Correlation Based Feature Selection with Clustering for Multi-Class Classification Tasks

Tanvir Alam (011 141 004)
Moumita Mahfuz (011 141 119)
Papiya Akter (011 141 164)

Department of Computer Science and Engineering

United International University

A thesis submitted for the degree of

B.Sc. in Computer Science & Engineering

May 2019

Abstract

In recent times high dimensional data is increasing rapidly. Reduce the dimensionality has become popular by feature selection process. So many scientists prefer to use correlation base feature selection method for grouping the attributes of dataset. The main purpose of feature selection is to elect the most problem related features and to remove unnecessary (noisy and redundant) features. Many types of correlation base feature selection have been proposed. In previous mutual information, correlation coefficient, and chi-square has been used to find the dependency between two features. In this paper we merge two methods to find correlation of features. First we create covariance matrix using formula for each instances. So we get covariance matrix which will be dimensional as main dataset. Then we integrate Affinity Propagation (AP) clustering for grouping the features and took random features from each cluster and append them. So we make model of that subset and use general machine learning algorithms and find accuracy then compare with the main dataset accuracy.

This Work is devoted to our beloved Parents and respected Teachers.

Acknowledgements

Firstly, We would like to express our sincere gratitude to our supervisor Dr. Dewan Md. Farid, Associate Professor, Department of Computer and Engineering, United Intentional University (UIU), for the continuous support, friendly attitude, motivation and encouragement for completing our thesis report. His guidance helped us in all imagined having a better supervisor and mentor for our thesis.

Beside our supervisor, we also thank to United International University (UIU) for providing us a wonderful environment of academic and fellow researchers. We are also thankful to Rafsanjani Muhammod and Sajid Ahmed for their useful advice in developing our ideas and spending time in giving us a foundation on which to build our research.

Last but not the least, we would like to thank our families, especially our parents for supporting us throughout completing this thesis.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Multi-Class Classification means each training point belongs to one of N different classes. If we see an example of Healthcare data set where blood pressure, weight, sugar level, cholesterol level etc are variables and they are inter-related to each other. We will get the output according to these variables value. Now-a-days most of the applications have huge amount of data in terms of features and instances. In data science dimensionality means how many features or attributes a dataset has. If a dataset has huge amount of features then we can called it a high dimensional data. If we think about Google or Facebook datasets we can assume that there data contains lot of features so called dimension. In recent years, data has become increasingly larger in both number of instances and number of features in many applications. This enormity may cause serious problems to many machine learning algorithms with respect to scalability and learning performance. To optimize the huge data correlation base feature selection has become a popular method for scientist as well as developers. Some of the recent research efforts in feature selection have been focused on these challenges from handling a huge number of instances [1] to dealing with high dimensional data [2] [3]. This work is concerned about feature selection for high dimensional data. In our work we took four datasets and proceed our feature selection process. Then apply many machine learning algorithms in updated datasets and find out the accuracy. We did get good result by our work.

1.1 Motivation

Data are increasing day by day. With increasing data unnecessary data are also increasing. Unnecessary means both noisy and redundant data. If a feature is not helping to increase classification accuracy then that feature is unnecessary feature. Now-a-days most of the application has huge amount of data in terms of features and instances. This high dimensional data are beyond human's ability to understand or handle. Finding relevant, non-redundant data from an application is a challenge for machine learning which contains hundreds to thousands of features. Feature selection is a vital technique for solving problems with high dimensional data. To work with high dimensional data there are many types of feature selection techniques such as Wrapper and Filters have been used. These techniques are not only to decrease dimension of data but also to overabundance of information, scalability and learning performance. To use machine learning algorithms with high dimensional data become tough so we go for feature selection. In this work we proposing a new feature selection technique where correlation is the base with clustering methods we create a new subset of relevant and non-redundant data.

1.2 Objectives of the Thesis

- In this thesis, we have reviewed several Co-relation base feature selection which have been proposed in the past 10 years. We have reviewed more than 25 research papers.
- This thesis describes several intrusion classification technology and it's architecture using machine learning algorithms.

1.3 Organization of the Thesis

The thesis is organized as follows:

Chapter 2 Provides review of related worked papers.

Chapter 3 Explain the commonly used machine learning algorithms in intrusion detection system with experimental analysis and evaluation of the models.

Chapter 4 Provides datasets description and result.

Chapter 5 presents the conclusions, summaries the thesis contributions, and discusses the future works.

Chapter 2

Related Work

Feature selection is a data preprocessing technique. By removing irrelevant and redundant data from dataset increase machine learning algorithm performance [4]. We applied feature selection methods in classification problems to select a minimize feature set that is more accurate and classified by learning algorithms faster. There are three or four different approach for feature selection Filters, Wrapper and Embedded methods. The filters work by measuring the information of features (Information Gain) and decide the result of feature selection [5] [6]. This type of approach works fast, the result of classification is not always satisfied. Because this model not containing any error rate controlling approaches and the result is not always stable. The filter methods are easier than any other feature selection approaches, but the efficiency of this approach is not always good. Filter approaches are not guaranteed the accuracy of the learning algorithms. When dataset is high dimensional, filter method makes free of learning algorithms. Which is actually a good decision. Another approach called wrapper. In Wrapper method it produces better subset of features and it runs slowly than filter method[4]. Mark A. Hall find new filter approach for feature selection. He used correlation based heuristic method where he found feature subset [7]. It selects features by two key steps: (1) By feature searching and (2) by measure of classification error rate. Through feature searching process it selects features from original data set and use as input for next classification process to test the error rate [8]. Wrappers work too slowly because the two key steps are time consuming. Because of selecting best features, the accuracy of wrapper methods are good. But wrapper methods have large computational complexity. Moreover, to perform the wrappers applications with huge

number of features is difficult for its complex calculation. Embedded methods use incorporate feature selection approach in training process as a part. The training process is commonly specific to a specific given learning algorithm. Embedded approaches are the traditional machine learning algorithms such as artificial neural networks. Another approach called CFC(Correlation based Feature Selection), that uses correlation based heuristic to access the value of the features. The effectiveness of CFS is evaluated by comparing it with a well known wrapper feature selector that uses a specific learning algorithm to guide its search for good features [4]. The CFC is mostly similar to wrapper but it needs less computation. [9] A proposed unsupervised feature selection approach which based on KNN and clustering algorithm. In this approach, which features have minimum distance they are selected and until selection process is done for all features eliminate k-neighbor.

