the data before and after removing correlation. After removal of correlation from the 100 continuous features, the number of features came down to about 70 (continuous). Effects of removal of correlation: Number of principal components required to explain 97 percent of the variance increase from three to six.

2.2 Discretization

From the output of the Principal Component Analysis, we selected best 6 features and then performed discretization on them. Multiple ways for discretization of Principal Components were explored. They are detailed in the next subsections.

1. Based on data distribution: The distribution of the data was plotted to estimate the number of natural bins appropriate for each principal component. The plots for two such features are presented below. The discretized features were used for the MDP to calculate ECR.
2. Equal width Bins: Each feature was categorized into 8-10 equal width bins. For this purpose, pandas.cut functionality was used.
3. Equal Frequency Bins: This was a variated of the equal width bins which ensured that bins with very few samples were not created. For this purpose, pandas.qcut functionality was used.

4. Clustering: All the previous methods required that we decide the number of bins into which the data should be divided. But we decided to try MeanShift algorithm to calculate the natural number of clusters in the data. That turned out to quite large. So, we decided to vary the parameters to the algorithm to decide on an appropriate number of bins.

3 Feature Selection

3.1 Forward step selection

For feature selection, we started with forward stepwise subset selection for selecting best features for the MDP. The objective was to select the best set of features from the total feature set. But since checking each possible combination was not computationally feasible, we decided to use a greedy approach and use forward stepwise selection for finding the best subset of the features. Using this approach, we obtained an ECR value of 75.80. The Features for this were {Level, cumul_Interaction}.

3.2 Greedy Discretization

To further improve the ECR value, we decided to incorporate the results from PCA with the results obtained from Forward stepwise selection. To do this, we employed a scheme of greedy discretization. We realized that, the various discretization techniques we had used, did not yield very good results. So, the number of bins into which each feature discretized could be a major factor for that. So, we decided to vary the number of bins of each feature and find a number which provides the best ECR value when combined with the feature set obtained from forward subset selection. The algorithm for that is as follows:

Greedy Discretization Algorithm
pcaFeatures = {pca1, pca2, pca3, pca4, pca5, pca6}
featuresSubset = {ouput from forward subset selection}
for *feature* in pcaFeatures:
maxEcr = 0, optimalBins = 0
: **for** *numBins* in range(2,20):
: featureD = Discretized *feature* with bins = *numBins*
: newFeatures = featuresSubset + featureD
: call induceMDP() with newFeatures, output ecr
: if ecr > maxEcr:
: update optimalBins
: featureD = Discretized *feature* with bins = *optimalBins*
: Add featureD to the featuresSubset

This helped improve the ECR value from 75.80 to 81.26

4 Processing Categorical Variables - Mixed Factor Analysis

After processing the continuous features, we decided to take up the categorical features next. For this, we used Mixed Factor Analysis. Mixed Factor Analysis is basically a feature selection technique similar to PCA for handling data which contains categorical features. The output of the Mixed Factor Analysis was a set of continuous features out of which top 8 features were selected.
These features were discretized using the techniques mentioned in the previous sections. Discretized Mixed Factor Analysis features along with the PCA features were used to find the optimal set of features which would result in the best ECR. The ECR resulting from this process was around 25.2.

Further, we ran our previously mentioned, greedy discretization algorithm to find optimal number of bins for each feature of MFA. This resulted in further improvement of the ECR to 86.29 for a 6th feature of Mixed Factor Analysis for 2 bins. The figure below shows how varying the number of bins changed the ECR. Combining the results of PCA and Mixed Factor Analysis
We used greedy discretization technique on the features from both PCA and MFA. The resulting ECR was 82.5

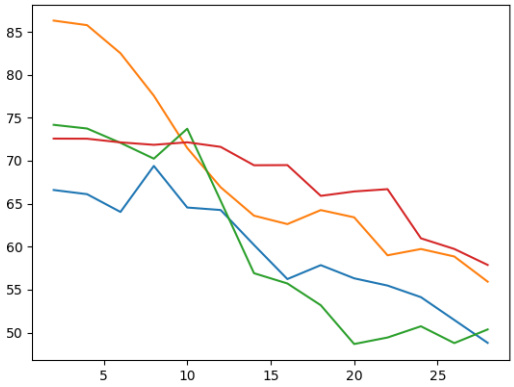

Figure 1: Results for Greedy Discretization

Table 1: Results

Method	ECR
PCA - Equal Width Bins	12.2
PCA - Clustering	10.4
Mixed Factor Analysis + PCA	25.2
Forward Subset Selection	75.80
Greedy Discretization + PCA + Forward Subset	81.26
Greedy Discretization + MFA + Forward Subset	86.29
Greedy Discretization + MFA + PCA + Forward Subset	82.5
Neural Networks	31.2

5 Neural Network Approach for Feature Compression

5.1 Method 1

After trying out the traditional approaches for feature selection, we decided to shift to a bit different approach. Using Neural networks for feature compression. We designed a neural network with two hidden layers, second layer having 8 neurons, and output layer same as input layer. This helped in compressing the 124 features to 8 features such that the same 8 features can map to the original 124. These 8 features were then discretized using the previously mentioned techniques and then given as input to MDP. Through this approach, we were able to achieve ECR value of 31.2.

5.2 Using MDP for training Neural Network.

In the next step, we decided to include the MDP process to train the neural network. This was done by using the 8-neuron layer output as input to MDP to calculate ECR and then defining the error function in terms of this ECR. This would help train the Neural network in such a way that would maximize the ECR value using the 8-nueron layer output. This could not be completed in time as the time required to train the neural network model was quite high.

6 Results

The results for the various techniques tried are presented below in the table. The best ECR achieved overall was 86.2 for a policy of 9000 rules. Out of which around 35 percent were defined and rest were “no rules”. Out of 3155 rules, 1073 were WE rules and 2082 PS rules. Results from all algorithms are in table 1.