Feature Selection Techniques to predict the religion of a Country from its Flag

Project submitted in partial fulfillment for the award of the degree of Master of Computer Application

Submitted by

Ashis Dhala Samanta (1970042)

Under the Guidance of

Partha Sarathi Pattnaik

Asst. Professor

April 2022

CERTIFICATE OF ORIGINALITY

This is to certify that the project report entitled

Feature Selection Techniques to predict the religion of a Country from its Flag

Submitted to School of Computer Applications, KIIT University in partial fulfilment of the requirement for the award of the degree of MASTER OF COMPUTER APPLICATION (MCA), is an authentic and original work carried out by Mr. Ashis Dhala Samanta with Roll no. 1970042 and Regd. No. 19266568749 under my guidance.

The matter embodied in this project is genuine work done by the student and has not been submitted whether to this University or to any other University / Institute for the fulfilment of the requirements of any course of study.

Ashis Dhala Samanta

Signature of the Student: Signature of the Guide

Date: 29th April 2022 Date: 29th April 2022

Name: Partha Sarathi Pattnaik

Asst. Professor

School of Computer Applications

KIIT University, Bhubaneswar

CERTIFICATE

This is to certify that the project work entitled Feature Selection Techniques to predict the religion of a Country from its Flag

Submitted by Ashis Dhala Samanta bearing roll no. 1970042, is an authentic and original work.

Signature	Signature
(Internal Examiner)	(External Examiner)
Date 29th April 2022	Date 29th April 2022

Declaration

I, Ashis Dhala Samanta, 1970042 do hereby declare that the project report entitled Feature Selection Techniques to predict the religion of a Country from its Flag submitted to School of Computer Applications, KIIT University, Bhubaneswar for the award of the degree of MASTER OF COMPUTER APPLICATION (MCA), is an authentic and original work carried out by me from 1st Jan 2022 to 20th April 2022 under the guidance of Partha Sarathi Patnaik sir.

Ashis Dhala Samanta

Signature of the student

Date 29th April 2022

Acknowledgement

I would like to express my gratitude and appreciation to all those who gave me the possibilities to complete this project. Special thanks are due to my supervisor Partha Sarathi Pattnaik, Asst. Professor, who help, stimulating suggestions and encouragement helped me in all time of fabrication process and in writing this report.

I would also like to acknowledge with much appreciation the crucial role of the staff in School of Computer Applications who helped me to improve my skills and potential. I would also like to extend my gratitude to the Director ma'am Dr. Veena Goswami for providing me with all the facilities that was required

Ashis Dhala Samanta

Date 29th April 2022 Signature of the student

Place Bhubaneswar Name of the student

Roll_no: 1970042

Table of Contents

1.Problem statement

The challenge is aimed at making use of machine learning and artificial intelligence in interpreting flag dataset. The dataset made available the country's religion from the colour of flag.

Feature Selection is a process of preparing data to be more effective and efficient for machine learning problems.

The purpose of feature selection is to select relevant features from huge no of features. Also to build simple model that is easy to understand data and takes less time to train the model and increase model performance.

The paper proposes two feature selection techniques namely Lasso and Select From Model (meta-transformer) to select relevant features from flag dataset that intensifies the model performance.

For prediction of religion of a country, three tree based classifiers are used - Random Forest, Decision Tree and Extra Trees model. Among these, Random Forest Classifier gives best prediction.

Keywords: Feature Selection, Decision Tree, Lasso, Random Forest, Extra Trees.

2. Introduction

Machine Learning

Machine learning is a subfield of artificial intelligence, which is broadly defined as the capability of a machine to imitate intelligent human behavior. Artificial intelligence systems are used to perform complex tasks in a way that is similar to how humans solve problems.

Examples: Image recognition is a well-known and widespread example of machine learning in the real world. It can identify an object as a digital image, based on the intensity of the pixels in black and white images or colour images.

Types: As new data is fed to these algorithms, they learn and optimize their operations to improve performance, developing 'intelligence' over time.

There are four types of machine learning algorithms: supervised, semi-supervised, unsupervised and reinforcement.

Supervised learning

Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher.

Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y).

In the real-world, supervised learning can be used for Risk Assessment, Image classification, Fraud Detection, spam filtering, etc.

In supervised learning, models are trained using labelled dataset, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.

Semi-supervised learning

In Semi-supervised learning, the algorithm is trained upon a combination of labeled and unlabeled data. Typically, this combination will contain a very small amount of labeled data and a very large amount of unlabeled data.

The basic procedure involved is that first, the programmer will cluster similar data using an unsupervised learning algorithm and then use the existing labeled data to label the rest of the unlabeled data.

The typical use cases of such type of algorithm have a common property among them – The acquisition of unlabeled data is relatively cheap while labeling the said data is very expensive.

Intuitively, one may imagine the three types of learning algorithms as Supervised learning where a student is under the supervision of a teacher at both home and school, Unsupervised learning where a student has to figure out a concept himself and Semi-Supervised learning where a teacher teaches a few concepts in class and gives questions as homework which are based on similar concepts.

A Semi-Supervised algorithm assumes the following about the data- Continuity Assumption: The algorithm assumes that the points which are closer to each other are more likely to have the same output label.

Cluster Assumption: The data can be divided into discrete clusters and points in the same cluster are more likely to share an output label.

Manifold Assumption: The data lie approximately on a manifold of much lower dimension than the input space. This assumption allows the use of distances and densities which are defined on a manifold.

Unsupervised learning

Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision.

Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.

Example: Suppose the unsupervised learning algorithm is given an input dataset containing images of different types of cats and dogs. The algorithm is never trained upon the given dataset, which means it does not have any idea about the features of the dataset.