Chapter 3

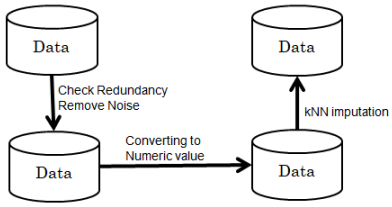
Proposed Method

Finding related features is not an easy task. It is big challenge to trace the nature of features and find out non-redundant, relevant and eliminate noisy, non-relevant features in such a huge dataset. Feature selection techniques use different evaluation measures to get better feature subset. For understanding nature of features we use different data preprocessing and clustering approaches. Through different data preprocessing approaches we remove the noisy data, convert class level to numeric value, handle missing value. Clustering method solves the dimensionality problem and gives better performance. We use Affinity Propagation Clustering approach. In this chapter we will describe the whole process in details.

3.1 Data Preprocessing

Data preprocessing is an unavoidable task for data analysis. It is a data mining technique which changed the raw dataset into an perceive format. In real world datasets are often incomplete, inconsistent, redundant, lacking in certain behaviors or trends and is likely to contain many errors and missing values. Data preprocess is needed for these issues. Now a day's real life data science activities, many categorical variables are exist in datasets. These types of categorical variables are stored in text values. Some examples include

Figure 3.1 Data Preprocessing.



game ("Football","Basketball","Cricket"), Fruit ("Mango","Apple","Pineapple") or some are like hobby ("Traveling","Reading","Gardening"). There is no answer on this problem for how to solve it in Data Science World. Data Science is not done any improve in this side of work. Fortunately, the python tools of pandas and scikitlearn provide several approaches that can be applied to transform the categorical value into suitable numeric values. We use python tools of pandas for converting the categorical values to numeric values.

Each approach has trade-offs and has potential impact on the outcome of the analysis.

3.1.1 Missing value Handling By KNN Imputation

KNN(k nearest neighbor) is an algorithm that is useful for matching a point with its closest k neighbor in a multi-dimensional space. Knn imputation works for continuous, discrete, ordinal and categorical missing data [10]. The k nearest neighbor algorithm finds missing data by finding the k closest neighbors by observing with missing data point and then imputing them based on the non missing values in the neighbors. We motivated to use knn imputation because a point value can be approximated by the values of the points that are closest to it, based on other variables. Finally we got error less and pure data sets.

3.2 Correlation Feature Grouping Method

Correlation is a statically term which is common usage refers to how close two variables are to having a liner relationship each other. Correlation based features mean correlation based heuristic to access the worth of the features [4]. Features those are highly correlated are more linearly dependent. They have nearly the same effect on the dependent variable. So we can drop one of the two features, if two features have high correlation. Correlation base feature selection has become important for big data as it finds subset of big data which needs less time and memory to find accuracy. And gives good effect on accuracy. This technique has many advantages and it avoids over-fitting [11]. Now-a-days researchers are trying to remove irrelevant and redundant features as many as possible [12] [11]. In recent years Different feature selection methods of theoretic criteria have been proposed by researchers [13][14][15]. In this thesis we use

Covariance Matrices formula to represent all the features as vectors. Than we use Affinity Propagation Clustering formula to make clusters of similar features.

3.2.1 Find Covariance Matrices Calculation for Each Sample

In data structure view covariance means how similar the variances of features are and in mathematical view the covariance matrices does the liner transform or the shearing. We have calculated the covariance matrix where each row can be thought of as a vector containing information about the relationship of a single feature with all the other features, and thus can be regarded as a feature vector for that feature. This way we have represented all the features as vectors in an n (number of features) dimensional space and clustered similar features together afterword. The covariance matrix has the following structure:

$$\begin{bmatrix} var_x & cov_{(x,y)} \\ cov_{(x,y)} & var_y \end{bmatrix}$$

where

$$var_x = \frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1} \tag{3.1}$$

$$var_y = \frac{\sum_{i=1}^{N}(y_i - \bar{y})^2}{N-1} \tag{3.2}$$

$$cov_{(x,y)} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{N-1} \tag{3.3}$$

If we see an example dataset like this:

Table 3.1: Example Dataset

X	Y
3	7
2	4

For this data,
$\bar{x} = \frac{3+2}{2} = \frac{5}{2}$
$\bar{y} = \frac{7+4}{2} = \frac{11}{2}$
$var_x = (3 - \frac{5}{2})^2 + (2 - \frac{5}{2})^2$
$var_y = (7 - \frac{11}{2})^2 + (4 - \frac{11}{2})^2$
$cov_{(x,y)} = (3 - \frac{5}{2})(7 - \frac{11}{2}) + (2 - \frac{5}{2})(4 - \frac{11}{2})$

3.2.2 Affinity Propagation Clustering

For sorting different types of data into homogeneous blocks, now a day clustering algorithms are very popular.[16]. To find correlation among the features we did Affinity Propagation clustering of the matrices. In 2007 Frey and Dueck published Affinity Propagation (AP). AP (affinity propagation) is getting day by day more popular because of its general applicability, simplicity and its great performance. AP(affinity propagation) measures the similarity between a pair of data points and use it as an input. As a potential exampler it contemplated all the data points [17]. Real value messages exchange between 2 data points. Massages are exchanges until a good quality exampler set and clusters do not arise. For solving different types of clustering problems we have used Affinity Propagation (AP). Because of AP(Affinity Propagation)'s general applicability, simplicity and better performance we got that Affinity Propagation(AP) uniformly found much more errorless clusters than other methods. And Affinity Propagation (AP) did this work in less thousand the amount of time. Clustering algorithms like K-Means and other similar algorithms we must have to choose the cluster number and have to choose the initial set of points. It is the major drawback of clustering algorithms like K-Means and other similar algorithms. [17]. AP (affinity propagation) measures the similarity between a pair of data points and uses it as an input. As a potential exampler it contemplated all the data points [17]. Real value messages exchange between 2 data points. Massages are exchanges until a good quality exemplar set and clusters do not arise. Affinity Propagation (AP) demands us to give two datasets as an input:

- If one data point is much similar to another data point that we can understand how well-suited they are to be one another's exemplar. Two points cannot stay in the same cluster if they have no similarity among them. For implementation dependency, the similarity between two data points can be set infinity or omitted.
- To be an exemplary, suitability of each data points representing preferences. We can describe it with some preferences and based on some prior information as to suitable roles for points.

