# main steps for doing data mining project using weka

**Technical Report** · February 2016

CITATION
1

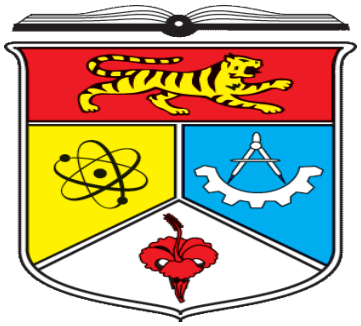
READS
12,498

**1 author:**

Dalia Sami Jasim
Universiti Kebangsaan Malaysia
**9** PUBLICATIONS **32** CITATIONS

SEE PROFILE

# Data Mining Approach and Its Application to Dresses Sales Recommendation



By

Dalia Sami Jasim

Abstract:

The fundament of data mining (DM) is to analyses data from various points of view. Classify the data and summarize it, DM has begun to be widespread in every and each application. Although we have huge magnitude of data, but we do not have helpful information in each field, there are a lot of DM software and tools to aid us the advantageous information. In this work we give the essentials of DM steps such as preprocessing data (remove outlier, replace missing values etc.), attribute selection, aim to choose just relevant attribute and removing the irrelevant attribute and redundant attribute, classification and assessment of varied classifier models using WEKA tool. The WEKA software is helpful for a lot of application's type, and it can be used in different applications. This tool is consisting of a lot of algorithms for attribute selection, classification regression and clustering.

**Keywords**
Data mining, Weka, preprocessing, classification.

## 1. Introduction :

Data mining (DM) or knowledge discovery is the procedure of using statistical techniques and knowledge-based methods to analyze the data to mine patterns having meaning from vast data sets and change these into helpful information. Through DM process, diverse techniques are utilized to detection relationships, patterns or associations among the dataset features, that can be transform into knowledge around past patterns and trends of future. In general whatever four types of relationships are discussing: classes, clusters, sequential and association's patterns. Classification category deals with DM techniques that looking for cluster and class relationships [1].

Classification point out the DM side that making an attempt to predict to which class every observation of the dataset must be placed by constructing a model based on some predictor attributes. Classification methods divided into: unsupervised or supervised. In supervised classification, one attribute of the dataset include predetermined values that represent a collection of the data. These collections are called classes. For unsupervised classification the objective is to partition into groups or clusters the observations of the dataset based on some logical relationship that exists among the values of the attributes but that must yet be discovered [2].

And to achieve classification we firstly need to preprocess the data that we already takes from UCI web page and to do that we will use Weka to achieve all data mining process. Weka tool is software for data mining existing below the General public license (GNU). The Weka system is developed at Waikato University in New Zealand. Weka is available for free at http://www.cs.waikato.ac.nz/ml/weka. The Developers used Java to write this system. Weka provides implementation of data mining and machine learning algorithms. User can achieve classification, association, clustering, filtering, regression, visualization etc. by using Weka tool. The focus of this report is to use an existing dataset (Dresses_Attribute_Sales Data Set for year 2014) from UCI Machine Learning Repository to preprocessing data for data mining process. This dataset contain Attributes of dresses and their recommendations according to their sales. Sales are monitor on the basis of alternate days. There are many preprocessing data techniques like (clean the data, reduce the size of data and transform data into appropriate type).then we can use this data set for a Recommender System. The objective of a Recommender System is to build up meaningful recommendations for users about items or products that might interest them.

## 2. **Related literature on using the dataset**:

Until now no one use dresses sales dataset from UCI, but there are several approaches have been proposed in the field of recommendation system using data mining approach such as [3] that proposed medical advices recommendation system based on a hybrid method using varied classifications and unified Collaborative Filtering. Multiple classifications based on decision tree algorithms are applied to build an accurate predictive model that predicts the disease risk diagnosis for the monitored cases. [4] Designed a novel book recommendation system. Readers will be redirected to the recommendation pages when they can't find the required book through the library bibliographic retrieval system. The recommendation pages contain all the essential and expanding book information for readers to refer to. Readers can recommend a book on these pages, and the recommendation data analyzed by the recommendation system to make scientific

purchasing decision, the author proposed two formulas to compute the book value and copy number respectively based on the recommendation data. In same trend [5] Presented a recommendation technique based on opinion mining to propose top ranked books on different discipline of the computer science. Based on the need of the customers and the reviews collected from them, they have categorized features for the books and analyze the features on the basis of several characteristics that they had categorized and reviews of the users. Assigned Weights to categorized features according to their importance and usage, and accordingly the ranks are given. Finally, top ten ranked books are listed.
[6] Proposed a movie recommendation system that has the ability to recommend movies to a new user as well as the others. It mines movie databases to collect all the important information, such as, popularity and attractiveness, required for recommendation. It generates movie swarms not only convenient for movie producer to plan a new movie but also useful for movie recommendation. [7] Introduced a different approach to recommender system which learn rules for user preferences using classification based on Decision Lists. they had followed two Decision List based classification algorithms like Repeated Incremental Pruning to Produce Error Reduction and Predictive Rule Mining, for learning rules for users past behaviors. The authors also list out their proposed recommendation algorithm and discuss the advantages as well as disadvantages of our approach to recommender system with the traditional approaches. They had validated their recommender system with the movie lens data set that contains hundred thousand movie ratings from different users, which is the bench mark dataset for recommender system testing.

## 2.1 PRELIMINARY ON DATA DESCRIPTION

## 2.2 DATA DESCRIPTION

The dataset represents dresses sales and the recommendations according to this sales. It is include 13 features and class representing Price, Size, Season, NeckLine , waiseline, Material, FabricType, Rating, Decoration, SleeveLength ,Style, Pattern, Type, Recommendation .

**2.3 Source of Data**: This dataset produced by Muhammad Usman & Adeel Ahmed, and Students at Air University.

**2.4 Description on Domain problems**: The domain problem we can use this data set with it is recommendation system that try to recommend products to customers depend on ratings or the behavior of past customer.

### 2.5 List of attributes with their types:

| ID | Attribute | Type |
|---|---|---|
| 1- | Dress_ID | numeric |
| 2- | Style | categorical |
| 3- | Price | categorical |
| 4- | Rating | numeric |
| 5- | Size | categorical |
| 6- | Season | categorical |
| 7- | NeckLine | categorical |
| 8- | SleeveLength | categorical |
| 9- | waiseline | categorical |
| 10- | Material | categorical |
| 11- | FabricType | categorical |
| 12- | Decoration | categorical |
| 13- | PatternType | categorical |
| 14- | Recommendation | numeric |

Tabel1. List of attributes

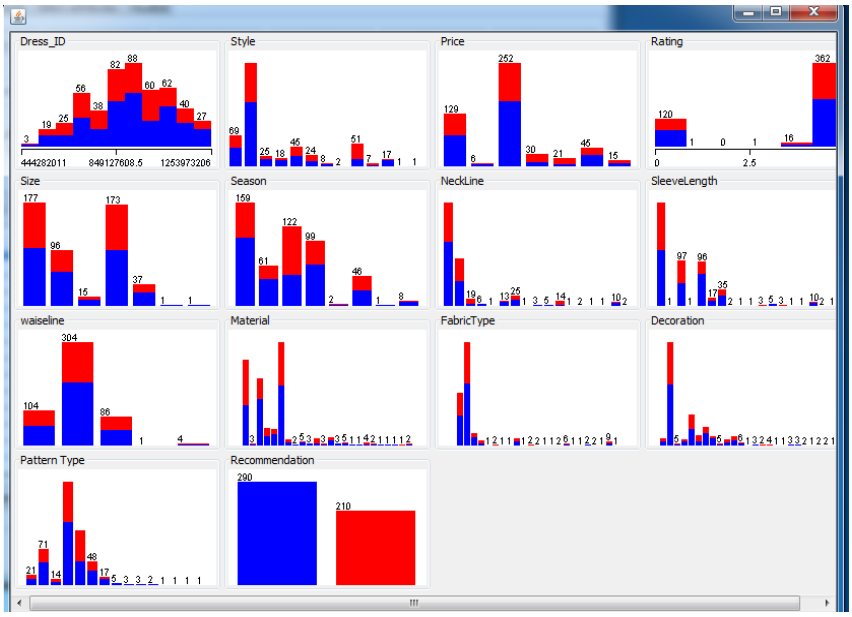



Figure 1.Attribute visualization

| | | | | | |
|---|---|---|---|---|---|
| Data Set Characteristics: | Text | Number of Instances: | 501 | Area: | Computer |
| Attribute Characteristics: | N/A | Number of Attributes: | 13 | Date Donated | 2014-02-19 |
| Associated Tasks: | Classification, Clustering | Missing Values? | Yes | Number of Web Hits: | 14457 |

Figure 2. DataSet description

### 3. Preliminary examination of data:

3.1 Are there any “obvious” patterns in the data which might be helpful?

No, I can’t see any obvious patterns in the dataset.

