Applied Data Mining - Machine Learning Methods

Kyriakos Chiotis (35278597)

***Abstract*—This document analyses and compares various data mining and machine learning techniques. These methods are applied on three different data-sets in order to handle different challenges of Data Science. Each data-set follows a fundamental analysis sequence which is data pre-processing, data partitioning(e.g. clustering) and classification. However, these steps differentiate according to the special properties of each data-set. The following sections regard data analysis on Pulsars, Mushrooms and Abalones. The objectives and tasks of this report are based on the final coursework of module SCC403 - Data Mining of Lancaster University.**

I. Introduction

The objective of data mining and machine learning is to use the modern high computational power to analyse and extrapolate big data. This is accomplished with low physical interpretability but could be justified through the various algorithms of machine learning. The primary role of data scientists is to know the theoretical background of these methods and apply these techniques efficiently. A fundamental procedure of data analysis concerns pre-processing, clustering and classification.

Data pre-processing is useful to convert raw data in an efficient format with techniques such as data cleaning, transformation and reduction. It is the first step of data analysis and it is very important for validated results in clustering and classification. Data clustering can be considered as linked to classification but also can be seen as a separate tool for data partitioning and analysis. Sometimes, it is considered as a part of pre-processing because it decomposes complex, non-linear problems into simpler, locally valid sub-problems(Divide et impera). Nevertheless, the main concept is to expose the natural pattern of the data which has areas of high density and border areas of lower density. Finally, classification is the process of assigning labels to data points. It is considered an instance of supervised learning because it tries off-line to fit a training set of data to predict the class of a feature. This is accomplished through various algorithms, called classifiers. An effective classifier can be accomplished through the correct approach of pre-processing phase and the full comprehension of the theoretical and practical background of classifying algorithms.

II. Pre-processing

Data pre-processing is organised in a gradual process which is described as follows:

1) Scale or encode features according to data type.
2) Feature selection/extraction.
3) Handle missing and imbalanced data.

The pre-processing follows these steps and each of them includes a set of actions to transform the problem of a data-set to a convenient form.

A. Pulsars data-set

Pulsars data contain 17,898 samples with 8 features and no missing data. All features are comprised of continuous values except the target class which is in a binary form. Also, the data are imbalanced while they consist of 1,639 positive examples and 6,259 negative examples. However, the imbalance techniques are demonstrated only on the last data-set.

Because of numerical continuous type of data, it is a good practice to scale the data in order to make them comparable by values and ranges and be applicable for many machine learning algorithms. The two most common techniques are normalization(or Min-Max scaling) and standardization(or Z-score normalization). For the later purposes of the analysis which are going to be explained, standardization is chosen. In standardization the features are scaled in order to follow a normal distribution with

$$\mu = 0 \quad and \quad \sigma = 1$$

and the new values, called standard scores(or z scores) are calculated as follows:

$$z = \frac{x - \mu}{\sigma}$$

The preferable choice is standardization because this scale helps to measure the similarity(useful in LDA) and the variance(useful in PCA) between the features. Similarity or proximity of features is calculated with Euclidean distance. Therefore, the scaled data are ready for the following task of Principal Component Analysis(PCA) to reduce the dimensionality of the data-set.[1]

PCA centralize the data on the origin and finds the best fitting line by maximizing the sum of the squared distances from the projected data points to the origin. This line is called Principal Component 1(or PC1). PC2 is perpendicular to PC1, PC3 perpendicular to PC1 and PC2, etc. The feature extraction is done according to the variation of PCs which is calculated as follows:

$$Variation = \frac{SS(distances)}{n - 1}$$

where n is the sample size. In our data-set, five principal components cover over 95% of the variation as shown in Figure 1 and therefore the data dimensions are reduced to five.

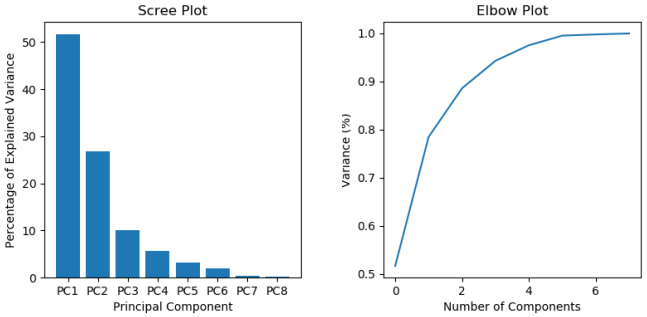

Fig. 1. Explained variance with Scree and Elbow plot

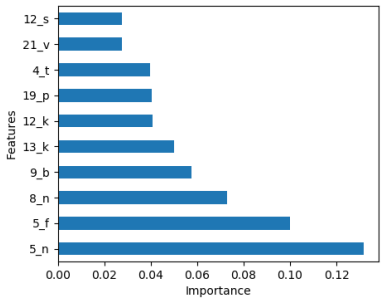

Fig. 2. Ten most important features according to Random Forest

Finally, the data pre-processing is completed and the data-set is ready for clustering and classification techniques.