Often in single matrix similarities and preferences, both are illustrated. Where ever prime diagonal values are preferences. Representing a frequent dataset through the

matrix is a good way. In matrix representation sparse are connections into points. Keeping similarity list to connected points is more practical instead of storing the nxn matrix whole in the storage. In other way manipulating matrix is the same thing as exchanging messages between points. It is done for implementation. Then the algorithms runs for a number of iterations till it repeat. There are two massage-passing steps in each iteration:

- Computing responsibilities : As cumulated evidence for how well suited for point i, point k as the exemplar to serve, Responsibility $r(i, k)$ reflects that. It takes all other possible exemplars for point i. Data point i sent responsibility to candidate exemplary point k.

- Computing availabilities : As cumulated evidence how well suite to choose point k as an exemplar for a point i, Availability $an(i, k)$ that. k should be an exemplary all other points must support that. From point k, as a candidate exemplar availability is sent to point i.

The algorithm uses the previous iteration's computed original availabilities and similarities for computing responsibilities. The similarity between as an exemplary, point k and point i and between point i and the other exemplars sum of the biggest availability and similarity point are set Responsibilities as input. If initially, the priori preference was big then the point is much suitable point as an exemplar. Sometimes a point thinks itself a good exemplar if the responsibility goes smaller. These two are competing with each other until one is selected in some iteration. Which candidate makes a good exemplar depends on computing as evidence in the responsibilities after computing availabilities. After that self-responsibility $r(k, k)$ is set for availability $a(i, k)$. It also added exemplar k's received positive responsibilities. At last, we got a terminating condition to terminate the process. The terminating condition is reaching the max number of iterations or when some threshold value increase then changes values. While Affinity Propagation (AP) is performing, at any point or any time if we add Availability (a) and Responsibility (r) matrices we get all clustering information. Point i's exemplar representation, for point i we have to add the k and the max $r(i, k) + a(i, k)$. By scanning the major diagonal we can get the set of exemplars. If an exemplar is point i, $r(i, i) + a(i, i) > 0$[17].

3.2.3 Random Features from Cluster Group

We have selected 30% features randomly from each of these clusters as representatives of all the features in that cluster and considered them for model training purpose with an attempt to mitigate the disaster of dimensionality while retaining as much information as possible.

3.3 Co-relation base feature selection with Clustering

3.3.1 Decision Tree ID3(Iterative Dichotomiser 3) Algorithm

The ID3 algorithm is Information Entropy-based a classification algorithm. In decision tree learning, Ross Quinlan invented ID3 (Iterative Dichotomiser 3) algorithm from data set to generate a decision tree. This algorithm is invented from the information to develop choice tress. Generally, in natural language processing domains and machine learning use ID3 algorithm. For model the classification technique, the decision tree yoke construct a tree. The most important thing is that ID3 calculation is its capacity to separate an unpredictable choice tree into a gathering of more straight forward choice tree [18]. Decision tree iteratively separated the data into little subsets. This process goes on until a decision tree building stopping criteria is met or all the subsets are related to a single class.

- For selecting the best feature A_j, information theory is used in ID3 algorithm. In a tree a root node must have maximum $InformationGain$. The A_j has max $InformationGain$.

- Shown in 3.4, $x_i \epsilon D$ needs the infromation in average amount for identifying a claas c_l, to classify an instance. From $|c_l, D|/|D|$, It estimated that the class c_l has a subset as x_i where probability is p_i .

$$Info(D) = -\sum_{i=1}^{N} p_i log_2(p_i) \tag{3.4}$$

- For correctly classifying $x_i \epsilon D$ among the partitioning by A_j, the expected information required is $Info_A(D)$. In $Info_A(D)$ calculation jth partition as the weight of $\frac{D_j}{D}$ acts like. Which shows in 3.5.

$$Info_A(D) = \sum_{j=1}^{n} \frac{D_j}{D} \times Info(D_j) \tag{3.5}$$

- Differece between $Info_A(D)$ and $Info(D)$ is defined as $InformationGain$ which shown in 3.6.

$$Gain(A) = Info(D) - Info_A(D) \tag{3.6}$$

Algorithm 1 ID3 Algorithm

```
ID3(Examples, TargetAttribute, Attributes)
For the tree have to create a root node
if All examples == Positive or(+)ve then
    return the tree Root, with label = +.
else if All examples == Nagetive or (-)ve then
    return the tree Root, with label = -.
else if Predicting attributes no. == empty then
    return the tree Root, with label = most common target attribute From the
    examples
else
    Best classifiers examples are T ← attribute
    Root of the Decision Tree attribute = T.
    for all M_i of T, each possible value do
        Under the root add a new branch, according to M_i = T
        Let examples (M_i) be the subset of examples that have the value M_i for T
        if Examples(M_i)== empty then
            Then below the new branch add a leaf node with label = most target value
            in the examples
        else
            below the new branch add the subtree ID3(Examples(M_i), TargetAttribute,
            Attribute- {T})
        end if
    end for
end if
return Root
```

3.3.2 Decision tree CART(Classification and Regression Trees)

In 1984 Breiman et al discover the idea of CART algorithm which based on regression and classification trees. A CART tree is one kind of binary tree. By splitting a node repeatedly into two different child node the CART tree is created. This process is beginning with the root node. The whole learning sample contains the root node. To make the child nodes 'purest' tree growing is a good process. In tree growing process choose a split among all the possible splits at each node. Splits of univariate are only considered in the CART algorithm. It means, one predictor variable makes only one split. All possible splits consist of possible splits of each predictor. If I categories, A nominal categorical variable X, for this predictor possible splits are $2^{I1}1$ in number. On X there are $K1$ different splits if, with K different values, X is an ordinal continuous variable or categorical variable. On each node from the root node, the tree starts growing by repeatedly by the following steps.

