Predicting the Star types with using Supervised Learning Approaches and their Comparison to Capability

Kaan KALAN, 201611032, c1611032@student.cankaya.edu.tr

Çankaya University, Computer Engineering Department, Turkey

Abstract—Many factors affect the success of Machine Learning on a given task. In this paper, we are exploring the effect of preprocessing statements at different classification methods for predict Star Type. We are considering 3 main classification method for measure the prediction capability. As a result, we recommend the most suitable preprocessing steps and classification method in comparison with the data set we have. In first step of our method, classification capability results saved without using any preprocessing stage. In the second step, data set visualized and attribute's effect to predicting star type determined. In the third step, every preprocess stages contribution to classification is recorded when working with evenly distributed but attribute values irregular dataset. The experimental results show that using preprocess stage has huge effect to improve classification methods capability when using for specified classification method. At the end of paper, recommended a preprocess combination for this kind of datasets and compared the effects of this stages on the Classification methods.

Keywords: Preprocess, Star type, Prediction, Classification

1 Introduction

In today's world, astronomy and cosmology are concerned with analyze the characterization of millions of objects that can be identified with their optical spectra [1]. There are a lot of different tools that can assist or fully automate the classification of stars and galaxies from astronomical images. Source Extractor (SExtractor) uses an Artificial Neural Network (ANN) to perform the star classification.[2] Also Advanced image processing techniques and powerful learning algorithms make automated star and galaxy classification a faster alternative against to their manual counterpart [3].
In this paper, recommended a preprocess combination for better predict to star type with kNN, SVM and Naive Bayes classifier algorithms. The data preprocessing generally have important impact on generalization performance of a supervised ML algorithm. Detecting outliers, feature normalization, feature selection are most known preprocess statements [4].
This paper addresses issues of data pre-processing that can have a significant impact on generalization performance of a ML classifications algorithms. We used aforementioned pre-process stages one by one and tested their effect at the classification capability. Then, we merged them with meaningful combination and compared on the classification algorithms again.

The next section covers the several published articles related to the subject of this study. Section 3 includes more detailed information about used data set, used pre-processing statements, and compared classification methods. Section 4 includes the main comparison between pre-processed and un-preprocessed data classification capabilities with different ML algorithms. At the last section, gave a quick summary about the project and some deductions were made and gave ideas for future works.

2 Related Work

Peng et al. [5] tried to present new approach for improve the generalization capability. In their project, k-Nearest Neighbors(kNN) integrated into Support Vector Machine (SVM) and created new method. This method used for fixing the some predict error and improve the predict capability. With this approach, got up to 97% prediction success.

Also, at different researches, there are some applications that determine whether the object is star or galaxy. O'Keefe et al[6] worked on Source Extractor (SExtractor). This application uses advance image processing techniques but when conditions are not ideal, the results produced are often very poor and, in some cases, completely unreliable. Depending on the quality of the image, Application may not be able to clearly distinguish between stars and galaxies. In this research, the results have been tested with Random Forest (RF), Decision Tree(DT) and ANN data mining techniques and also this researched data set is balanced like ours. They used these methods on the images. Also detailed literature information can be found in surveys by Bertin et al. [7].

Kotsiantis et al.[4] used some pre-processing methods for different kind of data types like if there are some missing values, they present one preprocess stage. Or for other different situation exist, offers different techniques. Kotsiantis et al[4] provides general tips and explain how works different preprocessing methods for this situation.

In another research, described the use of a new artificial neural network, called the difference boosting neural network (DBNN), for automated classification problems in astronomical (star galaxy) data analysis. Results compared with common Source Extractor (SExtractor) package. With this research Philip et al.[7] presents DBNN's performance is about similar with SExtractor but DBNN is faster at train and classification.

At another research, some specific preprocess methods effect results at different data sets. Tried to find one best standard wrapper approach. But results proved that there is no one best specific model for different datasets. But experiments show that using feature selection absolutely increase the classification accuracy.