3.2 Are there any experts that understand the data well and with whom you can talk?

No, I don’t know any experts that know the data and I can speak with him.

3.3 Are values missing for some attributes?

Yes, there are a missing values in attributes (price , season, neckline , waiseline , material , FabricType ,decoration , PatternType) .

| No. | Dress_ID Numeric | Style Nominal | Price Nominal | Rating Numeric | Size Nominal | Season Nominal | NeckLine Nominal | SleeveLength Nominal | waiseline Nominal | Material Nominal | FabricType Nominal | Decoration Nominal | Pattern Ty Nominal |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 258 | 1.10727989 4E9 | party | very-high | 0.0 | M | Spring | v-neck | threequater | natural | lycra | sattin | applique | null |
| 259 | 1.163040038E9 | party | high | 0.0 | M | Winter | Sweetheart | sleevless | natural | null | null | pearls | null |
| 260 | 1.09187743E9 | party | high | 0.0 | M | Spring | Sweetheart | sleevless | null | nylon | null | null | null |
| 261 | 1.053467336E9 | party | Average | 5.0 | M | Winter | o-neck | capsleeves | empire | polyster | null | lace | null |
| 262 | 6.09889168E8 | party | high | 4.7 | M | Winter | sweetheart | sleeveless | natural | null | chiffon | flowers | null |
| 263 | 6.6272774E8 | party | | 4.8 | free | Winter | o-neck | sleevless | empire | null | null | embroidary | null |
| 264 | 1.090993173E9 | party | | 4.5 | L | Summer | | full | | | | | |
| 265 | 9.63684109E8 | party | very-high | 4.9 | M | Spring | Sweetheart | sleeevless | empire | null | chiffon | beading | null |
| 266 | 6.19817668E8 | party | high | 4.6 | M | winter | Sweetheart | sleevless | natural | null | chiffon | null | null |
| 267 | 1.066740059E9 | party | Average | 5.0 | M | Spring | Sweetheart | sleevless | natural | polyster | null | null | null |
| 268 | 7.3218448E8 | party | very-high | 4.8 | L | Spring | Sweetheart | sleevless | empire | microfi... | organza | beading | null |
| 269 | 1.044683061E9 | party | very-high | 0.0 | M | Winter | Sweetheart | sleevless | natural | polyster | chiffon | beading | null |
| 270 | 5.79010251E8 | party | high | 4.6 | L | Spring | slash-neck | sleevless | empire | silk | chiffon | beading | null |
| 271 | 7.51364623E8 | party | Average | 4.8 | L | | Sweetheart | sleevless | empire | null | null | pleat | null |
| 272 | 8.98316819E8 | party | Medium | 4.6 | S | Winter | Scoop | short | empire | null | null | crystal | null |
| 273 | 6.22667306E8 | party | very-high | 4.9 | M | Spring | Sweetheart | sleevless | natural | silk | chiffon | beading | null |
| 274 | 8.09344968E8 | party | Medium | 4.8 | S | Spring | v-neck | sleevless | empire | null | null | null | null |
| 275 | 7.54138176E8 | party | very-high | 4.2 | L | Winter | v-neck | sleevless | null | null | null | lace | null |

Figure 3. Show Missing values in gray box

3.4 Examination of dataset using statistical methods:

When we load the dataset in Weka, in the right side we can see information about the selected attribute, like its value and how many times an instance in the dataset has a particular value, mean and Standard Deviation matures for numeric values.And the class attribute is numeric so to achieve classification requirements we must change the class from numeric to nominal using the path:
Choose→filters→unsupervised →attribute → NumericToNominal
Also with this dataset we have ID attribute ,so we have to delete this attribute because it is full discrete and not give any benefit to classification .

**4. Material and Methods**: Describe the preprocessing technique used to each attribute in the dataset:

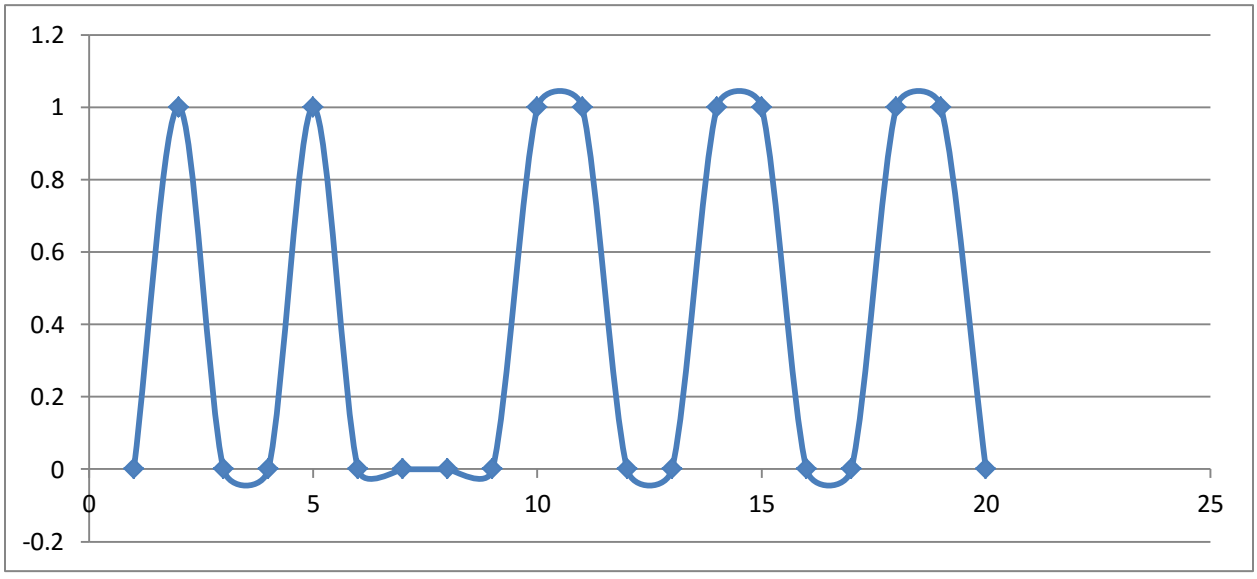



Figure 4 sample of raw data

4.1 Data cleaning: my dataset (Dresses Sales) need to clean from these problems:

- Incomplete: lacking attribute values, missing values within 4 records, so we can delete these records.
- inconsistence : containing discrepancies in codes or names, in this dataset we have duplicate attribute values :

Selected attribute

Name: Season Missing: 0 (0%) Distinct: 8 Type: Nominal Unique: 1 (0%)

| No. | Label | Count | Weight |
|---|---|---|---|
| 1 | Summer | 160 | 160.0 |
| 2 | Autumn | 61 | 61.0 |
| 3 | Spring | 122 | 122.0 |
| 4 | Winter | 99 | 99.0 |
| 5 | spring | 2 | 2.0 |
| 6 | winter | 46 | 46.0 |
| 7 | summer | 1 | 1.0 |

Figure 5. example inconsistence name of attribute

So we can handle this problem using **notepad++** by follow the path **Search → Replace**

