Classification performed on an Asteroids dataset (August 2018)

Shivangi Srivastava, *Machine Learning Intern, AITS*

Link to AITS website: ai-techsystems.com

***Abstract*—This project report discusses the classification technique in machine learning using the decision tree algorithm and compares the performance of this algorithm with the PCA (principal component analysis) algorithm. The data is about Asteroids - NeoWs. NeoWs (Near Earth Object Web Service) is a RESTful web service for near earth Asteroid information. With NeoWs a user can: search for Asteroids based on their closest approach date to Earth, lookup a specific Asteroid with its NASA JPL small body id, as well as browse the overall data-set.**

***Index Terms*—Asteroids, Decision tree algorithm, PCA**

I. Introduction

This project report is a comparison between the performance of the model trained using the original data of the asteroids dataset and the principal components obtained after applying the PCA on the dataset.

The variation in the results obtained from both the algorithms has been analyzed and conclusions have been drawn on that basis.

II. Steps involved in making the project

A. Introduction to the dataset used

The dataset used in this project has been taken from Kaggle. The link to which is https://www.kaggle.com/shrutimehta/nasa-asteroids-classification. The dataset contains various columns containing the details linked to some asteroids. On the basis of these features, one has to predict whether a specific asteroid is dangerous or not. It is a simple classification problem, which requires the data to be classified into 2 categories – dangerous asteroids and non-dangerous asteroids.

B. Algorithms used

The machine learning algorithm that has been used to treat the raw data in this project is “Decision tree algorithm”. The reason behind the selection of this algorithm is that among all the other classification algorithms that were used on this dataset, it gave the maximum accuracy rate of 93.25%. The dataset was divided into training set and test set in the ratio of 4:1.

After applying the decision tree algorithm, principal component analysis (PCA) was done on the dataset, using which the accuracy obtained was nearly 81%. As is clear from the results, the performance of the model dropped drastically on the application of PCA. The reason behind this has been discussed in the upcoming sections.

C. Columns in the dataset

[1] The various columns present in the dataset are:

1. Neo Reference ID
2. Name
3. Absolute Magnitude
4. Est Dia in KM(min)
5. Est Dia in KM(max)
6. Est Dia in M(min)
7. Est Dia in M(max)
8. Est Dia in Miles(min)
9. Est Dia in Miles(max)
10. Est Dia in Feet(min)
11. Est Dia in Feet(max)
12. Close Approach Date
13. Epoch Date Close Approach
14. Relative Velocity km per sec
15. Relative Velocity km per hr
16. Miles per hour
17. Miss Dist.(Astronomical)
18. Miss Dist.(lunar)
19. Miss Dist.(kilometers)
20. Miss Dist.(miles)
21. Orbiting Body
22. Orbit ID
23. Orbit Determination Date
24. Orbit Uncertainity
25. Minimum Orbit Intersection
26. Jupiter Tisserand Invariant
27. Epoch Osculation
28. Eccentricity

29. Semi Major Axis
30. Inclination
31. Asc Node Longitude
32. Orbital Period
33. Perihelion Distance
34. Perihelion Arg
35. Aphelion Dist
36. Perihelion Time
37. Mean Anomaly
38. Mean Motion
39. Equinox
40. Hazardous

Some columns like 'Neo reference ID', 'Name', etc, were not used in training the dataset as they don't decide the hazardous nature of an asteroid.

The last column 'Hazardous' is the dependent variable as it tells whether or not the asteroid is dangerous.

Rest other columns in the dataset were treated as independent variables and were used in training the dataset.

D. Decision tree algortihm

Decision Tree algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, decision tree algorithm can be used for solving **regression and classification problems** too.

The general motive of using Decision Tree is to create a training model which can use to predict class or value of target variables by **learning decision rules** inferred from prior data(training data).

The understanding level of Decision Trees algorithm is so easy compared with other classification algorithms. The decision tree algorithm tries to solve the problem, by using tree representation. Each **internal node** of the tree corresponds to an attribute, and each **leaf node** corresponds to a class label. The pseudo-code for the same goes is as follows:

1. Place the best attribute of the dataset at the **root** of the tree.
2. Split the training set into **subsets**. Subsets should be made in such a way that each subset contains data with the same value for an attribute.
3. Repeat step 1 and step 2 on each subset until you find **leaf nodes** in all the branches of the tree.
4. In decision trees, for predicting a class label for a record we start from the **root** of the tree. We compare the values of the root attribute with record's attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.
5. We continue comparing our record's attribute values with other **internal nodes** of the tree until we reach **a leaf node** with predicted class value. As we know how the modeled decision tree can be used to predict the target class or the value. Now let's understanding how we can create the decision tree model.