The task of the unsupervised learning algorithm is to identify the image features on their own. Unsupervised learning algorithm will perform this task by clustering the image dataset into the groups according to similarities between images.

Reinforcement learning

It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation. Reinforcement learning differs from supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task.

In the absence of a training dataset, it is bound to learn from its experience.

Main points in Reinforcement learning –

Input: The input should be an initial state from which the model will start

Output: There are many possible outputs as there are a variety of solutions to a particular problem

Training: The training is based upon the input, The model will return a state and the user will decide to reward or punish the model based on its output.

The model keeps continues to learn.

The best solution is decided based on the maximum reward.

2.1 Classification

Classification is a process of categorizing a given set of data into classes, it can be performed on both structured or unstructured data.

The process starts with predicting the class of given data points. The classes are often referred to as target, label or category. The classification predictive modeling is the task of approximating the mapping function from input variables to discrete output variables. The main goal is to identify which class/category the new data will fall into.

In machine learning, classification is a supervised learning concept which basically categorizes a set of data into classes. The most common classification problems are – speech recognition, face detection, handwriting recognition, document classification, etc. It can be either a binary classification problem or a multi-class problem too.

There are a bunch of machine learning algorithms for classification in machine learning. Let us take a look at those classification algorithms in machine learning.

A common job of machine learning algorithms is to recognize objects and being able to separate them into categories. This process is called classification, and it helps us segregate vast quantities of data into discrete values, i.e.: distinct, like 0/1, True/False, or a pre-defined output label class.

There are four types of classification. They are Geographical classification, Chronological classification, Qualitative classification, Quantitative classification.

Classification Terminologies in Machine Learning

Classifier – It is an algorithm that is used to map the input data to a specific category.

Classification Model – The model predicts or draws a conclusion to the input data given for training, it will predict the class or category for the data.

Feature – A feature is an individual measurable property of the phenomenon being observed.

Binary Classification – It is a type of classification with two outcomes, for e.g. – either true or false.

Multi-Class Classification – The classification with more than two classes, in multi-class classification each sample is assigned to one and only one label or target.

Multi-label Classification – This is a type of classification where each sample is assigned to a set of labels or targets.

Train the Classifier – Each classifier in sci-kit learn uses the fit(X, y) method to fit the model for training the train X and train label y.

Predict the Target – For an unlabeled observation X, the predict(X) method returns predicted label y.

Evaluate – This basically means the evaluation of the model i.e classification report, accuracy score, etc.

2.2 Clustering

Clustering or cluster analysis is a machine learning technique, which groups the unlabeled dataset. It can be defined as "A way of grouping the data points into different clusters, consisting of similar data points. The objects with the possible similarities remain in a group that has less or no similarities with another group."

It does it by finding some similar patterns in the unlabeled dataset such as shape, size, color, behavior, etc., and divides them as per the presence and absence of those similar patterns.

It is an unsupervised learning method, hence no supervision is provided to the algorithm, and it deals with the unlabeled dataset.

After applying this clustering technique, each cluster or group is provided with a cluster-ID. ML system can use this id to simplify the processing of large and complex datasets.

The clustering technique is commonly used for statistical data analysis. Clustering is somewhere similar to the classification algorithm, but the difference is the type of dataset that we are using. In classification, we work with the labeled data set, whereas in clustering, we work with the unlabeled dataset.

The clustering technique can be widely used in various tasks. Some most common uses of this technique are:

Market Segmentation

Statistical data analysis

Social network analysis

Image segmentation

Anomaly detection, etc.

2.3 Support Vector Machine (SVM)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane.

SVM can be of two types:

Linear SVM: Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

Non-linear SVM: Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

Hyperplane and Support Vectors in the SVM algorithm

Hyperplane

There can be multiple lines/decision boundaries to segregate the classes in n-dimensional space, but we need to find out the best decision boundary that helps to classify the data points. This best boundary is known as the hyperplane of SVM.

The dimensions of the hyperplane depend on the features present in the dataset, which means if there are 2 features (as shown in image), then hyperplane will be a straight line. And if there are 3 features, then hyperplane will be a 2-dimension plane.

We always create a hyperplane that has a maximum margin, which means the maximum distance between the data points.

Support Vectors:

The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.

Decision Tree Classification Algorithm

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems.

It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset.

It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.

It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.

In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm.

A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

A decision tree can contain categorical data (YES/NO) as well as numeric data.

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision tree:

Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.

The logic behind the decision tree can be easily understood because it shows a tree-like structure.

Decision Tree Terminologies

Root Node: Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.

Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.

Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.

Branch/Sub Tree: A tree formed by splitting the tree.

Pruning: Pruning is the process of removing the unwanted branches from the tree.

Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

How does the Decision Tree algorithm Work?

In a decision tree, for predicting the class of the given dataset, the algorithm starts from the root node of the tree.

This algorithm compares the values of root attribute with the record (real dataset) attribute and, based on the comparison, follows the branch and jumps to the next node.

For the next node, the algorithm again compares the attribute value with the other sub-nodes and move further. It continues the process until it reaches the leaf node of the tree. The complete process can be better understood using the below algorithm

Step-1: Begin the tree with the root node, says S, which contains the complete dataset.

Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).

Step-3: Divide the S into subsets that contains possible values for the best attributes.

Step-4: Generate the decision tree node, which contains the best attribute.

Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

2.4 Support Vector Machine in Kernlab

The plot function for binary classification KSVM objects displays a contour plot of the decision values with the corresponding support vectors highlighted. The predict function can return class probabilities for classification problems by setting the type parameter to "probabilities". KSVM shows minimum number of support vectors than SVM.