3.3.3 Random Forest

Random Forest is a multipurpose machine learning algorithm and skilled of performing both regression and classification responsibilities [19]. Random forest algorithm developed by Leo Breiman. This algorithm uses an ensemble of classification trees. By using data's bootstrap sample each classification trees are built. In each split, the set of candidate variables are a subset of random variables. For a successful approach, RF uses bagging algorithm. And for tree building, it chooses the random variable selection. Obtain low bias trees as each and every trees are grown fully or unpruned. Random Forest machine learning algorithm is generally immune to data noise and over fitting and for remote sensing data, it is useful that's the main reason for choosing this algorithm to our work. And also this algorithm can handle high dimension data and typically achieve higher accuracy than single layer neural networks and decision tree algorithm [20][21]. Again it is also good for estimating missing data though we have used KNN-imputation for handling missing data here. For balancing error in imbalanced data it has a method called weighted random forest (WRF). Some important features of Random Forest are:

1. It has an effective method for estimating missing data.

2. It has a method, weighted random forest (WRF), for balancing error in imbalanced data.

3. It estimates the importance of variables used in the classification.

Algorithm 2 Random Forest Algorithm

function SegmentPixel(pixel)

result equals to zero or null
for $i\epsilon$ F **do**
  node equals to $F[i]$
  while $l[node]$ not equal -1 **do**
    if $pixel[J[node]]$ getter than equal to $S[node]$ **then**
      node equals to addition of $(L[node], F[i])$
    else
      node equals to addition of $(R[node], F[i])$
    end if
  end while
  result equals to addition of $(result, C[node])$
end for
return result
end function

Accuracy : The random forest (RF) algorithm provides the percentage of missclassification and internal measure of error. From this, the global accuracy of the model can be calculated. Based on the confusion matrix performance of each class can be evaluated.

3.3.4 AdaBoost Algorithm

AdaBoost is the short form of Adaptive Boosting. It is a machine learning model designed by Yoav Freund and Robert Schapire. The main reason for adaptive is that it makes weaker those instances which were misclassified by previous classifiers. In general, this model in combination with learning algorithms to increase their performance[19].For feature extraction which algorithm is also be used [22].It is an ensemble classifier that creates strong classifier by combining weak classifiers. It is sensitive to noisy data and

outliers. The Adaptive boosting algorithm is much efficient ensemble learning algorithm. It iteratively generates a set of diverse week learners. And finally, combine their outputs using the weighted majority voting rules like the final decision. This algorithm calls a sample learning algorithm in each iteration which is named as a base learner and creates the classifier. The coefficient is appointed to the classifier. By weighted voting, the final classification result is obtained which is related to the weight coefficient of weak classifiers. The algorithm increases the weight in the last voting if the weak learner error is low [23].The real Adaboost algorithm gives minor error rates than the diverse adaptive boosting algorithm.

Algorithm 3 AdaBoost Algorithm

Input: T is the training data, n is the no. of iterations and a learning scheme.

Output: Ensemble model, $M*$

Method:

1: initialize weight, $x_i \epsilon T$ to $\frac{1}{t}$
2: **for** i equals 1 **to** n **do**
3: sample T with replacement according to instance weight to obtain T_i;
4: use T_i, and learning scheme to derive a model, M_i;
5: compute error (M_i)
6: **if** $error(M_i) \geq 0.5$ **then**
7: go back step 3 and retry;
8: **end if**
9: **for** each correctly classified $x_i \epsilon D$ **do**
10: weight of x_i multiply by $\frac{error(M_i)}{1-error(M_i)}$;
11: **end for**
12: instances weight normalization;
13: **end for**

To use $M*$ to classify a new instance, x_new.

1: initialize weight of each class to zero;
2: **for** $i = 1$ **to** n **do**
3: $w_i = log\frac{1-erreor(M_i)}{error(M_i)}$; {w}eight of the classifier's vote
4: $c = M_i(x_new)$; {c}lass prediction by M_i
5: add w_i to weight for class c;
6: **end for**
7: **return** class with largest weight;

3.3.5 Logistic Regression

After Linear Regression algorithm, Logistic regression has become popular machine learning algorithm. Though both algorithms are similar in most of the cases. Generally, Logistic regression is popular for classification tasks whereas the other is used for prediction or forecasting. For example, finding an email is spam or not spam a tumor is benign or malignant these type of classification tasks Logistic regression mainly people use. Again, it is useful for rudiment and useful thing for classification tasks. A linear equation with independent forecasting to predict a value this algorithm is used. Here the value can be from negative to positive infinity. Here We needed true or false output.

3.3.6 Bagging Algorithm

Many real life datasets include imbalance data as learning classifiers. Imbalance dataset is where one of the classes includes much smaller number of examples than the other majority classes [24]. Bagging name came from Bootstrap aggregating presented by Breiman in 1996. It works by generating different bootstrap samples by voting classifiers. It injects some random perturbation into parallel training sets as adopting bootstrap sampling. It works by sampling with replacement from the main training dataset. Generally, the size of sample set and original set is equal. If the training set has N rows the probability of a row to be selected at least one time is $1-(1-1/N)N$. If the size of N is large it will be about $1-1/e$. About 63.2% on average Each bootstrap sample will contain unique examples from the training set [24]. Here the number of component classifiers took as K and training data D of size N. for generating component Classifiers Ci, same learning algorithm is applied. The final ensemble $C*$ created by aggregating. Kuncheva (2014) is a book to learn more about bagging why it works better [25].

Algorithm 4 Bagging Algorithm

Input: T is the training data, n is no. of iterations and a learning scheme.