3 Methodology

In this study, we used Orange for apply the all used method and simulate the results. First, we pick the Star Type attribute as a target value. After specifying the target attribute, started to data analysis stage. In this stage, we check the correlation between attributes and visualize the data distribution. According to these information, we start to use different significant preprocess steps with different combinations and detect their effect on various classification algorithms. The general schema of the proposed work is shown in Fig. 1. In the following sections, used Dataset, preprocessing steps and classification results are explained

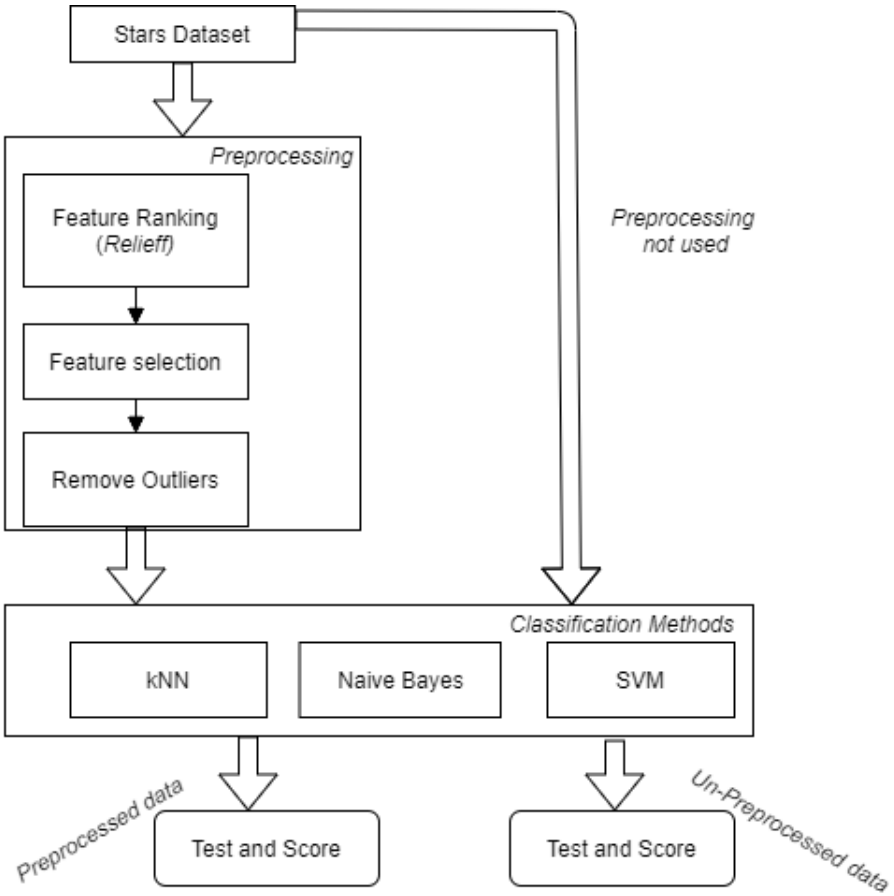

Fig. 1. General schema of the proposed method

3.1 Dataset

Used dataset was made for make a star classifier using Deep Learned Models like DNN. Most of the data was found from web and missing data were manually calculated by data collector with using several equations in astrophysics like: Stefan-Boltzmann's law of Black body radiation (To find the luminosity of a star), Wienn's Displacement law (for finding surface temperature of a star using wavelength), Relationship with absolute magnitude and radius of a star using parallax.
Dataset has 240 instance and 7 features. At the beginning, our target attributes data type was Numeric. But we translate it to categorical value. Because Star type values are numerically insignificant. Converted Dataset's feature data types and their short definitions with converted form shown At the Table1.