The comparison of SVM and KSVM was made for the twitter data set. The experimental results show that the number of support vectors by KSVM is very low when compare with normal SVM.

Random Forest Tree

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. "Instead of relying on

one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Below are some points that explain why we should use the Random Forest algorithm:

It takes less training time as compared to other algorithms.

It predicts output with high accuracy, even for the large dataset it runs efficiently.

It can also maintain accuracy when a large proportion of data is missing.

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

Applications of Random Forest

There are mainly four sectors where Random forest mostly used:

Banking: Banking sector mostly uses this algorithm for the identification of loan risk.

Medicine: With the help of this algorithm, disease trends and risks of the disease can be identified.

Land Use: We can identify the areas of similar land use by this algorithm.

Marketing: Marketing trends can be identified using this algorithm.

Advantages of Random Forest

Random Forest is capable of performing both Classification and Regression tasks.

It is capable of handling large datasets with high dimensionality.

It enhances the accuracy of the model and prevents the overfitting issue.

Disadvantages of Random Forest

Although random forest can be used for both classification and regression tasks, it is not more suitable for Regression tasks.

3 Methodology

This section explains the general framework in of the paper. The framework constitutes dataset and the machine learning techniques used for feature selection as well as for prediction of religion.

The approach follows the process: Using training dataset, feature selection by SelectFromModel and Lasso are implemented. The output from the feature selection process is used by tree-based classifiers that are used to predict religion of the country.

Finally, accuracy results of classifiers obtained by taking features which are selected by said techniques is compared with accuracy results obtained by taking all features without any selection technique.

3.1 Dataset

The first step consists of collecting dataset of flags from UCI machine learning repository. The dataset consists of many country names and their flag details.

Each flag's information is considered as an attribute (feature). In this dataset there are 30 features of 194 countries. Here, ten features are numeric-valued and others are Boolean or nominal-valued. The features are Country_Name(Sl.No), Land mass, Zone, Area, Population, Language, Bars, Stripes, Colour, Red, Green, Blue, Gold, White, Black, Orange, Mainhue, Circles, Crosses, Saltires, Quarters, Sunstars, Crescent, Triangle, Icon, Animate, Text, Topleft, Botright and Religion As our problem statement is to predict religion of country, the religion needs to be categorized. The categorization is as follows: 0=Catholic, 1=Other Christian, 2=Muslim, 3=Buddhist, 4=Hindu, 5=Ethnic, 6=Marxist and 7=Others. So we find importance of rest of the features besides religion.

By using random forest classifier, we fit the model and find importance of other features. Figure-1 shows importance of different features mentioned in our study. X and Y axis represents name of the features and importance range respectively.

It shows Landmass has more importance priority over others and Saltires, Text has less importance than others.

3.2 Feature Selection

Feature selection also called as variable selection or attributes selection. Feature selection is a process of selecting relevant features or we can say reducing irrelevant or partially relevant features without loss of total information.

It also helps to understand desired features and their importance which makes model simple and easy to explain.

Feature selection also helps to know irrelevant and redundant attributes that may have negative impact on model performance [decrease accuracy of model]. Some benefits of feature selection prior to(before) model your data are: less data means that reduces training time, less misleading data that improves model performance, less redundant data means that reduces overfitting.

There are three types of algorithms for feature selection: Filter methods, Wrapper methods, Embedded methods. In our problem we used Lasso and Tree based feature selection using Feature Importance (using SelectFromModel). Lasso is embedded method of feature selection.

3.2.1 Feature Selection: Lasso

LASSO (Least Absolute Shrinkage and Select ion Operator) operates L1 regularization that adds penalty equivalent to absolute value of the magnitude of coefficients. It takes alpha (α) as a parameter which presents trade-off between balancing residuals of sum of square and magnitude of coefficients. The α can take any value (0, ∞, 0< α< ∞). In Lasso for high value of α, few features are selected. Because of most of the coefficients become zero i.e. called sparsity.

The features with zero coefficients are excluded from the model. Where the no. of features are in millions, the sparse solution provides computational advantages to the model.

In our problem we use LassoLarsCV to find the value of α and get α=0.1. Figure -2 shows model coefficients for Lasso regression where alpha value is 0.1. X and Y axis represents features name and scores respectively.