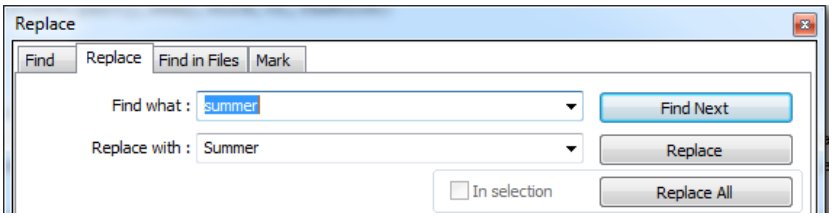
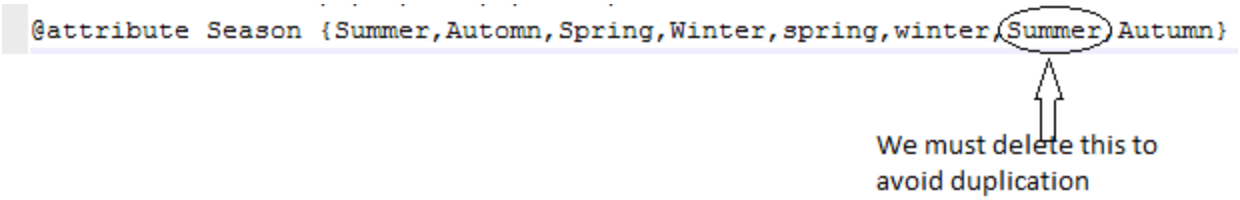



Figure 6. Replace window in notepad++

After click on Replace all the result is:




Now we still have same problem with **Autumn** and it is duplication value **Automn**, **Spring** and it is duplication value **spring**, and **Winter** with its duplication value **winter**, so we can handle this problem with same procedure as above, and the result by checking with Weka is :

Selected attribute

Name: Season Missing: 2 (0%) Distinct: 8 Type: Nominal Unique: 1 (0%)

| No. | Label | Count | Weight |
|---|---|---|---|
| 1 | Summer | 159 | 159.0 |
| 2 | Automn | 61 | 61.0 |
| 3 | Spring | 122 | 122.0 |
| 4 | Winter | 99 | 99.0 |
| 5 | spring | 2 | 2.0 |
| 6 | winter | 46 | 46.0 |
| 7 | summer | 1 | 1.0 |

Figure 7. Attribute Season before remove duplicate names

Selected attribute

Name: Season Missing: 0 (0%) Distinct: 4 Type: Nominal Unique: 0 (0%)

| No. | Label | Count | Weight |
|---|---|---|---|
| 1 | Summer | 161 | 161.0 |
| 2 | Autumn | 69 | 69.0 |
| 3 | Spring | 124 | 124.0 |
| 4 | Winter | 145 | 145.0 |

Figure 8. Attribute Season after remove duplicate names

We have same problem with attributes, NeckLine(Sweetheart- sweetheart), Style(Sexy- sexy), price(Low-low,High-high),size(small,s,S),sleevelength(sleeveless-sleevless-sleeevless sleeveless , cap_sleeves - capsleeves , halesleeve - half ), Meterial (chiffon - chiffonfabric) , so we can solve this problem with same procedure.

- noisy: containing errors or outliers : to check if we have any outliers in our dataset (Dresses Sales)we use the filter ***InterquartileRange*** , this filter adds new attributes that point out whether values of instances can be considered outliers or extreme values.

  Choose→ filters → unsupervised→ attribute→ ***InterquartileRange*** then click apply and the result is:

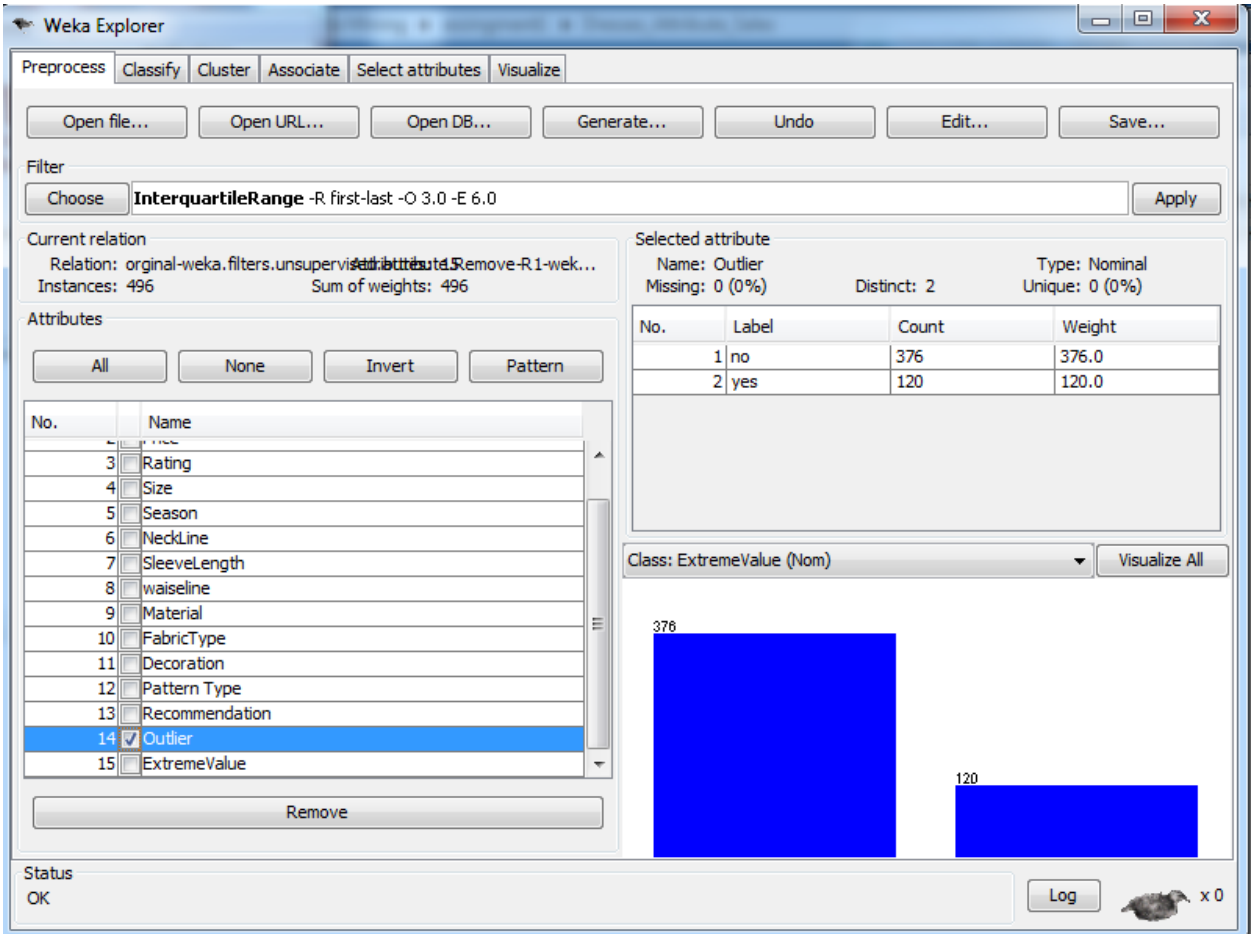



Figure 9. the Outliers after implement IQR filter on the dataset

As we show in figure above the filter add new attributes (outlier and extreme values) so referring to this result we have 120 outlier in our dataset, and about extreme values the result is zero so we don't have any extreme values.Now to remove outlier values we can use the path :

Choose→ filters → unsupervised→ instance→RemoveWithValues→ set the properties of attributIndes to 14(the index of outlier attribute) and nominalIndices to last(to remove just the last values that have the value yes) and after applying the result is:

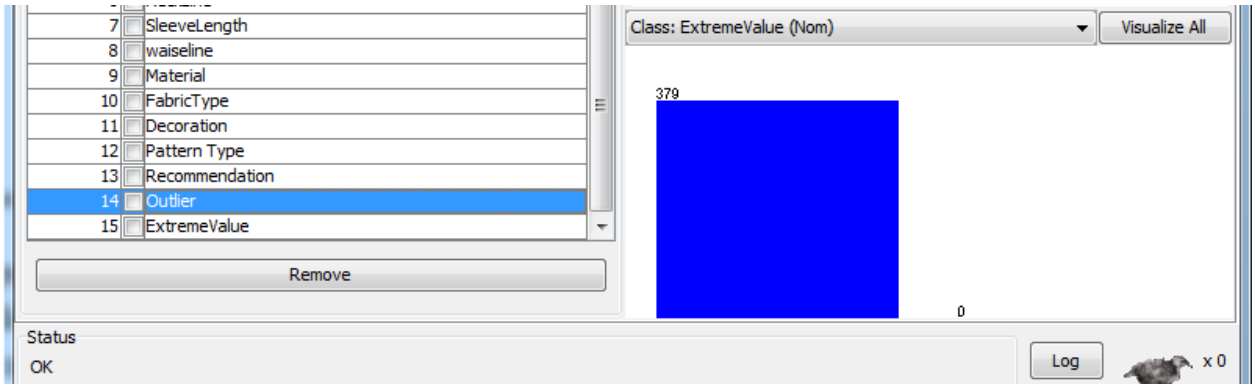



Figure 10. Removing the Outliers 1

**4.2 Data reduction**: get a new data representation that is considerably smaller in magnitude but so far produces the same (or roughly same) analytical results. Data reduction strategies:

- Numerosity reduction : we can be applied Numerosity reduction technique to decrease the data volume by choosing alternative, smaller forms of data representation

Here we change the attribute values from nominal to numeric, as example:
Attribute season: Summer, Autumn, Spring, Winter → Change to: 1,2,3,4 And so on for all nominal values.