Output: Ensemble model, $M*$

Method:

1: **for** i = 1 **to** k **do**
2: By sampling S with replacement have to create bootstrap sample S_i;
3: For driving a model, M_i must have to use S_i and the learning scheme;
4: **end for**

Using $M*$ for classifying a new instance,x_new:

Each M_i $M*$ classify x_new and return the majority vote

3.3.7 SVM(Support Vector Machine) algorithm

From Machine Learning Perspective and embedded system Support vector machine(SVM) has been chosen as it represents a framework. Support vector machine is a linear or can be non-linear classifier with mathematical operation that can differentiate two dissimilar objects. Here we use the following pseudo code:

Algorithm 5 Training an Support Vector Matchine

Require: X and y loded with training labeled data, $\alpha \Leftarrow 0$ or $\alpha \Leftarrow$ *partially trained SVM*

1: $C \Leftarrow$some value(10 for example)
2: **repeat**
3: **for all** $\{x_i, y_i\}, \{x_j, y_j\}$ **do**
4: Optimize a_i and a_j
5: **end for**
6: **until** no changes in α or other resource constraint criteria met.

Ensure: Retain only support vectors ($\alpha_i > 0$)

Chapter 4

Data sets And Comparison of Performance

4.1 Datasets

In our work we use four big data sets which have features number greater than 100 and instances number greater than 1000. We use HAPT (Smartphone-Based Recognition of Human Activities and Postural Transitions) data set, Gas sensors for home activity monitoring Data Set, Weight Lifting Exercises monitored with Inertial Measurement Units Data Set and SECOM Data Set. The characteristic of the data sets are Multivariate, attribute characteristic is real. The complete description of the data sets giver below subsections.

4.1.1 HAPT (Smartphone-Based Recognition of Human Activities and Postural Transitions) Data Set

The recording of thirty (30) experimental subjects acting normal works and postural alternations, when having a loin-placed Smartphone with embedded inertial sensors, from that the activity recognition dataset created. These experiments were performed upon a team of 30 volunteers between 19-48 years of age bracket. The volunteers performed one draft of activities composed of 6 normal activities: three (3) fixed pose (lying, sitting, standing) and three (3) moving activities (walking, walking upstairs and walking downstairs). Between the fixed poses, pose transitions also occurred in these experiments. The changing poses are: sit to stand, sit to lie, stand to sit, stand to lie,

lie to stand and lie to sit.

Table 4.1: HAPT (Smartphone-Based Recognition of Human Activities and Postural Transitions) Data set

Data Set Characteristics:	Multivariate, Time-Series	Number of Instances:	10929	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	561	Date Donated:	29-07-2015
Associated Tasks:	Classification	Missing Values?	N/A	Number of Web Hits:	105067

4.1.2 Gas sensors for home activity monitoring Data Set

There are 100 of recordings a sensor array under different In house decoration, there are some conditions: wine, curtains, tables, wine, wallpapers, and fruits decorations. Humidity, temperature sensors, and eight (8) mox gas sensors are included in the array. In house decoration, there are some conditions: wine, curtains, tables, wine, wallpapers, and fruits decorations. Humidity, temperature sensors, and eight (8) mox gas sensors are included in the array. While two (2) different subjects' stimuli: fruits decoration and wallpaper in the background of house activity those sensors were illuminated. By positioning the stimulus sensors to close the reaction of wallpaper and fruits decoration were stored. The average stimulation duration is 42 minutes. For five (5) different conditions the dataset has different time series for: wine, curtains, tables, wine, wallpapers, and fruits decoration activity. There are 42 inductions with curtains, 38 inductions with wine and many more.

Table 4.2: Gas sensor for home activity monitoring Data set

Data Set Characteristics:	Multivariate	Number of Instances:	13910	Area:	Computer
Attribute Characteristics:	Real	Number of Attributes:	129	Date Donated:	25-04-2012
Associated Tasks:	Classification	Missing Values?	N/A	Number of Web Hits:	120830

4.1.3 Weight Lifting Exercises monitored with Inertial Measurement Units Data Set

Five different kinds of weight lifting exercise for biceps curl were asked to perform among six 19-24 years old health subjects. All the exercises were performed by one of the professionals.

Table 4.3: Weight Lifting Exercises monitored with Inertial measurement Units Data set

Data Set Characteristics:	Multivariate	Number of Instances:	39242	Area:	Physical
Attribute Characteristics:	Real	Number of Attributes:	152	Date Donated:	24-11-2013
Associated Tasks:	Classification	Missing Values?	Yes	Number of Web Hits:	34198

4.1.4 SECOM(Semi-Conductor Manufacturing Process) Data Set

This dataset is collected from a semiconductor manufacturing system. Basically a modern complex semiconductor manufacturing system is under consistent observation. It observed via monitoring the variables/signals which are collected from the system measurement points or the sensors.

Table 4.4: SECOM (Semi-Conductor Manufacturing Process) Data Set

Data Set Characteristics:	Multivariate	Number of Instances:	1567	Area:	Computer
Attribute Characteristics:	Real	Number of Attributes:	591	Date Donated:	19-11-2008
Associated Tasks:	Classification, Causal-Discovery	Missing Values?	Yes	Number of Web Hits:	80090

In a marked observation system, all the signals are not equally important. A combo of some useful information (noise as well as redundant information) is contained by the measured signals. The important pieces of information are mostly driven into the latter

two numbers. Most of the engineers have a huge number of signals from the requirement. For identifying the most important signals we can apply feature selection. In feature selection technical we can assume every type of signals as a feature. For determining the key factoring contribution to downstream in the system, the system engineer might use these signals.