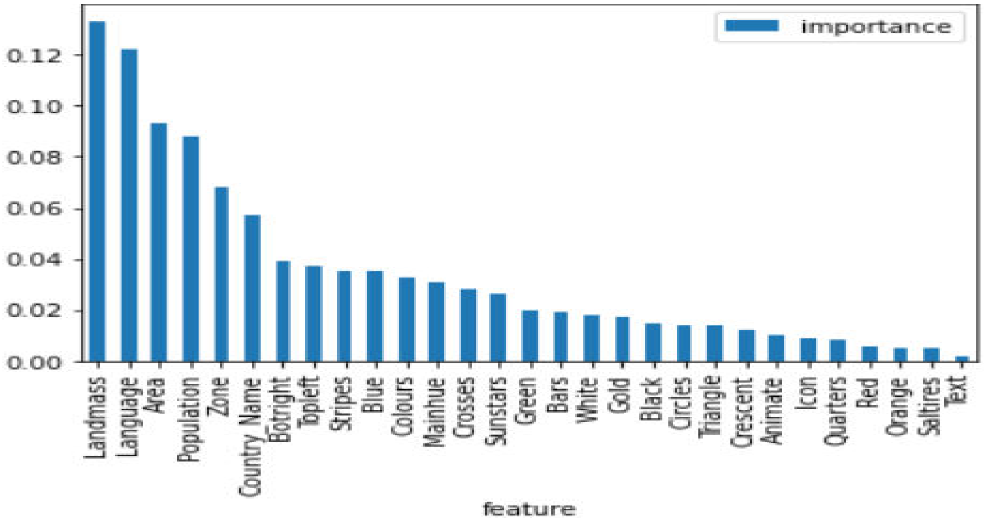


Figure-1 Importance of different features

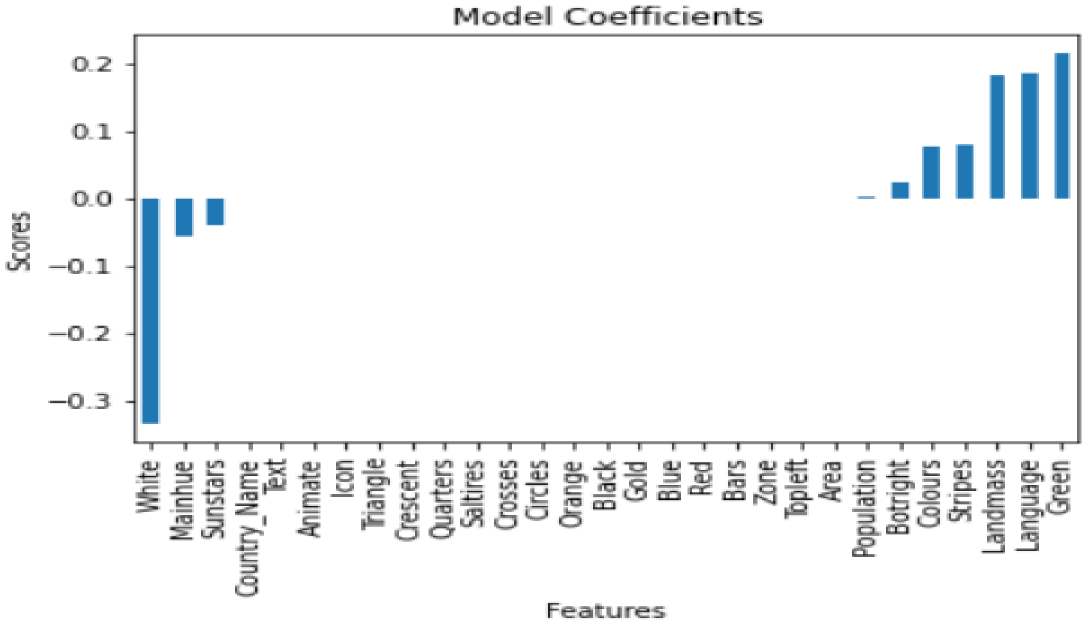


Figure-2 model coefficients for Lasso regression

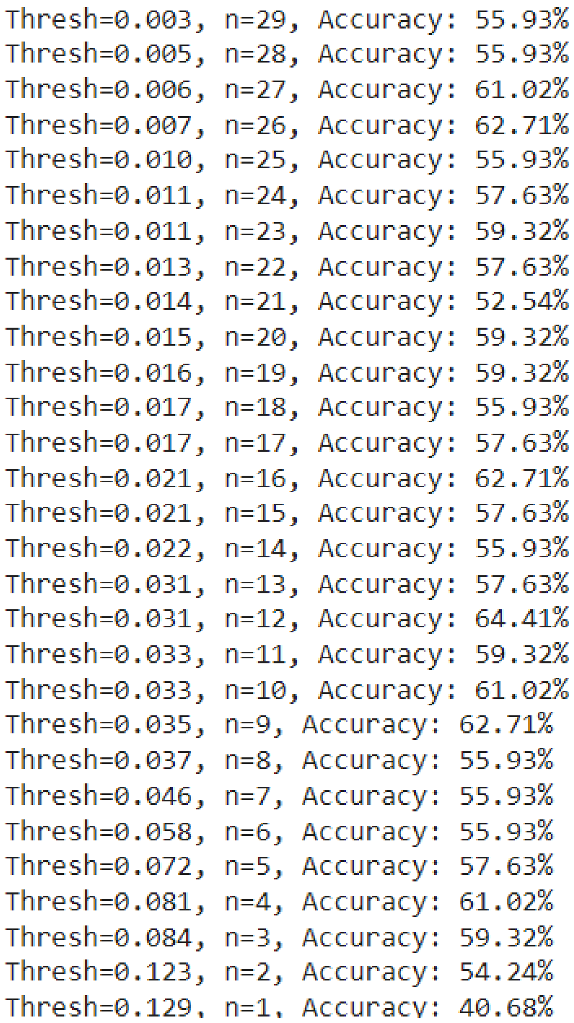
From this figure, we consider the case of sparsity. Maximum coefficients of features are zero. So only ten features: White, Mainhue , Sunstars, Population, Botright, Colours, Stripes, Language, Land mass, Green are selected to fit into the model.

3.2.2 Feature Selection: Tree Based

Tree based estimators i.e. RandomForestClassifier, ExtratreesClassifier used to measure features' importance. SelectFromModel is a meta-transformer used with estimator that has coef_ or feature_importances_ attribute after fitting .

In our problem we used tree-based estimator i.e., RandomForestClassifier followed by SelectFromModel for choosing features using features' importance. Those features are selected which has coef_ or feature_importances values greater than given threshold value.