Some values take a large scale as example attribute material from 0-23. So to reduce this scale we use the filter discretizes using:
Style 12→6 NeckLine 16→8 SleeveLength 11→5 Material 24→10 FabricType 22→10 Decoration 24→12 PatternType 14→7
Choose →Filters → unsupervised→attribute→discretize →

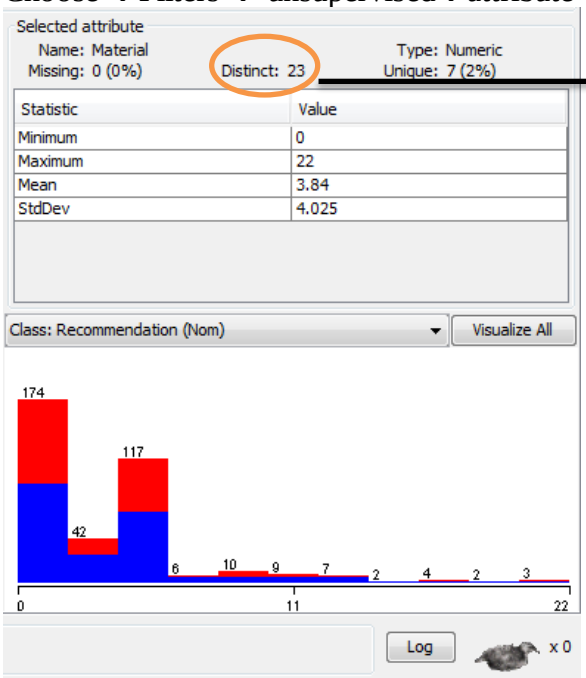



Figure 11. Example of attributes before discretization

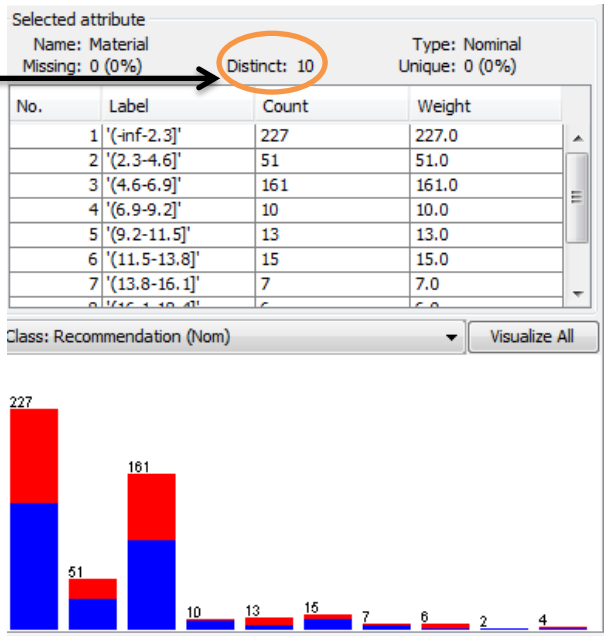



Figure 12. Example of attributes after discretization

And so on for all high scale attribute.

- Dimensionality reduction: Dimensionality reduction reduces the dataset volume by removing irrelevant attributes.

- Here we use the filter AttributeSelection:
  Choose→filters→supervized→attribute→AttributeSelection And the result is:

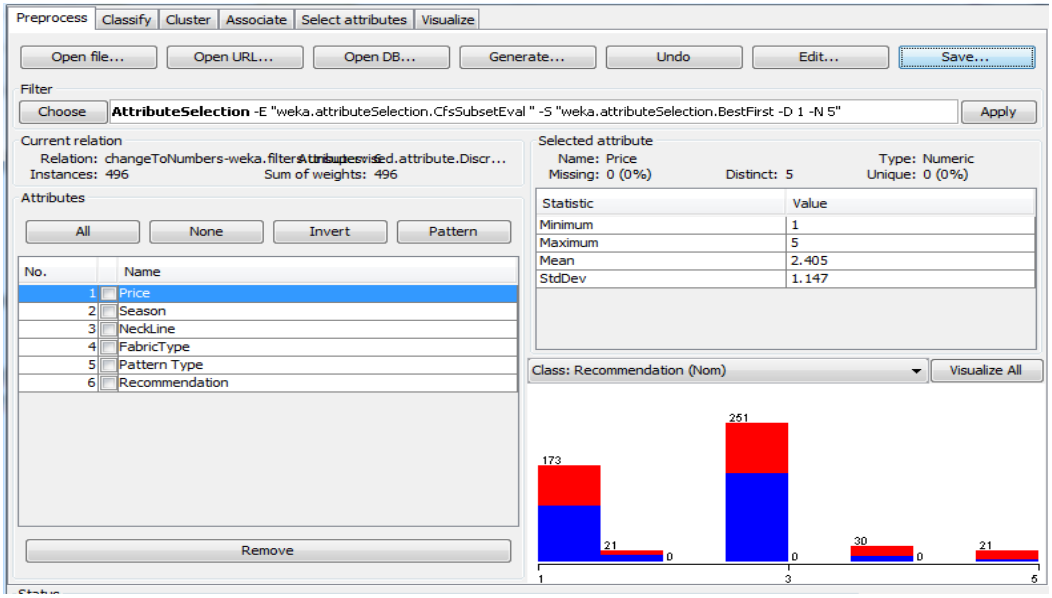



Figure 13. Attributes after applying attribute selection

**5 Classification**: Classification is a machine learning and Data mining technique applied to predict membership for a particular group of data instances. It's the problem of finding the model for class attribute as a function of the values of other attributes and predicting an accurate class assignment for test dataset. And the classification algorithms we will use in this work are:

### 5.1 Decision tree:

Decision tree can be used as a model for sequential decision problems under uncertainty. A decision tree describes graphically the decisions to be made, the events that may occur, and the outcomes associated with combinations of decisions and events. Probabilities are assigned to the events, and values are determined for each outcome. A major goal of the analysis is to determine the best decisions. decision tree advantages are (it is easy to fit, easy to employ, and easy to interpret as a fixed sequence of simple tests, It is non-linear, so It is work much better than linear models for highly non-linear functions).and disadvantage of decision tree is it can be extremely sensitive to small confusion in the data: a small change can result in a Significantly different tree. It can easily overfit. This can be eliminating by validation methods and pruning, but this is a grey area. It can have problems out-of-sample prediction. [8] introduced an extensions to the Decision tree construction which enhance the efficiency and utility of the method for a movie recommendation service, and implemented decision tree learning algorithm on the Netflix training data (100Mratings), and then evaluated its performance on the Netflix test data (2.8M ratings). [9] Described a case study of the exploitation of Decision Trees for creating an industrial recommender system. The aim of this system is to recommend items of a fashion retail store chain in Spain, producing leaflets for loyal customers announcing new products that they are likely to want to purchase.

[3] Proposed medical advices recommendation system based on a hybrid method using multiple classifications and unified Collaborative Filtering. Multiple classifications based on decision tree algorithms are applied to build an accurate predictive model that predicts the disease risk diagnosis for the monitored cases.
[10] Investigated the assumption that decision trees perform better than twenty other classification algorithms in classifying binary data by comparing the decision trees with a fuzzy set-based classifier and the naive Bayes on real and artificial datasets. And in the field of job recommender systems[11] presented recommend jobs to the candidates according to their matching profiles using decision tree where In these each internal node represents a condition, every edge coming out represents the choice and every leaf node represents the classification or the decision.