Assumptions while making a decision tree model

The below are the some of the assumptions we make while using Decision tree:

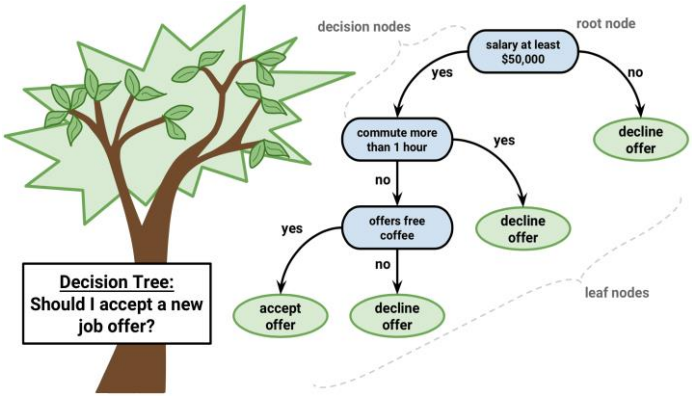
- At the beginning, the whole training set is considered as the **root.**
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- Records are **distributed recursively** on the basis of attribute values.
- Order to placing attributes as root or internal node of the tree is done by using some statistical approach.

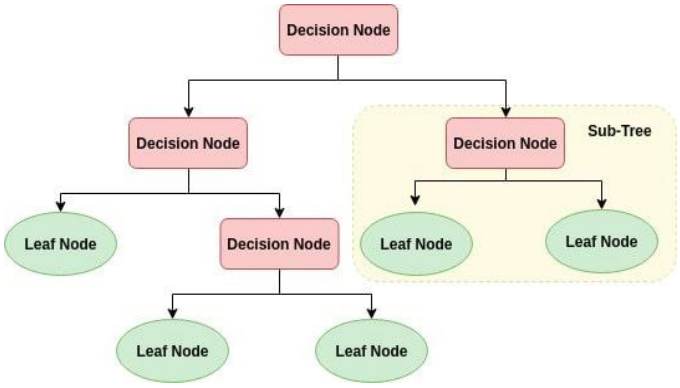
Advantages

1. Decision Trees are easy to explain. It results in a set of rules.
2. It follows the same approach as humans generally follow while making decisions.
3. Interpretation of a complex Decision Tree model can be simplified by its visualizations. Even a naive person can understand logic.
4. The Number of hyper-parameters to be tuned is almost null.

Disadvantages

1. There is a high probability of overfitting in Decision Tree.
2. Generally, it gives low prediction accuracy for a dataset as compared to other machine learning algorithms.
3. Information gain in a decision tree with categorical variables gives a biased response for attributes with greater no. of categories.
4. Calculations can become complex when there are many class labels.





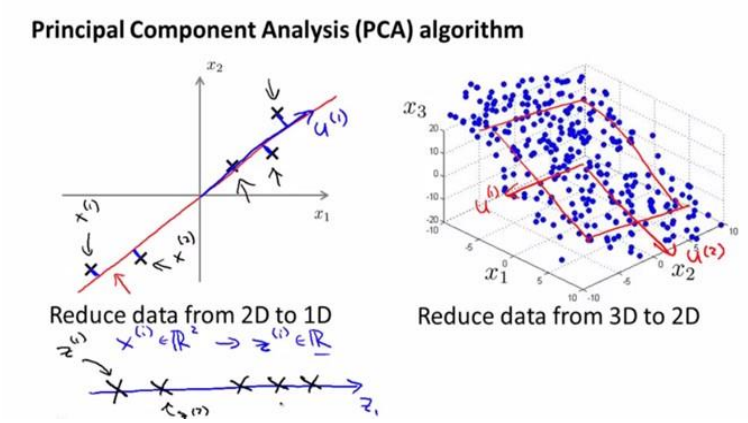
E. Principal Component Analysis (PCA)

The main idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of many variables correlated with each other, either heavily or lightly, while retaining the variation present in the dataset, up to the maximum extent. The same is done by transforming the variables to a new set of variables, which are known as the principal components (or simply, the PCs) and are orthogonal, ordered such that the retention of variation present in the original variables decreases as we move down in the order. So, in this way, the 1st principal component retains maximum variation that was present in the original components. The principal components are the eigenvectors of a covariance matrix, and hence they are orthogonal.

Importantly, the dataset on which PCA technique is to be used must be scaled. The results are also sensitive to the relative scaling. As a layman, it is a method of summarizing data. Imagine some wine bottles on a dining table. Each wine is described by its attributes like colour, strength, age, etc. But redundancy will arise because many of them will measure related properties. So what PCA will do in this case is summarize each wine in the stock with less characteristics.

Intuitively, Principal Component Analysis can supply the user with a lower-dimensional picture, a projection or "shadow" of this object when viewed from its most informative viewpoint.

F. Figures and Tables



G. Drop in performance on using PCA

[2] This drastic drop in performance is due to the fact that the PCA algorithm only considers some of the features present in the dataset. Using PCA can lose some spatial information which is important for classification, so the classification accuracy decreases. This occurs because the new features space are linear combination from original features. Therefore, may there is loss of information.

III. REFERENCES

[1]https://www.kaggle.com/shrutimehta/nasa-asteroids-classification

[2]https://www.researchgate.net/post/Is_there_a_specific_reason_that_using_PCA_gives_worse_results_than_without_using_it_in_SVM_classification