The threshold value may be, mean of the importance, median of importance, a float value or none. Next step is how we set threshold value. For this follow figure-3 which shows model (RandomForestClassifier) performance with different threshold values. Here each feature importance which is evaluated by RandomForestClassifier treated as a threshold value. For each threshold value we get the accuracy. Here n is the no. of features and Thresh is Threshold value.



```
Thresh=0.003, n=29, Accuracy: 55.93%
Thresh=0.005, n=28, Accuracy: 55.93%
Thresh=0.006, n=27, Accuracy: 61.02%
Thresh=0.007, n=26, Accuracy: 62.71%
Thresh=0.010, n=25, Accuracy: 55.93%
Thresh=0.011, n=24, Accuracy: 57.63%
Thresh=0.011, n=23, Accuracy: 59.32%
Thresh=0.013, n=22, Accuracy: 57.63%
Thresh=0.014, n=21, Accuracy: 52.54%
Thresh=0.015, n=20, Accuracy: 59.32%
Thresh=0.016, n=19, Accuracy: 59.32%
Thresh=0.017, n=18, Accuracy: 55.93%
Thresh=0.017, n=17, Accuracy: 57.63%
Thresh=0.021, n=16, Accuracy: 62.71%
Thresh=0.021, n=15, Accuracy: 57.63%
Thresh=0.022, n=14, Accuracy: 55.93%
Thresh=0.031, n=13, Accuracy: 57.63%
Thresh=0.031, n=12, Accuracy: 64.41%
Thresh=0.033, n=11, Accuracy: 59.32%
Thresh=0.033, n=10, Accuracy: 61.02%
Thresh=0.035, n=9, Accuracy: 62.71%
Thresh=0.037, n=8, Accuracy: 55.93%
Thresh=0.046, n=7, Accuracy: 55.93%
Thresh=0.058, n=6, Accuracy: 55.93%
Thresh=0.072, n=5, Accuracy: 57.63%
Thresh=0.081, n=4, Accuracy: 61.02%
Thresh=0.084, n=3, Accuracy: 59.32%
Thresh=0.123, n=2, Accuracy: 54.24%
Thresh=0.129, n=1, Accuracy: 40.68%
```

Figure-3 Model performance with different threshold values

From figure-3, we figured out for threshold 0.031 and no. of features 12, accuracy is highest at 64.41%. After that as no. of features increases accuracy decreases. So why should we go for more no. of features? So, we set threshold value to 0.031.

As threshold value varies, we get different set of features. Figure-4 to figure-6 explains different set of features selected by different threshold values. X and Y axis represents features name and scores respectively.

Figure-4 shows set of features where threshold is median of importance. The selected features are Bars, Stripes, Orange, Topleft ,Text , Landmass, Area, Zone, Country_name(Sl.No.) and Population. Population has highest importance than others. Figure -5 shows set of features where threshold is mean of importance. The selected features are Bars, Orange, Text, Topleft , Landmass, Area, Zone Country_name and Population. Population has highest importance than others. Figure-6 shows set of features where threshold is 0. 030. The selected features are Bars, Circles, Orange, Stripes, Topleft , Text, Landmass, Area, Zone, Country_name(Sl.No) and Population. Population has highest importance than others. Figure -7 shows set of features where threshold is none. The selected features are Bars, Topleft , Text, Landmass, Area, Zone, Country_name(Sl.No.) and Population. Country_name(Sl.No.) has highest importance than others.