B. Mushrooms data-set

Mushrooms data contain 8,124 samples with 22 features. Also, the data are balanced while they consist of 4,208 edible mushrooms and 3,916 poisonous mushrooms. However, this data-set contains 2,480 missing values in the form of '?' at stalk-root feature. Some options to handle missing data:

1) Remove rows with null values.
2) Remove features with null values.
3) Predict(impute) the missing data.

Removing the rows would be quite ineffective solution while the rows with missing data cover the 30% of the data. Removing the entire feature is not also preferable while information is lost. Two ways of imputing categorical data can be achieved by filling in the missing values with the most common value or according to a feature with high correlation. So explanatory data analysis with bar plots, Cramer's V and Theil's U are some ways to check correlation between categorical nominal features. The further analysis is based on the option of filling the missing data according to label frequency.

Mushrooms data-set consists of categorical data which need to be converted to numerical for further analysis. Label encoding is used for binary features and one-hot encoding for features of more than 2 categories. These features now are called dummy variables. One-hot encoding increases the dimensions of our data-set to 113 which implies a high risk of massively overfitting our model. This time techniques like PCA or LDA have no effect as the data is categorical and feature selection could be tried.

Feature selection could be achieved by calculating the feature importance. Ensemble methods which are used by decision trees, rank the features and select the most important[2]. Reducing the dimensions of data lead to several benefits such as:

1) Reduces the risk of overfitting.
2) Speeds up the classification algorithms.
3) Increases model intepretability.

Even if the running phase is the pre-processing, data is splitted to train and test set and Random Forest classifier is used to calculate the importance of the features as shown in Figure 2.

Figure 2 indicates that full and none odor and gill size of mushrooms are the three most important features. Consequently, the rest features could be omitted and our data-set is ready for classification.

C. Abalone data-set

Abalone data contain 4,174 samples with 9 features and no missing data. However, the data are imbalanced while they consist of 32 positive examples and 4,142 negative examples. Initially, it is helpful to scale the numerical features using standardization and convert the 'sex' feature to a dummy variable.

This data-set is highly imbalanced, with only 0.7% of class target being classified as positive. This imbalance make the classifying algorithms ineffective as most of them are trying to maximize the accuracy. Even without training a model, the prediction accuracy on a test set of this data-set would be approximately 99% using a classification based on the frequency of the labels. Therefore, the application of other metrics may give better insight[3]. These metrics are listed below:

1) Confusion Matrix: An overview of the correct and incorrect predictions.
2) Precision: The trustworthiness of the predictions.
3) Recall: The ability to predict the targeted class.
4) F1-score: The harmonic mean of precision and recall.

After data split to train and test set, a logistic regression classifier can be used to check the above metrics.

		Predicted	
		P	**N**
Actual	**P**	TP: 0	FP: 7
	N	FN: 0	TN: 828

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = 0.99 \quad (1)$$

$$Precision = \frac{TP}{TP + FP} = 0 \tag{2}$$

$$Recall = \frac{TP}{TP + FN} = 0 \tag{3}$$

It is obvious that the positive class is poorly handled from the model and resampling techniques would be appropriate. These methods act on the data-set as follows:

- Undersampling the majority class.
- Oversampling the minority class.
- Generating synthetic data.

Oversampling is preferred to undersampling while the data-set has not enough data to work with. However, the new metrics of oversampling do not imply an effective model, too. Accuracy decreases to 77%, recall increases to 0.57 and precision increases only to 0.02. Consequently, the poor precision of the model indicates the usage of synthetic samples.

One way to generate synthetic data is the Synthetic Minority Oversampling Technique or SMOTE. SMOTE is based on nearest neighbour algorithm. In particular, SMOTE process identifies the feature vector and its nearest neighbour. Afterwards, it calculates their difference using a distance metric and multiplies the difference with a random number between 0 and 1. The result is a new point or a new sample. Consequently, the new metrics slightly improve. To conclude, the resampling methods and logistic regression did not fit an effective model. However, testing more classification algorithms may result to better metrics. This part is illustrated on classification section.