Table 1. Feature properties

Attribute Name	Attribute data type	Short Definition
Temperature(K)	Numeric	Surface temperatures of several stars
Luminosity(L/Lo)	Numeric	Luminosity of several stars calculated with respect to sun(L/Lo)
Radius(R/Ro)	Numeric	Radius of several stars calculated with respect to sun(R/Ro)
Absolute Magnitude (Mv)	Numeric	Absolute Visual magnitude (Mv) of several stars
Star type	Categorical	This column is the target class (6 classes ranging from 0-5) 0 -> Brown Dwarf 1-> Red Dwarf 2 -> White Dwarf 3-> Main Sequence 4 -> Supergiant 5 -> Hypergiant
Star color	Categorical	Colors of each star after Spectral Analysis
Spectra Class	Categorical	Spectral classes of each star(O, B, A, F, G, K, M)

Also, dataset's distribution is shown in Fig.2. In our dataset, feature value ranges are so unproportioned. For example, Absolute magnitude values change between -11.9 and 20.1 but Luminosity change 0.09 and 849k. Even this observation gives us some ideas about methods to can be applied.

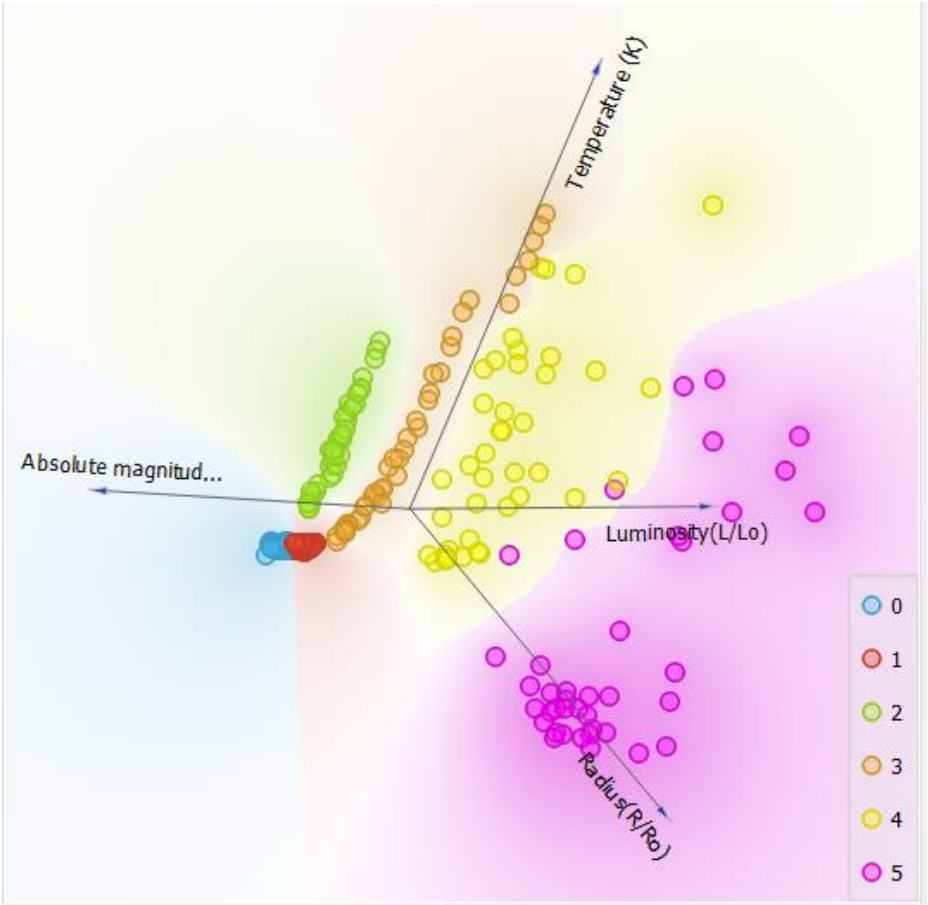

Fig. 2. Linear Projection of the Dataset

3.2 Preprocessing

While doing this project, the following Z Score Normalization (Standardization), Outliers and ReliefF Feature Selection were used. Normalization is the process of rescaling data without changing its behavior or qualification. The main purpose of this process is to create a common scale without disturbing the values of the data in the dataset and the differences in the value range. Its formula is shown in Eq. (1)

$$x' = \frac{x - average(x)}{std.dev(x)} \quad (1)$$

Outliers are excessive values that deviate from other data of the dataset. It is observation deviating from the general pattern on an example. These values are removed from the dataset as they will negatively affect the estimates to be made. The feature selection process reduces the size of the data by removing the unrelated and unnecessary features of the subset to be used in model making. ReliefF calculates a feature score for each feature which can then be applied to rank and select top scoring features for feature selection.