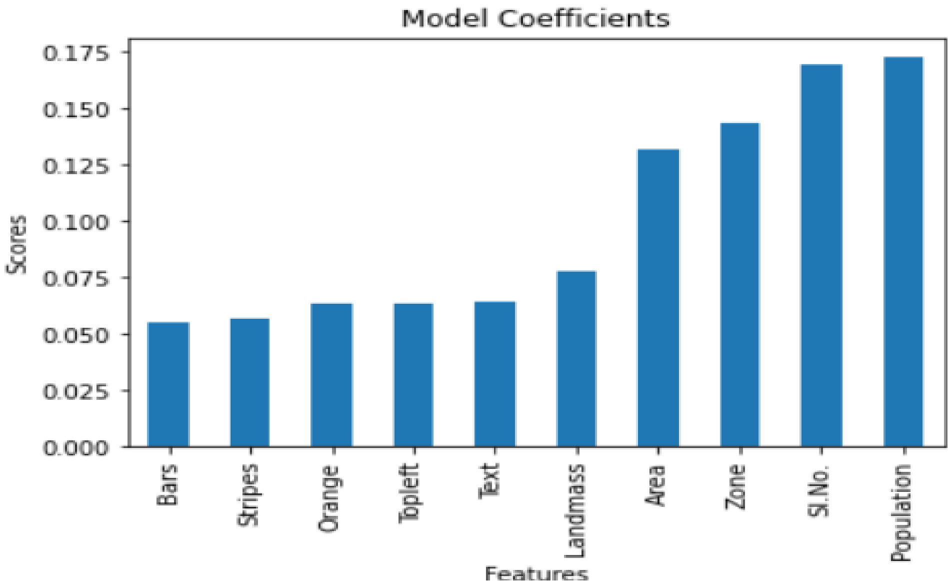


Figure-4 Features where thresh = Median

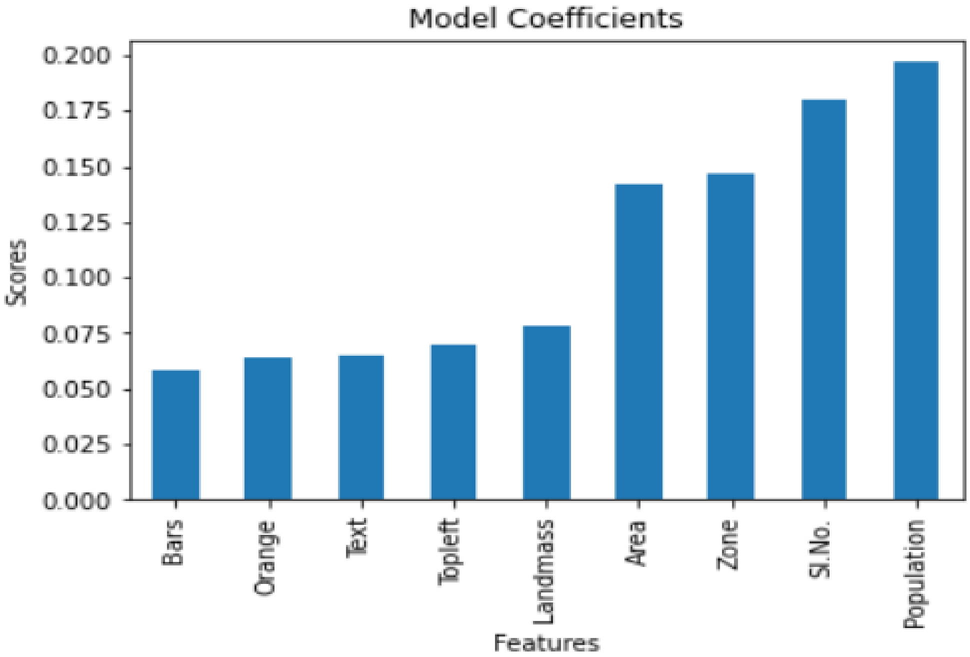


Figure-5 Features where thresh=Mean

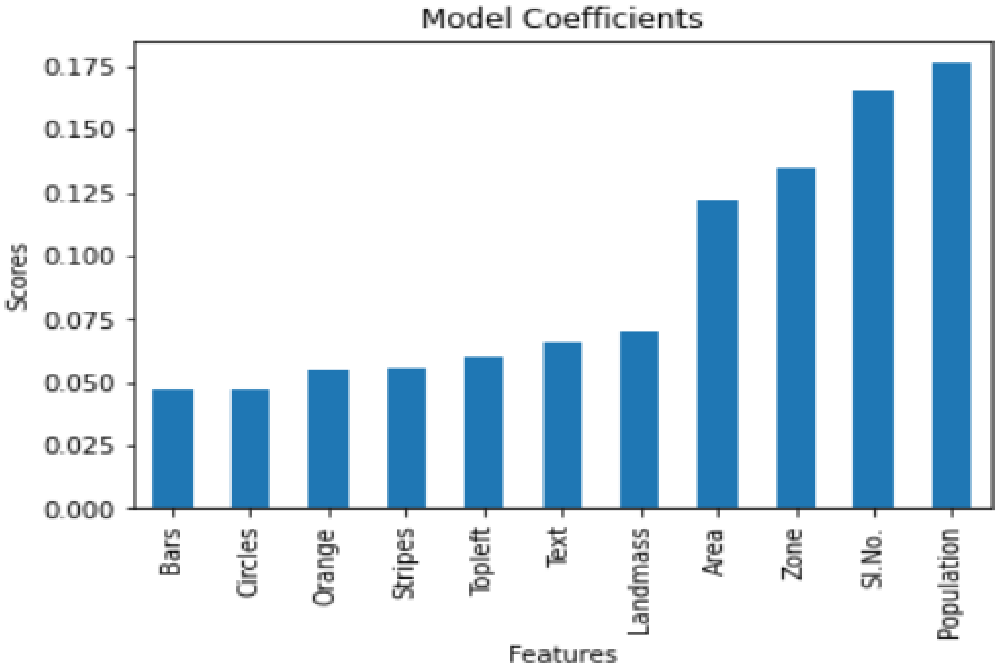


Figure-6 Features where thresh = 0.31

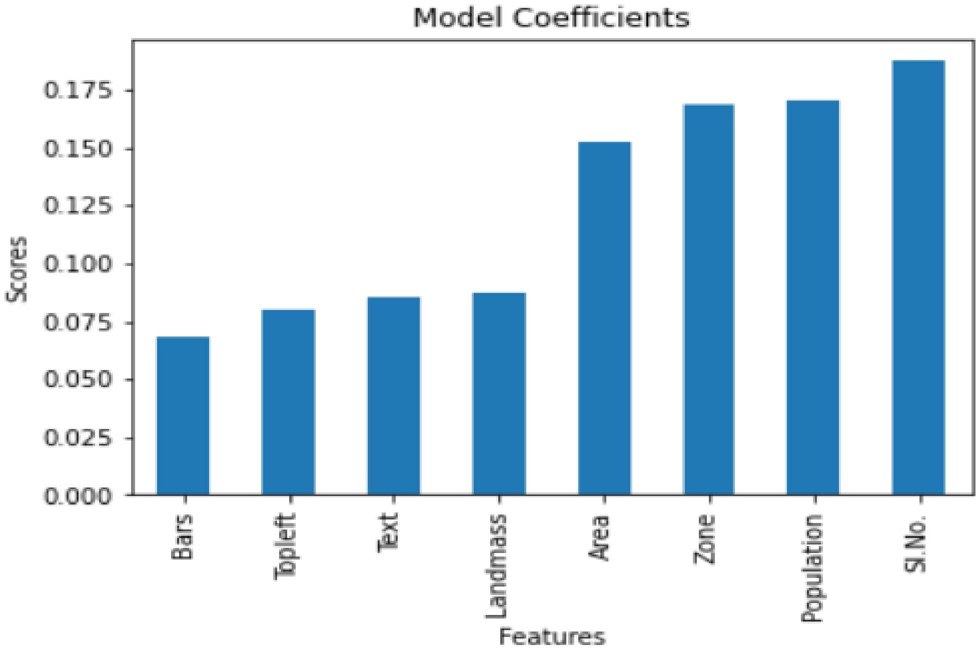


Figure-7 Features where thresh = None

After selecting relevant features we process our dataset with these features using tree based classifiers, Random Forest Classifier, Extra Trees Classifier and Decision Tree Classifier and predict the religion of a country.