III. Clustering

Clustering and Segmentation is used to group data space. Each group(or cluster) consists of data samples with high similarity indicating patterns which may give a better insight to the problem. The clustering process could be described as follows:

1) Confirm that data are numerical and comparable.
2) Pre-process data with scaling, feature selection and feature extraction techniques.
3) Define the similarity measure.
4) Choose clustering method and number of clusters.
5) Interpret and evaluate the resulted partitioning.

In this document, similarity is measured according to the distance metrics as mentioned before. Two of the most popular algorithms of data partitioning are Agglomerative Hierarchical Clustering and k-means Clustering. Both of them are going to be applied and analysed thoroughly.

A. Pulsars data-set

The aforementioned process is going to be used as a guide on clustering the data-set. Initially, it is confirmed that data are numerical and therefore can be compared with each other. Also, it should be stressed out as a reminder the current status of the data after pre-processing. The data were scaled with standardization and the dimensions were decreased to 5 after PCA.

The Agglomerative type of Hierarchical Clustering is straight forward. The algorithm compute the proximity matrix and set each data point as a cluster. Thus, it compares the distances of the clusters and merge the two closest. This process is iterative until a single cluster remains. It should be mentioned that euclidean is used as a distance metric and the linkage function which links the pairs of data points is based on Ward's Criterion[4].

Ward linkage method minimizes the variance of the clusters being merged while the distance is the sum of squared deviations from data points to centroids[5]. Consequently, it operates well in case of noisy data and according to outliers detection of pre-processing phase is the preferable option. The application of Hierarchical clustering technique could be visualized using a Dendrogram like in Figure 3.

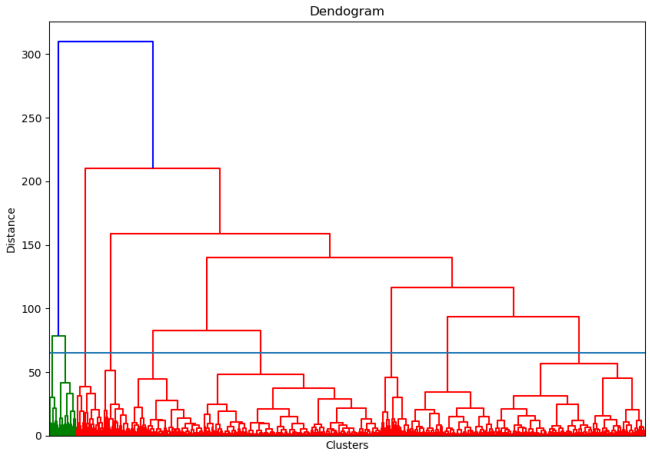

Fig. 3. Dendrogram of Agglomerative Hierarchical Clustering

Choosing the number of clusters would be easier for astrophysics experts who may interpret the clustering results. However, using the heights of branches in Figure 3, the data space could be partitioned in 9 clusters. This is visualised by the horizontal line of dendrogram. Also, the clusters could be visualised using a scatter plot and the two features with the most variance according to PCA.

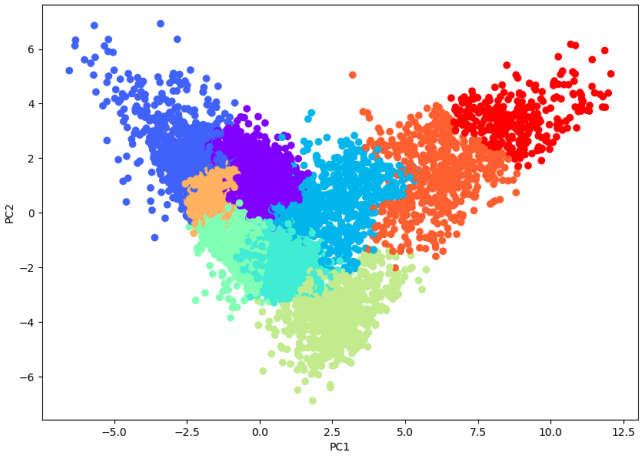

Fig. 4. Scatter plot of Hierarchical partitioned data

Likewise, the interpretation and evaluation of clustering in Figure 4 could be analysed by astrophysics experts. Hierarchical clustering requires a lot of memory (exponentially growing) with space complexity $O(n^2)$. Also, the method requires many iterations to reach a result with time complexity $O(n^3)$

The k-mean algorithm is using a different approach. It begins with a random number of clusters(centroids) and assign each data point to the closest cluster according to a distance metric. It calculates the average of the assigned points and set new cluster centroids. This process is iterative until the cluster assignments remain stable[6]. This time the number of clusters could be determined by elbow method. Starting with 10 initial clusters(guided by Hierarchical Clustering), the elbow is plotted in Figure 5.