3.3 Classification Methods

In this study, we conducted different experiments with different parameter settings in the same dataset to observe the performance of the classifiers in the processed dataset and the untreated dataset. As we mentioned earlier in this study, we use three different classifiers to carry out our experiments.

kNN

One of the features of the classification is to look at the closeness of the new individual to the previous individuals to k. There is no training phase. It does not learn the training data, instead it memorizes the training data set. When we want to make an estimate, it looks for the nearest neighbors in the whole dataset. The k value is determined. K value is the number of elements to be checked. When a value comes, the distance between the incoming value is calculated by taking the nearest k element. Euclidean function is generally used in the distance calculation process, we used Euclidean in our application, besides, it can be used in Manhattan, Minkowski and Hamming functions.

Naive Bayes

Naive Bayes classifier is a probabilistic classifier based on Bayes' theorem with independent assumptions. It calculates the probability of each situation for a data and classifies it according to the highest probability value. If a value in the test set has an invisible value in the training set, it cannot estimate and returns a probability value of 0.

SVM

The algorithm tries to find the best line that divides it into two groups. It allows the line to be drawn to be adjusted from the farthest place to the elements of the two groups. In this way, we can be able to classify the data we have and the future data. The points cut by the parallel range drawn are the support points. Class labels with the form of support vector machines are generally used as (-1, + 1).

4 Results

The first scenario, the following values were obtained without applying any preprocessing method to dataset. And the target variable star type is selected.

Table 2. Classification Methods Estimation of First Scenario

Settings

Sampling type: Leave one out
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA
kNN	5.371	2.735	0.6166666666666667
SVM	12.095	4.439	0.7333333333333333
Naive Bayes	1.276	0.325	0.9125

The distribution of the data is shown in Fig. 3.

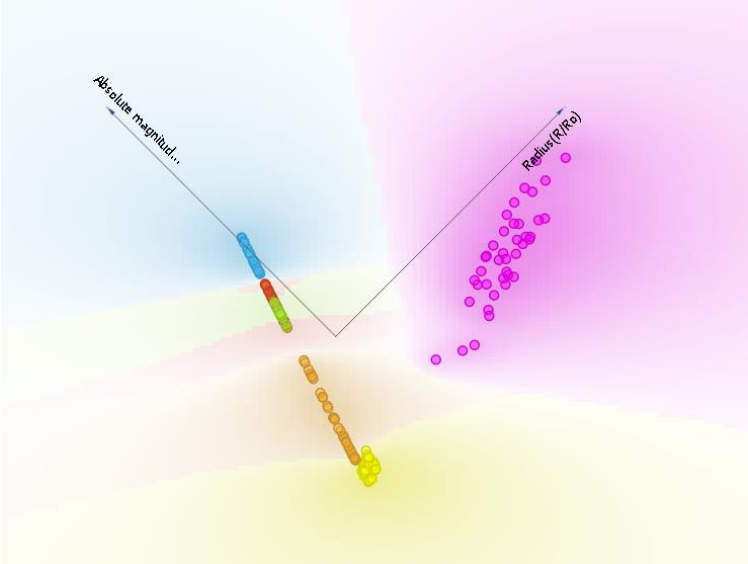

Fig. 3. Distribution of the first scenario with absolute magnitude and radius

When the results were examined, it was seen that the best result was obtained with the Naive Bayes method. When dataset was examined, it was thought that some data might affect the result.

Second scenario, the z score normalization method is applied to the dataset. Unnecessary and worthless data is determined with this method. These determined values are removed from the dataset using outliers. In order to subtract these values, when we make the kernel value 0.05 and the regularization value 5, 13 data are subtracted from the dataset. And the target variable star type is selected.