3.3 Decision Tree Classifier

Decision Tree looks like a hierarchical structure (top -down approach) which is composed of nodes and edges. As we know Decision tree has many parts like root, internal node, terminal node and edges.

The terminal node (leaf node) has result that means class label. The root and internal node decide the records belong to what class according to their attributes and their rules.

Otherwise we can say Decision Tree Classifier raises a series of thoroughly articulated questions about the test record features. Each time it gets an answer, a follow -up question is triggered until class label of the record is finalized.

There are two advantages of decision trees: (a) the extension of level of the tree to facilitate it to classify training dataset accurately (b) the pruning stage, in order to increase classification accuracy.

3.4 Random Forest Classifier

Random forest classifier is an ensemble algorithm. It creates a forest of Decision trees from randomly selected training dataset where the features of test object must be present in each Decision tree of the forest.

The final classification value of test object is decided by aggregating the votes returned by different Decision trees. Each Decision tree is created by following rules:

a) If N is the no. of objects in a data set, then Random Forest selects training set randomly of N objects from the original data set.

b) If M is the no. of attributes in a data set, then value m is smaller than M. This m value is constant until forest building.

c) At each node of the tree, the split criterion is computed on randomly selected m attributes. The attribute with the best result is used to split the node. Gini index was the original split criterion used by Random Forest.

d) There is no pruning after building each Decision tree.

3.5 Extra Trees Classifier

An Extra Trees classifier is a variant of rando m forest. It is also called as Extremely randomized trees. The Extra Trees differ from Random Forest as follows:

a)The bagging procedure may not be applied by Extremely randomized trees to construct a set of the training samples for each tree.

The same input training set is used to train all trees.

b) Extremely randomized trees take a node split where attribute index and attribute splitting value are chosen randomly.

c) For high no. of noisy features Extra Trees gives worst performance.

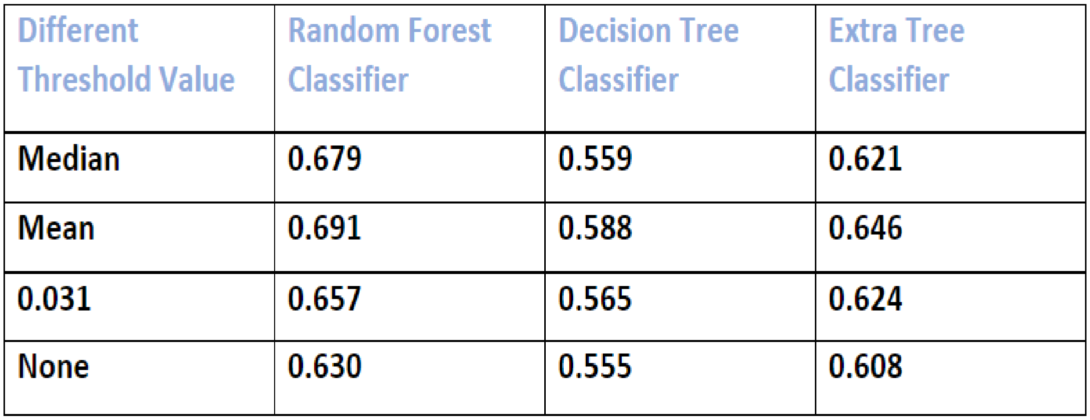
d) For provided optimal feature selection, Extra Trees can be calculated faster. From bias/variance analysis it can be concluded that with an increased randomization to an optimal level, there is a slight decrease in variance with a significant increase in bias.

4. Result and Analysis

In this section we present the performance results of proposed classifiers (as discussed above) based on features selected by Lasso and SelectFromModel. We also explain some results based on different threshold value taken by SelectFromModel.

Here we also observe that when all features are considered, accuracy is decreased as compared to when relevant features are selected. Table 1 shows accuracy results of classifiers (RandomForest, DecisionTree, ExtraTrees) with different threshold values (mean, median, 0.031, None). For threshold value mean, it gives best results to predict the religion of a country for all classifiers 0.691, 0.588 and 0.646 respectively than others. For threshold value None, it gives lowest results for all classifiers 0.630, 0.555, 0.608 respectively than others. Table 2 shows of accuracy results (to

predict the religion of country) of classifiers with feature selection (by Lasso and SelectFromModel) and without feature selection.