4.2 Experimental Analysis

We took 30% random features from each group. And append all the features and use some machine learning algorithms on the small data set and also use those algorithms on previous full data sets. Some of data sets for different algorithms We get higher accuracy on our correlation base data then the full data set accuracy for some machine learning algorithms. We compare the main data set's accuracy with the correlation accuracy after performing clustering. We also plot graph for different algorithms for visualize the comparison. If we see table 4.1 we can see the comparison between accuracy main data set and after applying correlation feature selection process where we apply Decision tree (ID3). In table 4.1 we applied ID3 first time in main datasets and second time when after processed correlation based feature selection. For HAPT dataset with 561 attributes we get 91.63% where after applying correlation based feature selection with 176 attributes we get 92.34% accuracy which is similar to main accuracy. In Gas sensor

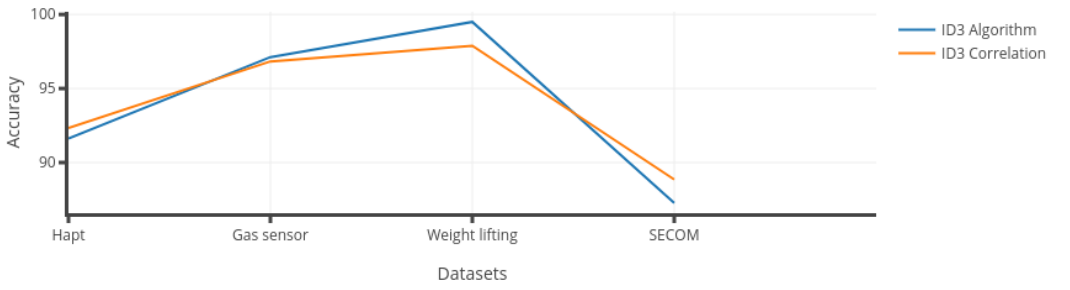
Table 4.5: Comparing accuracy of selected datasets for Decision Tree ID3(Iterative Dichotomiser 3)

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	91.63%	176	92.34%
Gas sensor	129	97.12%	41	96.83%
Weight lifting	158	99.50%	50	97.89%
SECOM	590	87.26%	173	88.85%

dataset for 129 attributes we get 97.12% accuracy. After applying correlation based feature selection for 41 attributes we get 96.83% accuracy. Number of attributes almost one third time less than main dataset but accuracy almost similar. In Weight lifting

dataset for 158 attributes we get 99.50% accuracy. After applying correlation based feature selection we get 41 attributes. Than we apply ID3 machine learning algorithm and we get 97.89% accuracy. Number of attributes almost one third time less than main dataset and the accuracy are almost similar. As other datasets SECOM dataset has multidimensional data. It has 590 attributes which is huge in size. After applying correlation based feature selection we get 173 attributes which is very much less than the original dataset. We apply Decision Tree algorithm in both cases. We get 87.26% accuracy in the original dataset and 88.85% accuracy after applying correlation based feature selection. These results are quite impressive. If we see its graph in figure **??** x-axis shows the accuracy and y-axis shows the datasets. The blue line is for original dataset and the orange line is for the correlation based feature selection. We can see the blue and the orange line is almost near to each other.

Figure 4.1 Comparing accuracy for Decision Tree ID3 (Iterative Dichotomiser 3) Algorithm by Graph.



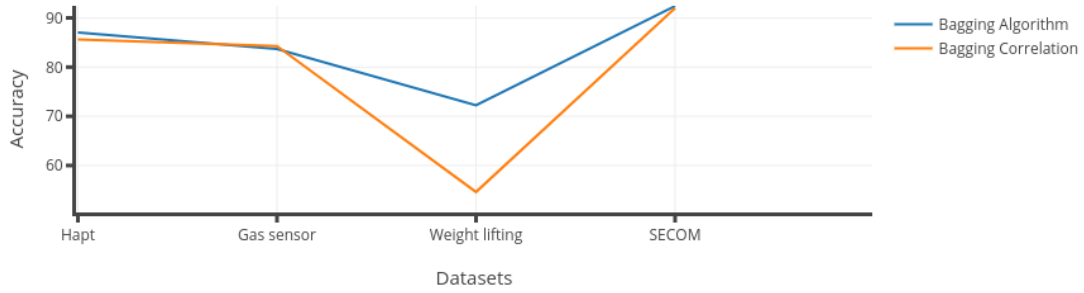
It means for all the datasets the accuracy is almost similar. We applied various machine learning algorithm in our four datasets before applying correlation feature selection and after correlation feature selection. Some time we get bad output but we get good result than bad. Most of the time our accuracy is almost similar to the original datasets process. For bagging algorithm we can see that the actual accuracy of HAPT dataset is 87.02% and for correlation-based feature selection it is 85.65% or for weight lifting dataset the actual accuracy is 72.48% and accuracy of correlation based feature selection is 54.62% which is much lower than the actual one. In the graph, we can see that the difference between the orange and blue line is much bigger for weight lifting dataset. As we said that our correlation-based feature selection process does not

Table 4.6: Veracity and comparing of selected datasets for Bagging Algorithm

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	87.02%	176	85.65%
Gas sensor	129	83.69%	41	84.28%
Weight lifting	158	72.24%	50	54.62%
SECOM	590	92.41%	173	92.02%

give efficient result for any specific dataset or any specific algorithms. This technique sometimes works perfectly.

Figure 4.2 Veracity and comparing of selected datasets for Bagging Algorithm.



The effect of random forest algorithm is like as bagging algorithm. Both are facing problem in weight lifting dataset. As like bagging algorithm, random forest algorithm also shows much lower accuracy than the actual accuracy. The correlation-based feature selection process shows 67.11% accuracy where the actual accuracy of weight lifting dataset for the random forest algorithm is 74.61%.

Table 4.7: Accuracy of selected datasets for Random Forest

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	83.96%	176	83.49%
Gas sensor	129	85.41%	41	84.85%
Weight lifting	158	74.61%	50	67.11%
SECOM	590	93.37%	173	93.30%

Figure 4.3 Accuracy of selected datasets for Random Forest Algorithm and compare .