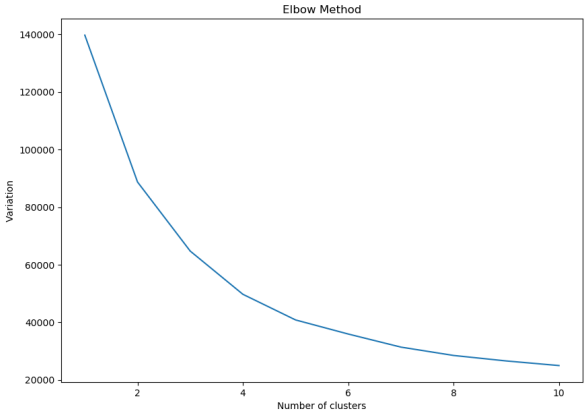

Fig. 5. Elbow plot of k-mean clustering

In Figure 5, it is obvious a smoother reduction in variation after ninth cluster. Therefore, 9 clusters are chosen which could be visualised and compared to Hierarchical clustering as it uses the same number of clusters.

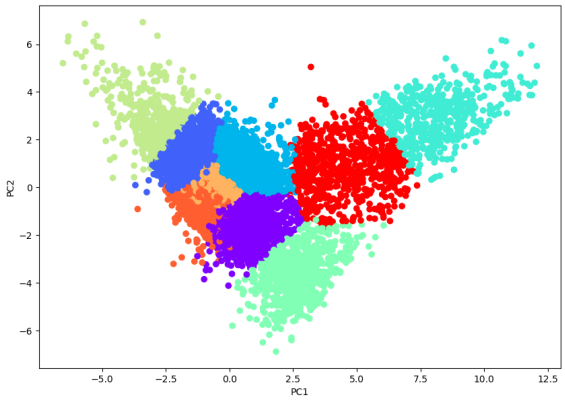

Fig. 6. Scatter plot of k-mean partitioned data

Similarly, the partitioned space of the data-set in Figure 6 may be interpreted and evaluated by astrophysics experts. In contrast with Hierarchical clustering, k-means is quite faster and also proposes visually a configurable number of clusters. However, it is based on the arbitrary set of initial clusters number and may need some test cases. Taking into consideration the two clustering algorithms, it is concluded a similarity in the way the data are partitioned.

B. Mushrooms and Abalone data-sets

As mentioned above, clustering techniques need comparable data. Comparable data consist of continuous numerical and ordinal categorical features. However, most of mushrooms' features are nominal categorical which implies ineffective clustering algorithms. Abalone data-set consist of continuous variables except one nominal 'sex'. Using a subset of the data-set for clustering is not preferred because it violates the correlation between the features and information is omitted.

IV. Classification

Classifiers can be trained by supervised learning and training data (with labels) in order to predict a class label. This document applies and analyses some of these algorithms such as Logistic Regression and Decision Trees(Random Forest). Classification section could be divided into two phases. The first phase is the supervised training of the classifier. The second phase is the application of the classifier on a test set and the evaluation of its results. Some evaluation techniques that are going to be used are Cross Validation, Receiver Operating Characteristic(ROC) curve and Area Under the ROC curve(AUC). These methods consist of the basic classifying metrics accuracy, precision and recall.

Classification is the final objective on Mushroom and Abalone data-sets where the class label is going to be predicted. It is very important to choose wisely the classifiers according to the data type. As both data-sets contain categorical

nominal features, logistic regression and decision tree(Random Forest) algorithms are the preferable options. The classification process which is used could be described as follows:

1) Data split in training and test set.
2) Fit the training set with a classifier.
3) Predict the class on the test set.
4) Cross Validation.
5) Interpretation and evaluation of the results.

This process is going to be used as a guide on the aforementioned classification techniques which are introduced and described below.

Logistic regression outputs a probability for each instance which is placed on a sigmoid function(logistic regression function). Therefore, it is essential to set a threshold according to which each instance is classified 0 or 1. The threshold is set according to the importance of the predicted result. For example, it is obvious that predicting a poisonous mushroom is much more important than predicting an edible one. For start the threshold is set 0.5.