Table 3. ReliefF Feature Selection Result table linked to Star Type

Input

Features: Star color, Absolute magnitude(Mv), Radius(R/Ro), Luminosity(L/Lo), Temperature (K), Spectral Class
Meta attributes: Selected, Selected (1), Selected (2)
Target: Star type

Ranks

	#	ReliefF
Spectral Class	7.0	0.48144915758834117
Star color	17.0	0.44788532838707
Absolute magnitude(Mv)		0.3696525228760127
Radius(R/Ro)		0.19934004970954972
Luminosity(L/Lo)		0.1066963164747025
Temperature (K)		0.09682837481980332

Output

Features: Spectral Class, Star color, Absolute magnitude(Mv), Radius(R/Ro)
Meta attributes: Selected, Selected (1), Selected (2)
Target: Star type

Feature selection method is applied to increase the success of prediction methods. The number of features was reduced to 4. The distribution of the data is shown in Fig. 4.

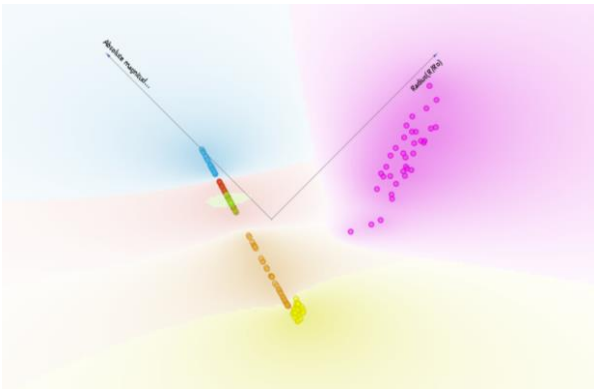

Fig. 4. Distribution of the second scenario with absolute magnitude and radius

Table 4. Classification Methods Estimation of Second Scenario

Settings

Sampling type: Leave one out
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA
kNN	6.320	3.590	0.9955947136563876
SVM	16.034	4.308	0.7929515418502202
Naive Bayes	2.100	0.343	0.8986784140969163

Considering the results, there is a serious increase compared to the values of first scenario.

Third scenario, Normalization operations are performed using the values below.

Table 5. outliers result table

Data

Input instances: 240
Inliers: 190
Outliers: 50

Detection

Detection method: One class SVM with non-linear kernel (RBF)
Regularization (nu): 22
Kernel coefficient: 0.25000000000000006

According to the data given by normalization process, 50 data are extracted from the dataset. With the ReliefF Feature Selection, the data set has been reduced by 4 features. The following values were obtained with the 190 data. The distribution of the data is shown in Fig. 5.

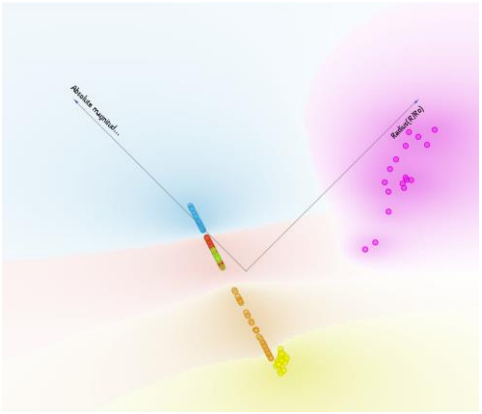

Fig. 5. Distribution of the third scenario with absolute magnitude and radius

Table 6. Classification Methods Estimation of Third Scenario

Settings

Sampling type: Leave one out
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA
kNN	3.115	2.189	0.9823788546255506
SVM	9.353	2.449	0.7048458149779736
Naive Bayes	2.073	0.109	0.8590308370044053

According to the results, it is better than the results of section first scenario. However, considering the results of section second scenario, there is a decrease in values.

Fourth scenario, Normalization and outliers' processes used in section second scenario are done. Then, feature numbers were reduced by using feature selection. The following values were obtained. And the target variable star type is selected. The distribution of the data is shown in Fig.6.