**5.2 Naïve Bayes**

The Naive Bayesian (NB) classifier is based on Bayes' theorem with independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods. An advantage of NB classifier is that it only needs a small size of training data to assessment the parameters (means and variances of the variables) necessary for classification. Because independent variables are supposed, only the variances of the variables for each class need to be determined and not the entire covariance matrix. The disadvantage is the conditional independence of class assumption by the Naïve Bayes is not always true, thus leading to low accuracy in some cases. [12] Presented a unique switching hybrid recommendation approach by combining a Naive Bayes classification approach with the collaborative filtering. Experimental results on two different datasets, show that the proposed algorithm is scalable and provided better performance in terms of accuracy and coverage–than other algorithms while at the same time eliminates some recorded problems with the recommender systems, in the same area [13] proposed a Naïve Bayes Classifier technique that uses the user rating for the item and item information to construct a model with time awareness. Naïve Bayes classifier was used to construct a user model by combining user-item rating, item feature's information and time information, which improve recommendation accuracy and can provide an item to user preference at that time which can satisfy users more.
[14]presented content based methods for recommendation systems which are mainly based on naïve Bayesian machine learning algorithm, presented a study of techniques which suggest naïve Bayesian algorithm for similarity in recommendation systems.[15] proposes a book recommendation technique based on opinion mining and Naïve Bayes classifier to recommend top ranking books to buyers . the author extracted, summarize and categorize all the customer reviews of a book, also considered the important factor like price of the book while recommendation and presented a novel tabular efficient method for recommending books to the buyer, especially when the buyer is coming first time to the website. [16] Presented a Naive Bayes Classifier Weighing Technique that applied to use with Singular Value Decomposition Technique for solving sparsity problem. The comparison Mean Absolute Error between Hybrid Naive Bayes Classifier Weighing and Singular Value Decomposition Technique and Pure Singular Value Decomposition Technique that found Hybrid Naive Bayes Classifier Weighing and Singular Value Decomposition yield lower Mean Absolute Error than Singular Value Decomposition.

### 5.3 Artificial Neural Network

Artificial Neural Network (ANN) get inspired from the biological nervous systems way, like the brain, process information. The main ingredient of this algorithm is the unfamiliar structure of the information processing system. It is consists of a large number of strongly interconnected processing elements (neurons) working in harmony to solve particular problems. ANNs, similar to human, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.

Some advantages of neural networks:

1. More like a real nervous system.
2. Parallel organization permits solutions to problems where multiple constraints must be satisfied simultaneously.
3. Graceful degradation.
4. Rules are implicit rather than explicit.

A disadvantage:

1. Difficulty with infinite recursion and structured representations.

Many researchers using ANN in their works like [17] that proposed a neural network hybrid recommender system have the ability to supply customers, associated with XML-based personal agents within a multi-agent system called MARF, with suggestions about flights purchases. MARF agents continuously monitor customers' interests and preferences in their commercial Web activities, by constructing and automatically maintaining their profiles. In order to highlight the benefits provided by the proposed flight recommender. In [18] a combination of content-based, model and memory-based collaborative filtering techniques is used in order to remove the drawbacks of Content based recommender systems, and to present predicted ratings more accurately. The training of the data is done using feedforward backpropagation neural network and the system performance is analyzed under various circumstances like number of users, their ratings and system model. [19] Studied Artificial Neural Network and optimized it based on classification system. The performance of the ANN based on classification system was evaluated by varying number of generations and computing accuracy at different factors. Three standard data had been used to computer the accuracy. [20] Presented road detector that extracts curb and navigable surface information from a multilayer laser sensor data. The road data was trained with an artificial neural network and classified into eight road geometries: straight road, right turn, left turn, right side road, left side road, T intersection, Y intersection and crossroad. The main advantage of this method is its robustness to light variations for detecting distinct roads even in the presence of noisy data thanks to the ANN. [21] studied a RS for movie lovers using neural networks in collaborative filtering systems for consumers' experiential decisions. The experimental results reveal that it not only improves the accuracy of predicting movie ratings but also increases data transfer rates and provides richer user experiences.

## 6. EXPERIMENT AND RESULT

We apply **decision tree** algorithm with ten different models of dividing the dataset to training and testing Ranging from (90:10) to (10:90) and the result as shown in table2.

| Model | | Results for 5 attributes and the class | | | | | Results for 12 attributes and the class | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Model num. | Data allocation | accuracy | Num. of rules | Len. of rules | Mean Square Error | | accuracy | Num. of rules | Len. of rules | Mean Square Error |
| 1- | 90:10 | 66 % | 37 | 19 | 0.4715 | | **65.7895 %** | 63 | 32 | 0.5064 |
| 2- | 80:20 | 67.6768 % | 37 | 19 | 0.4903 | | 58.6667 % | 63 | 32 | 0.5103 |
| 3- | **70:30** | **69.1275 %** | 37 | 19 | 0.4762 | | 52.2124 % | 63 | 32 | 0.5816 |
| 4- | 60:40 | 59.0909 % | 37 | 19 | 0.4968 | | 54.6667 % | 63 | 32 | 0.5679 |
| 5- | 50:50 | 63.7097 % | 37 | 19 | 0.489 | | 61.1702 % | 63 | 32 | 0.5181 |
| 6- | 40:60 | 64.094 % | 37 | 19 | 0.4795 | | 55.3097 % | 63 | 32 | 0.6245 |
| 7- | 30:70 | 57.6369 % | 37 | 19 | 0.502 | | 55.5133 % | 63 | 32 | 0.5606 |
| 8- | 20:80 | 60.7053 % | 37 | 19 | 0.5181 | | 52.4917 % | 63 | 32 | 0.563 |
| 9- | 10:90 | 54.7085 % | 37 | 19 | 0.5498 | | 51.4793 % | 63 | 32 | 0.6231 |
| 10- | 66:34 | 65.6805 % | 37 | 19 | 0.4725 | | 53.125 % | 63 | 32 | 0.5394 |

Table 2. DT accuracy Results for 10 models using 5 attribute and 12 attribute

Then apply **Naïve Bayes** algorithm with ten different attempts of dividing the dataset to training and testing Ranging from (90:10) to (10:90) and the result as shown in table3.

| Model | | Results for 5 attributes and the class | | | Results for 12 attributes and the class | |
|---|---|---|---|---|---|---|
| Model num. | Data allocation | accuracy | MSE(Mean Square Error) | | accuracy | MSE(Mean Square Error) |
| 1- | **90:10** | **72 %** | 0.4354 | | 57.8947 % | 0.5032 |
| 2- | 80:20 | 65.6566 % | 0.4593 | | **62.6667 %** | 0.475 |
| 3- | 70:30 | 64.4295 % | 0.461 | | 53.0973 % | 0.5034 |
| 4- | 60:40 | 62.6263 % | 0.4832 | | 56 % | 0.4998 |
| 5- | 50:50 | 61.2903 % | 0.482 | | 58.5106 % | 0.4964 |
| 6- | 40:60 | 63.4228 % | 0.4799 | | 58.8496 % | 0.4987 |
| 7- | 30:70 | 61.9597 % | 0.4832 | | 55.8935 % | 0.5158 |
| 8- | 20:80 | 60.2015 % | 0.5065 | | 53.4884 % | 0.5381 |
| 9- | 10:90 | 56.9507 % | 0.5331 | | 51.4793 % | 0.5698 |
| 10- | 66:34 | 65.6805 % | 0.4601 | | 52.3438 % | 0.5097 |

Table 3. NB accuracy Results for 10 models using 5 attribute and 12 attribute

Lastly, we apply **Neural Network** (Multi-Layer Perceptron) algorithm using two Hidden layers with ten different attempts of dividing the dataset to training and testing Ranging from (90:10) to (10:90) and the result as shown in table4.