Decision trees operate using a tree structure with each internal node corresponds to an attribute and each leaf node corresponds to a class label. In contrast to logistic regression, decision tree algorithms do not require linear relationship between the features and handle better categorical data(ex. mushrooms data-set). On the other hand, they have high risk of overfitting and may lose information when they deal with continuous numeric data. Random Forest is based on the same tree structure logic but it combines several decision trees. Therefore, the predictive power is increased and the computational power is decreased.

Cross validation splits the data in a configurable K number of blocks and repeats steps 2 and 3 of the classification process above. Therefore, it increases the algorithm performance with more reliable metrics. Ten-fold cross validation is a common practice and is applied on the data-sets[8].

Performance metrics were analysed on pre-process section of imbalance data. In addition to this analysis, ROC and AUC is a powerful performance tool. In short, it is a metric of distinguishing between classes and is based on Sensitivity(or Recall or TPR) and Specificity(or FPR):

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

A. Mushrooms data-set

Initially, it should be mentioned the current status of the data after pre-processing. The data are comprised of categorical features which were converted to dummy variables. Missing data were imputed with the most common value and feature selection was reduced the data-set dimensions to 3. Consequently, the data is splitted in training and test set because of feature selection with random forest. Finally, the targeted class was identified as poisonous and edible mushrooms and separated from the rest data. The application of the classifiers on the data produce the following results.

	Logistic Regression	Decision Tree	Random Forest
Accuracy	0.97	0.97	0.97
Precision	0.97	0.97	0.97
Recall	0.96	0.96	0.96
F1-Score	0.97	0.97	0.97
AUC	0.97	0.97	0.97
Time	0.1875	0.0468	1.6406

The above metrics indicate that all classifiers handle efficiently the targeted class and the metrics are the same except CPU processing time. This similarity is based on the feature reduction and the simplification of the model.

B. Abalone data-set

Initially, the pre-processing changes should be pointed out as a refresher. The continuous features were scaled with standardization technique and 'sex' feature was transformed to dummy variable. Also, a logistic regression was applied to choose an imbalance technique and therefore synthetic data were produced with SMOTE method.

In pre-processing phase, it was observed that logistic regression is not efficient with a simple data split. Consequently, cross validation and decision tree are applied to check for improvements.

		Logistic Regression	Decision Tree	Random Forest
Original	Accuracy	0.99	0.98	0.99
	Precision	0.0	0.0	0.0
	Recall	0.0	0.0	0.0
	F1-Score	nan	nan	nan
	AUC	0.50	0.49	0.50
	Time	0.2343	0.0781	2.4687
SMOTE	Accuracy	0.80	0.96	0.98
	Precision	0.025	0.02	0.05
	Recall	0.60	0.09	0.08
	F1-Score	0.05	0.03	0.06
	AUC	0.70	0.70	0.70
	Time	0.4375	0.5468	9.6875

Taking into consideration the above metrics, Random Forest classifier is slightly better than the others. However, either way the precision or the trustworthiness of predicting the positive label is very low.

V. Conclusion

Taking everything into consideration, this document analyses and applies a large number of Data Mining and Machine Learning techniques. These methods could be summarised as follows:

1) Pre-processing
    a) Scaling with standardization and normalization
    b) Feature Selection with PCA
    c) Feature Extraction with Random Forest
    d) Missing value imputing
    e) Handling imbalanced data with Oversampling and SMOTE
2) Clustering
    a) Agglomerative Hierarchical Clustering
    b) k-mean Clustering
3) Classification
    a) Classification with Logistic Regression, Decision Tree and Random Forest
    b) Cross Validation
    c) Classifier evaluation metrics

References

[1] https://statquest.org/video-index/ - Principal Component Analysis (PCA) Step-by-Step PCA in Python
[2] https://towardsdatascience.com/feature-selection-using-random-forest-26d7b747597f
[3] https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28
[4] Charu C. Aggarwal, Chandan K. Reddy - Data Clustering: Algorithms and Applications 1st Edition (4.3.1.3 Ward's Criterion)
[5] Trevor Hastie, Robert Tibshirani, Jerome Friedman - The Elements of Statistical Learning 2001 - 14.3 Cluster Analysis Agglomerative Clustering
[6] Trevor Hastie, Robert Tibshirani, Jerome Friedman - The Elements of Statistical Learning 2001 - 13.2.1 K-means Clustering
[7] Trevor Hastie, Robert Tibshirani, Jerome Friedman - The Elements of Statistical Learning 2001 - 4.4 Logistic Regression
[8] https://towardsdatascience.com/train-test-split-and-cross-validation-in-python-80b61beca4b6

Appendix

Libraries in code:

- numpy
- pandas
- scipy
- sklearn
- matplotlib
- numpy