Different Threshold Value	Random Forest Classifier	Decision Tree Classifier	Extra Tree Classifier
Median	0.679	0.559	0.621
Mean	0.691	0.588	0.646
0.031	0.657	0.565	0.624
None	0.630	0.555	0.608

Table 1: Accuracy result of classifiers for various threshold values

	Random Forest Classifier	Decision Tree Classifier	Extra Tree Classifier
Select_From_Model	**0.691**	**0.588**	**0.646**
Lasso	**0.671**	**0.62**	**0.591**
No. Feature Selection	**0.635**	**0.565**	**0.628**

Table 2: Accuracy results of Classifiers with feature selection and without feature selection

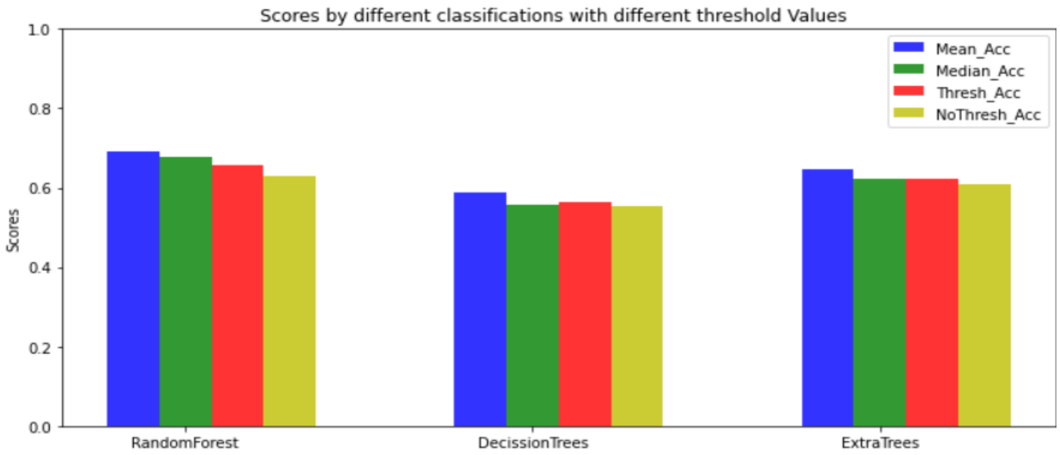


Figure-8 Compare accuracy of classifiers with different threshold values

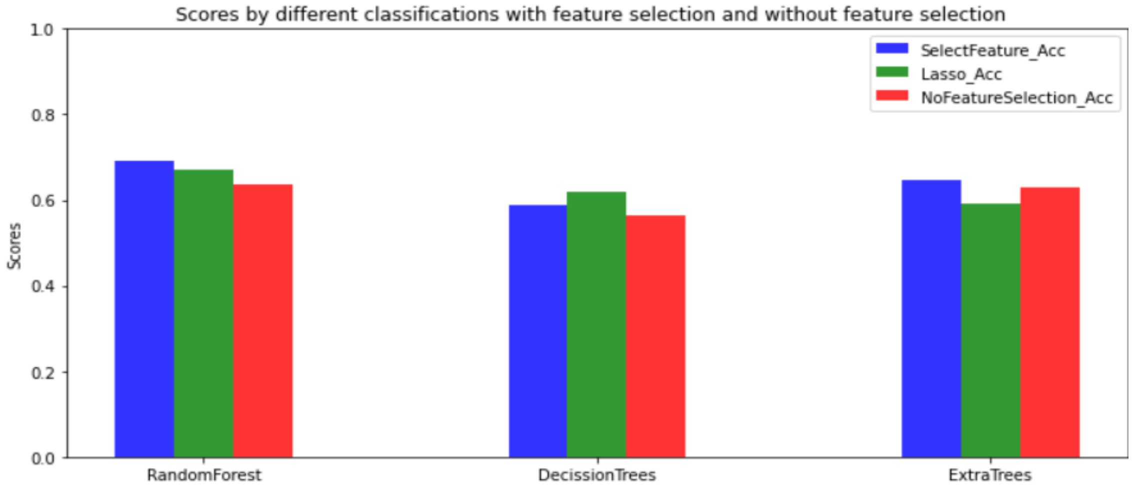


Figure-9 Compare accuracy of classifiers with feature selection and without feature selection

RandomForestClassifier gives best result compared to other two classifiers for any scenario(SelectFromModel, Lasso, No Feature Selection). Features selected by SelectFromModel are fitted into RandomForest gives highest accuracy of 0.691 among all results.

For RandomForest and ExtraTrees , SelectFromModel gives best result of 0.691 and 0.646 respectively than other two. For DecissionTree, Lasso gives best result of 0.62 over other two.

Figure-8 and Figure -9 are graphical rep resentation of Table 1 and Table 2 respectively. Figure-8 shows comparison of accuracy results of classifiers (RandomForest, DecisionTree, ExtraTrees) with different threshold values.

Mean threshold value gives best accuracy than other threshold values for all classifiers. So for next comparison with Lasso we take the accuracy result which is computed by mean threshold value.

Figure -9 shows comparison of accuracy results of classifiers with feature selection and without feature selection. Features selected by SelectFromModel give best accuracy than others for RandomForest and ExtraTrees. But for DecisionTree feature selected by Lasso gives best result.

For RandomForest and DecissionTree, without feature selection provides lowest accuracy than Lasso and SelectFromModel. But For ExtraTrees Lasso provides lowest accuracy than SelectFromModel and without feature selection.

5. Conclusion

Feature selection is a major issue to construct the model. This implies relevant features selected by feature selection are used to build the model. It enhances the performance of model, making it easy to understand by user, less time to train the model and reduce overfitting.

This paper proposes two methods Lasso and SelectFromModel to select relevant features from flag dataset. We use various threshold values for SelectFromModel to get different set of relevant features. We arrived at a conclusion that mean threshold value gives best result compared to other three values.

So features selected by SelectFromModel (mean threshold value) and Lasso are used to construct the model. Three tree based classifier namely RandomFotrest, DecisionTree and ExtraTrees constructed using selected features have been compared to predict the most accurate religion of a country. After implementation it was observed that Random Forest with accuracy of 0.691 more accurately predicts the religion than other classifiers.