| Model | | Results for 5 attributes and the class | | | | | | Results for 12 attributes and the class | |
|---|---|---|---|---|---|---|---|---|---|
| Model num. | Data allocation | Num. Hidden Nodes | Activation Function | Learning rate | Accuracy | MSE (Mean Square Error) | | Accuracy | MSE (Mean Square Error) |
| 1- | 90:10 | 2 | Sigmoid | 0.3 | **76 %** | 0.4316 | | 44.7368 % | 0.5531 |
| 2- | 80:20 | 2 | Sigmoid | 0.3 | 66.6667 % | 0.5075 | | 56 % | 0.5111 |
| 3- | 70:30 | 2 | Sigmoid | 0.3 | 65.1007 % | 0.498 | | 46.0177 % | 0.5603 |
| 4- | 60:40 | 2 | Sigmoid | 0.3 | 51.5152 % | 0.5185 | | **57.3333 %** | 0.539 |
| 5- | 50:50 | 2 | Sigmoid | 0.3 | 58.4677 % | 0.5088 | | 53.7234 % | 0.6002 |
| 6- | 40:60 | 2 | Sigmoid | 0.3 | 52.349 % | 0.527 | | 50 % | 0.6431 |
| 7- | 30:70 | 2 | Sigmoid | 0.3 | 53.6023 % | 0.5416 | | 52.4715 % | 0.6027 |
| 8- | 20:80 | 2 | Sigmoid | 0.3 | 54.1562 % | 0.562 | | 50.1661 % | 0.658 |
| 9- | 10:90 | 2 | Sigmoid | 0.3 | 57.6233 % | 0.5735 | | 52.3669 % | 0.6466 |
| 10- | 66:34 | 2 | Sigmoid | 0.3 | 64.497 % | 0.4829 | | 53.9063 % | 0.5778 |

Table 4. ANN accuracy Results for 10 models using 5 attribute and 12 attribute

| Model | Decision Tree | Naïve Bayes | Neural Network |
|---|---|---|---|
| 90:10 | 66 % | 72 % | 76 % |
| 80:20 | 67.6768 % | 65.6566 % | 66.6667 % |
| 70:30 | 69.1275 % | 64.4295 % | 65.1007 % |
| 60:40 | 59.0909 % | 62.6263 % | 51.5152 % |
| 50:50 | 63.7097 % | 61.2903 % | 58.4677 % |
| 40:60 | 64.094 % | 63.4228 % | 52.349 % |
| 30:70 | 57.6369 % | 61.9597 % | 53.6023 % |
| 20:80 | 60.7053 % | 60.2015 % | 54.1562 % |
| 10:90 | 54.7085 % | 56.9507 % | 57.6233 % |
| 66:34 | 65.6805 % | 65.6805 % | 64.497 % |

Table 5 comparative of accuracy results for DT, NB and NN

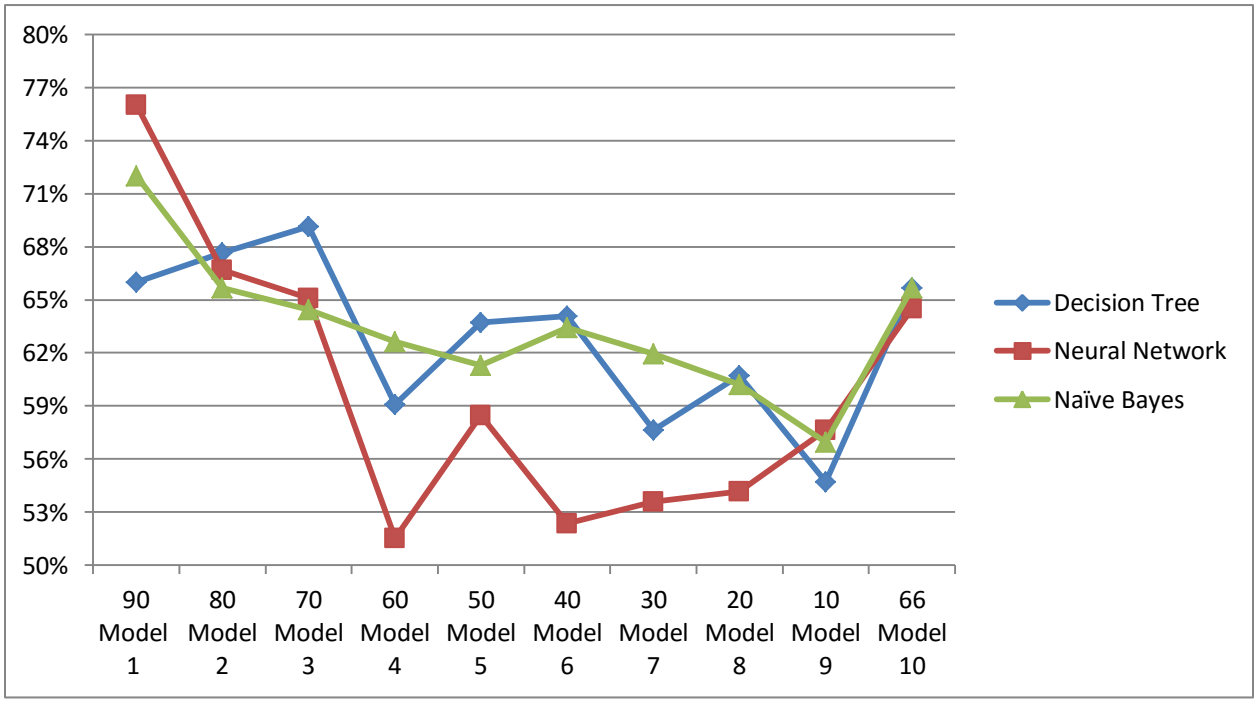



Figure 17. Comparative of accuracy results for DT, NB and NN

## 7 Conclusions

We used a dataset (dresses recommendation) from UCI, firstly we preprocess the dataset using Weka then we put the resulting preprocessed data in classification models for three algorithms (DT, NB, ANN). Because the Weka choose just (5) attribute by applying AttributeSelecton, We try to put the dataset with original number of attributes (12) in classifiers models to see the difference between the result using AttributeSelection and without using it, and as we can see from results the dataset give better results for all methods after attribute selection, that mean the existence of irrelevant attribute decrease the accuracy. Then we examine the best performance for the three techniques by comparative the accuracy and mean square error using the dataset after applying AttributeSelection. And as we can see our learning procedure is performing better with Naïve Bayes model in terms of higher accuracy rate and lower mean square error rate.

References:


1. N Padhy, Dr. P. Mishra, "The Survey of Data Mining Applications And Feature Scope" , (IJCSEIT) International Journal of Computer Science, Engineering and Information Technology, Vol.2, No.3, June 2012 .
2. Guerra L, McGarry M, Robles V, Bielza C, Larrañaga P, Yuste R. ,"Comparison between supervised and unsupervised classifications of neuronal cell types: A case study. Developmental neurobiology" , 71(1): 71-82,(2011).
3. Asmaa S. Hussein, Wail M. Omar, Xue Li, ModafarAti," Efficient Chronic Disease Diagnosis Prediction and Recommendation System" , IEEE EMBS International Conference on Biomedical Engineering and Sciences I Langkawi I 17th - 19th December 2012.
4. Binge Cui, Xin Chen," An Online Book Recommendation System Based on Web Service", Sixth International Conference on Fuzzy Systems and Knowledge Discovery,IEEE,2009.
5. Shahab Saquib Sohail, Jamshed Siddiqui, Rashid Ali," Book Recommendation System Using OpinionMining Technique",IEEE,2013.
6. Sajal Halder, A. M. Jehad Sarkar, Young-Koo Lee," Movie Recommendation System Based on Movie Swarm", Second International Conference on Cloud and Green Computing,IEEE,2012.
7. Abinash, Vineet, An Approach to Content Based Recommender Systems using Decision List based Classification with k-DNF Rule Set, International Conference on Information Technology , 2014.
8. Nadav Golbandi, Yehuda Koren, Ronny Lempel," Adaptive Bootstrapping of Recommender Systems Using Decision Trees", ACM 978-1-4503-0493-1/11/02,2011.
9. Iván Cantador, Desmond Elliott, Joemon M. Jose," A Case Study of Exploiting Decision Trees for an Industrial Recommender System", Lilybank Gardens, Glasgow, G12 8QQ, UK,2009.
10. Sofia Visa, Anca Ralescu, Mircea Ionescu," Investigating Learning Methods for Binary Data", IEEE,2007.
11. Anika Gupta, Dr. Deepak Garg," Applying Data Mining Techniques in Job Recommender System for Considering Candidate Job Preferences",IEEE,2014.
12. Mustansar Ali Ghazanfar and Adam Pr¨ugel-Bennett," An Improved Switching Hybrid Recommender System Using Naive Bayes Classifier and Collaborative Filtering", School of Electronics and Computer Science, University of Southampton, Highfield Campus, SO17 1BJ, United Kingdom,2010.
13. Sutheera Puntheeranurak, Pongpan Pitakpaisarnsin," Time-aware Recommender System Using Naïve Bayes Classifier Weighting Technique", 2nd International Symposium on Computer, Communication, Control and Automation (3CA 2013).
14. Meghna Khatri," A Survey of Naïve Bayesian Algorithms for Similarity in Recommendation Systems", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 5, May 2012.
15. Anand Shanker Tewari, Tasif Sultan Ansari, Asim Gopal Barman," Opinion Based Book Recommendation Using Naïve Bayes Classifier",IEEE,2014.
16. Sutheera Puntheeranurak, Supitchaya Sanprasert," Hybrid Naive Bayes Classifier Weighting and Singular Value Decomposition Technique for Recommender System",IEEE,2011.
17. Maria Nadia Postorino, Giuseppe M. L. Sarne ," A Neural Network Hybrid Recommender System, Proceedings of the 2011 conference on Neural Nets WIRN10: Proceedings of the 20th Italian Workshop on Neural Net,2012.
18. Anant Gupta, Dr. B. K. Tripathy," A Generic Hybrid Recommender System based onNeural Networks",IEEE.2014.
19. M.K.Kavitha Devi, R.Thirumalai Samy, S.Vinoth Kumar, Dr.P.Venkatesh," Probabilistic Neural Network approach to Alleviate sparsity and cold start problems in Collaborative Recommender Systems",IEEE,2010.
20. Alberto Y. Hata, Danilo Habermann, Fernando S. Osorio1, and Denis F. Wolf," Road Geometry Classification using ANN", 20141EEE Intelligent Vehicles Symposium (IV)June 8-11, 2014. Dearborn, Michigan, USA.
21. Arthur J. Lina, Chien-Lung Hsub, Eldon Y. Lic,," Improving the effectiveness of experiential decisions by recommendation systems",ACM,2014.
22. Machine Learning Repository (UCI) , https://archive.ics.uci.edu/ml/index.html.
23. R. Kirkby,E. Frank," WEKA Explorer User", University of Waikato, November 9, 2004.

Appendix

| Technique | Mean Square Error |
|---|---|
| Decision Tree | 0.49457 |
| Naïve Bayes | 0.47837 |
| Neural Network | 0.51514 |

Table 6. Comparative of Mean Accuracy for DT, NB and NN

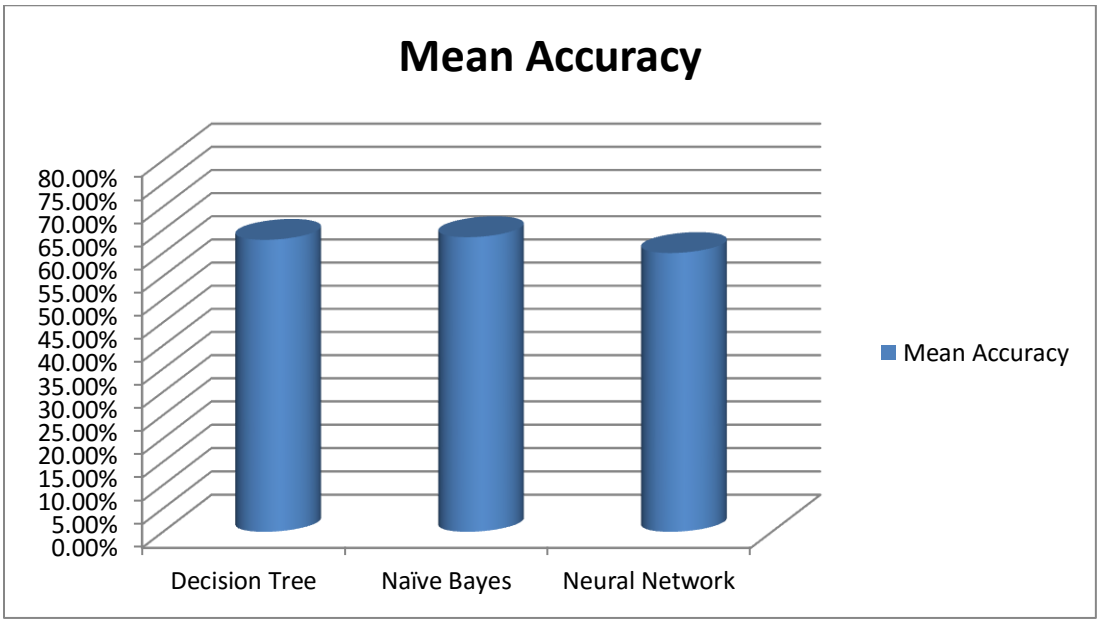



Figure 18. Comparative of Mean for Accuracy and Mean Square Error for DT, NB and NN

| No. | Dress_ID Numeric | Style Nominal | Price Nominal | Rating Numeric | Size Nominal | Season Nominal | NeckLine Nominal | SleeveLength Nominal | waiseline Nominal | Material Nominal | FabricType Nominal | Decoration Nominal | Pattern Type Nominal | **Recommendation** Numeric |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.006032852E9 | Sexy | Low | 4.6 | M | Summer | o-neck | sleevless | empire | null | chiffon | ruffles | animal | 1.0 |
| 2 | 1.212192089E9 | Casual | Low | 0.0 | L | Summer | o-neck | Petal | natural | microfi... | null | ruffles | animal | 0.0 |
| 3 | 1.190380701E9 | vintage | High | 0.0 | L | Automn | o-neck | full | natural | polyster | null | null | print | 0.0 |
| 4 | 9.66005983E8 | Brief | Average | 4.6 | L | Spring | o-neck | full | natural | silk | chiffon | embroidary | print | 1.0 |
| 5 | 8.76339541E8 | cute | Low | 4.5 | M | Summer | o-neck | butterfly | natural | chiffon... | chiffon | bow | dot | 0.0 |
| 6 | 1.068332458E9 | bohem... | Low | 0.0 | M | Summer | v-neck | sleevless | empire | null | null | null | print | 0.0 |
| 7 | 1.220707172E9 | Casual | Average | 0.0 | XL | Summer | o-neck | full | null | cotton | null | null | solid | 0.0 |
| 8 | 1.219677488E9 | Novelty | Average | 0.0 | free | Automn | o-neck | short | natural | polyster | broadcloth | lace | null | 0.0 |
| 9 | 1.113094204E9 | Flare | Average | 0.0 | free | Spring | v-neck | short | empire | cotton | broadcloth | beading | solid | 1.0 |
| 10 | 9.85292672E8 | bohem... | Low | 0.0 | free | Summer | v-neck | sleevless | natural | nylon | chiffon | null | null | 1.0 |
| 11 | 1.117293701E9 | party | Average | 5.0 | free | Summer | o-neck | full | natural | polyster | broadcloth | lace | solid | 0.0 |
| 12 | 8.9848153E8 | Flare | Average | 0.0 | free | Spring | v-neck | short | null | nylon | null | null | animal | 0.0 |
| 13 | 9.57723897E8 | sexy | Low | 4.7 | M | Winter | o-neck | threequarter | null | null | chiffon | lace | print | 1.0 |
| 14 | 7.49031896E8 | vintage | Average | 4.8 | M | Summer | o-neck | short | empire | cotton | jersey | null | animal | 1.0 |
| 15 | 1.055411544E9 | Casual | Low | 5.0 | M | Summer | boat-neck | short | null | cotton | null | sashes | solid | 0.0 |
| 16 | 1.162628131E9 | Casual | Low | 0.0 | free | Winter | boat-neck | full | null | other | other | lace | null | 0.0 |
| 17 | 6.24314841E8 | cute | Average | 4.7 | L | spring | o-neck | short | null | cotton | null | sashes | solid | 1.0 |
| 18 | 8.30467746E8 | bohem... | Medium | 5.0 | free | Automn | o-neck | full | natural | null | null | hollowout | patchwork | 1.0 |
| 19 | 8.40857118E8 | Brief | Average | 0.0 | M | Winter | peterpa... | threequarter | natural | cotton | null | null | patchwork | 0.0 |
| 20 | 1.113221101E9 | Sexy | Average | 5.0 | M | Automn | o-neck | sleevless | empire | milksilk | null | null | null | 1.0 |
| 21 | 8.61754372E8 | Sexy | Average | 4.5 | L | Automn | o-neck | full | null | cotton | null | beading | solid | 0.0 |
| 22 | 8.561781E8 | Casual | Low | 4.3 | M | Summer | o-neck | sleevless | natural | null | chiffon | null | solid | 0.0 |
| 23 | 1.122989777E9 | Brief | Low | 4.0 | XL | Summer | v-neck | short | natural | cotton | null | pockets | solid | 0.0 |
| 24 | 8.40516484E8 | Sexy | Average | 4.7 | S | Summer | v-neck | sleevless | empire | cotton | null | sequined | solid | 1.0 |