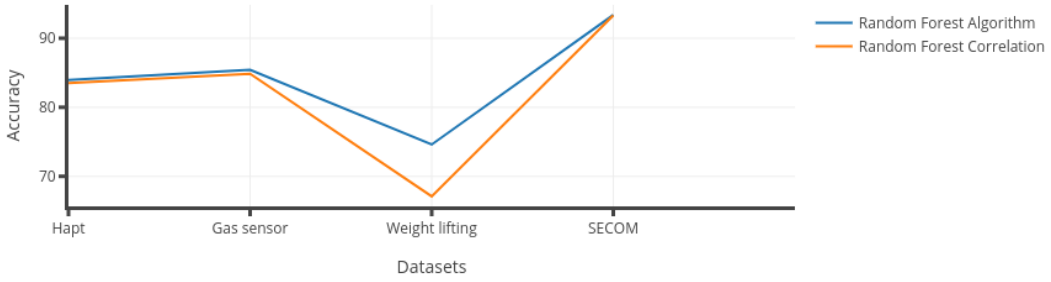


Table 4.8: Efficiency of selected datasets for AdaBoost Algorithm

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	46.74%	176	39.86%
Gas sensor	129	55.83%	41	58.76%
Weight lifting	158	69.52%	50	30.59%
SECOM	590	91.71%	173	92.92%

Figure 4.4 Efficiency of selected datasets for AdaBoost Algorithm by graph.

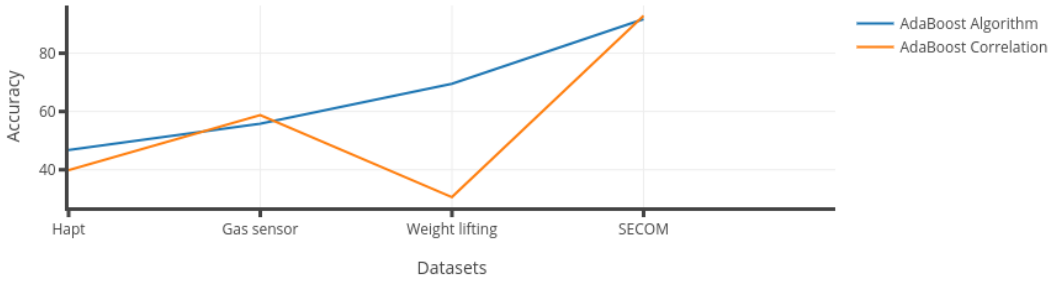


Table 4.9: Comparing efficiency of selected datasets for Logistic Regression

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	93.45%	176	91.34%
Gas sensor	129	89.04%	41	87.52%
Weight lifting	158	00.77%	50	00.77%
SECOM	590	90.75%	173	92.22%

Figure 4.5 Comparing efficiency for Logistic Regression.

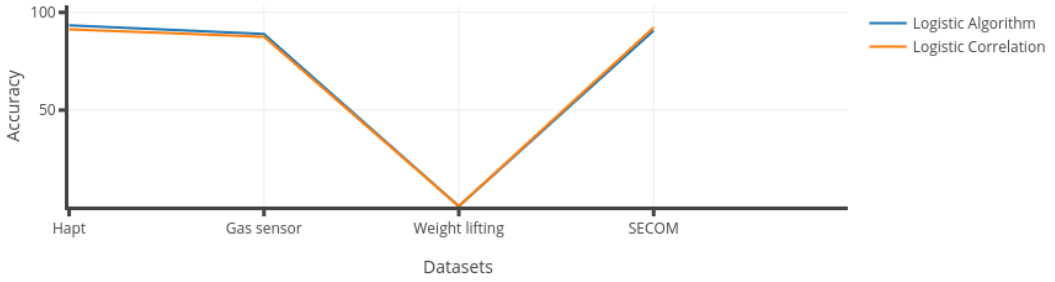


Table 4.10: Veracity and compare of selected datasets accuracy for Decision Tree CART (Classification and Regression Trees)

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	83.22%	176	81.97%
Gas sensor	129	77.99%	41	80.94%
Weight lifting	158	72.23%	50	59.77%
SECOM	590	86.09%	173	87.11%

Figure 4.6 Veracity and compare of selected datasets accuracy for Decision Tree CART (Classification and Regression Trees) Algorithm.

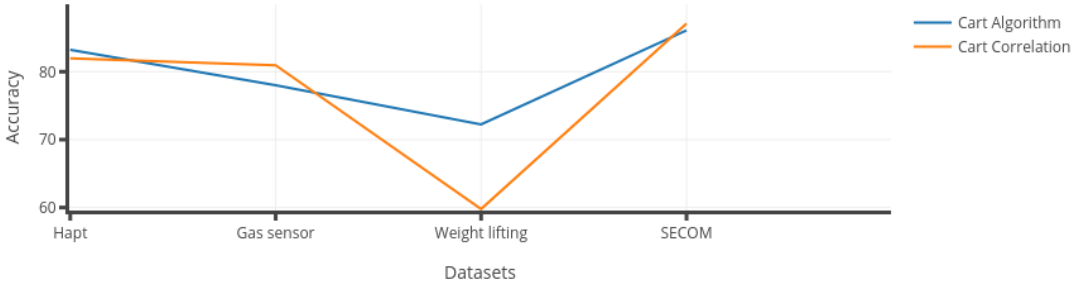
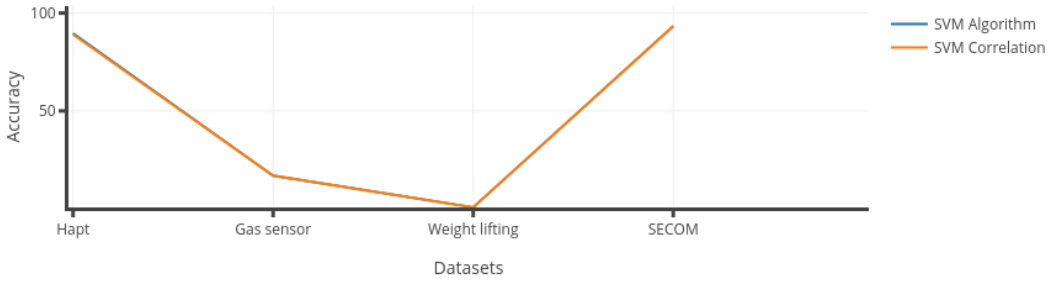


Table 4.11: Comparing accuracy of selected datasets for SVM(Support Vector Machine) algorithm

Name	Main datasets		Correlation base	
	Attribute	Accuracy	Attribute	Accuracy
Hapt	561	89.70%	176	89.02%
Gas sensor	129	16.97%	41	16.98%
Weight lifting	158	00.77%	50	00.77%
SECOM	590	93.37%	173	93.36%

Figure 4.7 Comparing accuracy for SVM (Support Vector Machine) Algorithm.