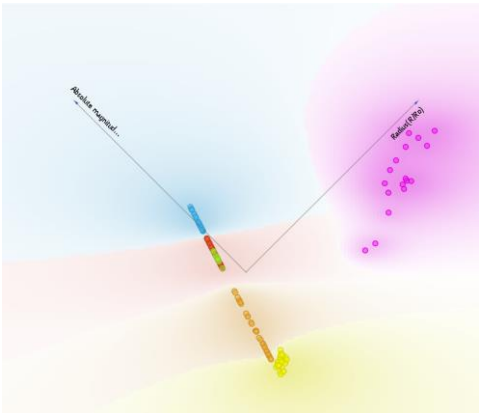

Fig. 6. Classification Methods Estimation of Fourth Scenario

Table 7. Classification Methods Estimation of Fourth Scenario

Settings

Sampling type: Leave one out
Target class: Average over classes

Scores

Model	Train time [s]	Test time [s]	CA
kNN	3.314	1.931	0.9736842105263158
SVM	8.152	2.190	0.8
Naive Bayes	0.941	0.188	0.9

According to the third scenario, it seen that the estimated values have increased. The feature selection positively affects the accuracy of the estimates. However, according to the scenario, the accuracy rate of the estimates is lower.

Summarize, In the first scenario, Naive Bayes estimates are the method with the highest accuracy value in the results obtained without preprocessing operations. As a result of the operations we performed in the second scenario, the accuracy rate of the Naive Bayes method decreased. The accuracy rate of the KNN method has increased. In third Scenario methods were applied by keeping the outliers range high. However, it was noticed that the data required by this method was deleted. In the fourth scenario, the effect of the reduction by using the third scenario's operation was examined. *As a result*, the best guessing scenario is the second scenario. The best estimation method.

5 Conclusion

This paper describes the effects of the pre-processing steps, the comparison of classification algorithms among each other in different combinations we make while finding the most appropriate predictive classification algorithm in our study. We compared the results of success rates according to classification accuracy, train, and test time evaluation results of kNN, Naive Bayes, SVM algorithms. Naive Bayes gave more successful results in the classification results without applying preprocessing steps to our data. When we reduced the number of features with rank from six to four, one of the pretreatment steps, this only affected kNN and its success increased. As a second step, when we applied the Rank step after the normalized feature to our data, we observed that the train and test times decreased and the successful prediction of the kNN algorithm increased. In our last experiment, we apply normalized features and outliers to our data, then we take the four best features we decided with the ReliefF values in the rank step and finish the pre-processing phase. When we compare the classification results of this experiment with the non-pretreated data, we see that SVM and Naive Bayes drop very low and kNN increases very high. Naive Bayes were faster at the test and train times than any other classification at each stage. The kNN algorithm provided high performance in data with preprocessing data while Naive Bayes algorithm always maintained its successful prediction rate.
They can be added to improve this work in the future: to analyze the estimation success of the classification algorithms by adding the images of the stars added to our data set as a new feature to our data. Secondly, to remove the target types which are the target variable from the data set, to create a class with unsupervised methods and to compare the prediction success in classification methods.

References

1. Krakowski T., Małek K., Bilicki M. et al 2016 A&A 596 A39
2. O'Keefe, Peter J., et al. "Star-Galaxy Classification Using Data Mining Techniques with Considerations for Unbalanced Datasets." Astronomical Data Analysis Software and Systems XVIII. Vol. 411. 2009.
3. Philip, Ninan Sajeeth, et al. "A difference boosting neural network for automated star-galaxy classification." Astronomy & Astrophysics 385.3 (2002): 1119-1126.
4. Kotsiantis, S. B., Dimitris Kanellopoulos, and P. E. Pintelas. "Data preprocessing for supervised learning." International Journal of Computer Science 1.2 (2006): 111-117.
5. Peng, NanBo, YanXia Zhang, and YongHeng Zhao. "A SVM-kNN method for quasar-star classification." Science China Physics, Mechanics and Astronomy 56.6 (2013): 1227-1234.
6. Bertin, E. & Arnouts S. 1996, A&AS, 117, 393
7. Philip, Ninan Sajeeth, et al. "A difference boosting neural network for automated star-galaxy classification." Astronomy & Astrophysics 385.3 (2002): 1119-1126.
8. Ahmed, Nesreen K., et al. "An empirical comparison of machine learning models for time series forecasting." Econometric Reviews 29.5-6 (2010): 594-621.