Chapter 5

Conclusions

5.1 Conclusions

Number of multi-Class dataset has been increasing day by day. To analysis these datasets has become difficult for scientist. By our method we can reduce the class label of multi-class dataset. Here a number of Machine learning methods are used to compare accuracy with four datasets. It is well known that there is no single method to get good result for every machine learning algorithms. Therefore, further efforts are need to improve this method for better performance. In this thesis we have reviewed more than 50 research papers related this topic in the period from 2008 to 2019. We have noticed that some machine learning algorithms shows worse performance for some datasets in this case we will ignore that datasets for that algorithm. Our experimental analysis indicates that our method gives better performance for most of the cases even though we have decreased the number of attribute 2 to 3 times.

5.2 Future Work

In this thesis we applied our correlation based selection method in many machine learning algorithms. In future we will compare our result correlation based selection method to other existing correlation feature selection methods. And we will try to update our working process so that not only high dimensional dataset but also lower dimension dataset can give better performance.

Bibliography

[1] H. Liu, H. Motoda, and L. Yu, "Feature selection with selective sampling," in *International Conference on Machine Learning*, 2002, pp. 395–402. 1

[2] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," in *International Conference on Machine Learning*, vol. 1, 2001, pp. 74–81. 1

[3] E. P. Xing, M. I. Jordan, R. M. Karp *et al.*, "Feature selection for high-dimensional genomic microarray data," in *International Conference on Machine Learning*, vol. 1. Citeseer, 2001, pp. 601–608. 1

[4] M. A. Hall and L. A. Smith, "Feature selection for machine learning: comparing a correlation-based filter approach to the wrapper." in *Florida Artificial Intelligence Research Society Conference conference*, vol. 1999, 1999, pp. 235–239. 4, 5, 7

[5] A. Al-Ani, "A dependency-based search strategy for feature selection," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12 392–12 398, 2009. 4

[6] B. Bonev, F. Escolano, and M. A. Cazorla, "A novel information theory method for filter feature selection," in *Mexican International Conference on Artificial Intelligence*. Springer, 2007, pp. 431–440. 4

[7] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in *Machine Learning Proceedings 1994*. Elsevier, 1994, pp. 121–129. 4

[8] H.-H. Hsu, C.-W. Hsieh *et al.*, "Feature selection via correlation coefficient clustering." *JSW*, vol. 5, no. 12, pp. 1371–1377, 2010. 4

[9] P. Mitra, C. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 3, pp. 301–312, 2002. 5

[10] L. Beretta and A. Santaniello, "Nearest neighbor imputation algorithms: a critical evaluation," *BMC medical informatics and decision making*, vol. 16, no. 3, p. 74, 2016. 7

[11] Z. Zeng, H. Zhang, R. Zhang, and C. Yin, "A novel feature selection method considering feature interaction," *Pattern Recognition*, vol. 48, no. 8, pp. 2656–2666, 2015. 7

[12] P. Maji and S. Paul, "Rough set based maximum relevance-maximum significance criterion and gene selection from microarray data," *International Journal of Approximate Reasoning*, vol. 52, no. 3, pp. 408–426, 2011. 7

[13] F. Jiang, Y. Sui, and L. Zhou, "A relative decision entropy-based feature selection approach," *Pattern Recognition*, vol. 48, no. 7, pp. 2151–2163, 2015. 7

[14] M. Sebban and R. Nock, "A hybrid filter/wrapper approach of feature selection using information theory," *Pattern Recognition*, vol. 35, no. 4, pp. 835–846, 2002. 7

[15] H. Yan, X. Yuan, S. Yan, and J. Yang, "Correntropy based feature selection using binary projection," *Pattern Recognition*, vol. 44, no. 12, pp. 2834–2842, 2011. 7

[16] J. A. Hartigan, "Direct clustering of a data matrix," *Journal of the american statistical association*, vol. 67, no. 337, pp. 123–129, 1972. 9

[17] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *science*, vol. 315, no. 5814, pp. 972–976, 2007. 9, 10

[18] P. Vasudevan, "Iterative dichotomiser-3 algorithm in data mining applied to diabetes database," *Journal of Computer Science*, vol. 10, no. 7, p. 1151, 2014. 11

[19] Y. K. Jakhar, N. Mishra, and R. Poonia, "Predication accuracy analysis of data mining algorithms on meteorological data using r programming," 2018. 13, 14

[20] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 24–31, 2016. 13

[21] J. J. Lawrence, Z. M. Grinspan, J. M. Statland, and C. J. McBain, "Muscarinic receptor activation tunes mouse stratum oriens interneurones to amplify spike reliability," *The Journal of physiology*, vol. 571, no. 3, pp. 555–562, 2006. 13

[22] S. M. Basha, D. S. Rajput, and V. Vandhan, "Impact of gradient ascent and boosting algorithm in classification," *International Journal of Intelligent Engineering and Systems (IJIES)*, vol. 11, no. 1, pp. 41–49, 2018. 14

[23] R. Ceylan and M. Barstugan, "Feature selection using ffs and pca in biomedical data classification with adaboost-svm," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 6, no. 1, pp. 33–39, 2018. 15

[24] M. Lango and J. Stefanowski, "Multi-class and feature selection extensions of roughly balanced bagging for imbalanced data," *Journal of Intelligent Information Systems*, vol. 50, no. 1, pp. 97–127, 2018. 16

[25] I. Abdallah, V. Dertimanis, H. Mylonas, K. Tatsis, E. Chatzi, N. Dervilis, K. Worden, and E. Maguire, "Fault diagnosis of wind turbine structures using decision tree learning algorithms with big data," in *Safety and Reliability–Safe Societies in a Changing World*. CRC Press, 2018, pp. 3053–3061. 16