Figure 19. Example of raw dataset

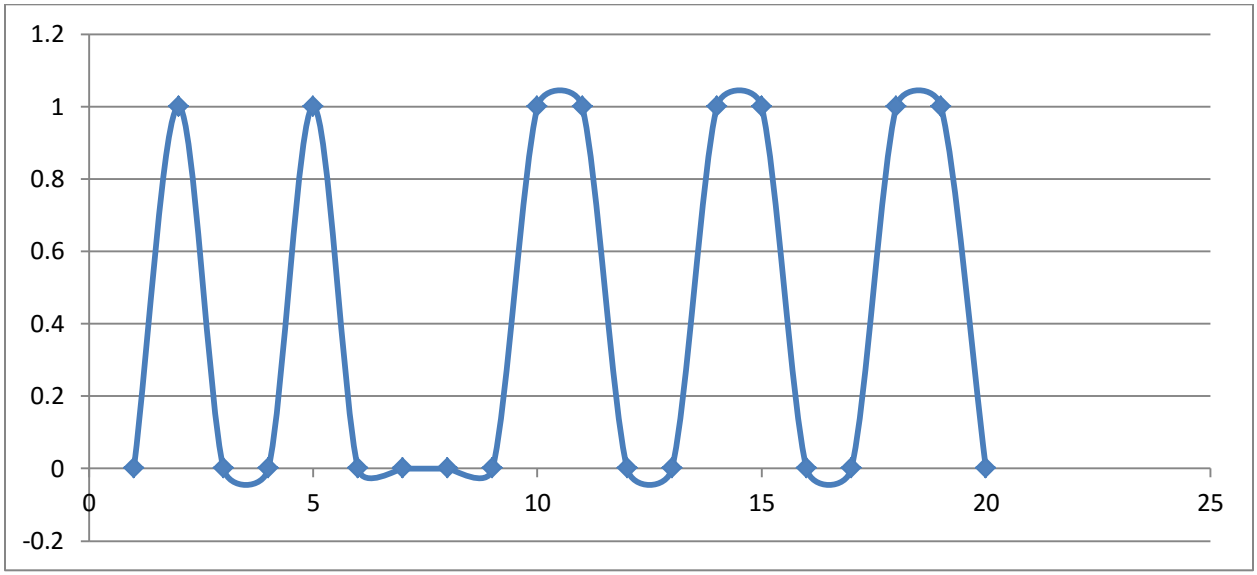



Figure 20. Sample of raw dataset

| No. | Style Numeric | Price Numeric | Rating Numeric | Size Numeric | Season Numeric | NeckLine Numeric | SleeveLength Numeric | waiseline Numeric | Material Numeric | FabricType Numeric | Decoration Numeric | Pattern Type Numeric | Recommendatio Nominal |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 114 | 4.0 | 3.0 | 4.6 | 5.0 | 4.0 | 1.0 | 2.0 | 2.0 | 12.0 | 0.0 | 6.0 | 4.0 | 0 |
| 81 | 9.0 | 3.0 | 4.8 | 3.0 | 4.0 | 2.0 | 4.0 | 2.0 | 12.0 | 2.0 | 4.0 | 4.0 | 0 |
| 116 | 2.0 | 1.0 | 4.8 | 4.0 | 1.0 | 1.0 | 4.0 | 1.0 | 12.0 | 1.0 | 6.0 | 7.0 | 0 |
| 120 | 4.0 | 3.0 | 4.3 | 2.0 | 3.0 | 1.0 | 2.0 | 2.0 | 12.0 | 6.0 | 9.0 | 5.0 | 1 |
| 334 | 1.0 | 4.0 | 5.0 | 2.0 | 1.0 | 2.0 | 1.0 | 2.0 | 12.0 | 0.0 | 0.0 | 6.0 | 0 |
| 129 | 3.0 | 2.0 | 4.7 | 5.0 | 3.0 | 6.0 | 1.0 | 2.0 | 12.0 | 0.0 | 4.0 | 4.0 | 1 |
| 369 | 5.0 | 3.0 | 4.6 | 2.0 | 3.0 | 1.0 | 6.0 | 2.0 | 13.0 | 1.0 | 0.0 | 0.0 | 0 |
| 371 | 4.0 | 3.0 | 4.7 | 2.0 | 4.0 | 1.0 | 6.0 | 2.0 | 13.0 | 1.0 | 0.0 | 6.0 | 0 |
| 284 | 2.0 | 1.0 | 5.0 | 2.0 | 3.0 | 1.0 | 1.0 | 2.0 | 14.0 | 3.0 | 0.0 | 2.0 | 0 |
| 146 | 2.0 | 1.0 | 5.0 | 5.0 | 3.0 | 7.0 | 1.0 | 1.0 | 14.0 | 0.0 | 0.0 | 6.0 | 1 |
| 206 | 5.0 | 3.0 | 4.4 | 5.0 | 3.0 | 10.0 | 1.0 | 1.0 | 14.0 | 11.0 | 16.0 | 4.0 | 1 |
| 295 | 2.0 | 3.0 | 5.0 | 2.0 | 3.0 | 1.0 | 10.0 | 0.0 | 14.0 | 0.0 | 0.0 | 0.0 | 0 |
| 117 | 1.0 | 2.0 | 4.6 | 3.0 | 1.0 | 1.0 | 4.0 | 2.0 | 14.0 | 3.0 | 4.0 | 4.0 | 0 |
| 152 | 1.0 | 1.0 | 4.3 | 2.0 | 2.0 | 8.0 | 4.0 | 2.0 | 15.0 | 0.0 | 4.0 | 2.0 | 0 |
| 161 | 2.0 | 3.0 | 5.0 | 5.0 | 1.0 | 2.0 | 1.0 | 0.0 | 16.0 | 0.0 | 0.0 | 5.0 | 0 |
| 225 | 2.0 | 3.0 | 4.8 | 3.0 | 2.0 | 1.0 | 4.0 | 2.0 | 17.0 | 0.0 | 0.0 | 0.0 | 0 |
| 171 | 2.0 | 2.0 | 4.6 | 2.0 | 4.0 | 1.0 | 2.0 | 2.0 | 17.0 | 0.0 | 0.0 | 4.0 | 1 |
| 200 | 8.0 | 5.0 | 4.7 | 3.0 | 4.0 | 7.0 | 1.0 | 1.0 | 18.0 | 1.0 | 5.0 | 0.0 | 1 |
| 328 | 1.0 | 1.0 | 4.4 | 5.0 | 1.0 | 14.0 | 1.0 | 2.0 | 18.0 | 0.0 | 3.0 | 4.0 | 0 |
| 267 | 2.0 | 1.0 | 4.7 | 1.0 | 4.0 | 1.0 | 1.0 | 2.0 | 19.0 | 0.0 | 4.0 | 4.0 | 0 |
| 296 | 2.0 | 1.0 | 5.0 | 5.0 | 4.0 | 1.0 | 2.0 | 2.0 | 20.0 | 0.0 | 10.0 | 4.0 | 0 |
| 308 | 2.0 | 1.0 | 5.0 | 5.0 | 1.0 | 1.0 | 1.0 | 2.0 | 21.0 | 3.0 | 19.0 | 4.0 | 1 |
| 316 | 5.0 | 1.0 | 4.8 | 2.0 | 3.0 | 1.0 | 4.0 | 2.0 | 22.0 | 17.0 | 3.0 | 3.0 | 1 |

Undo OK Cancel

Figure 21. Example of transformation data from nominal to numeric

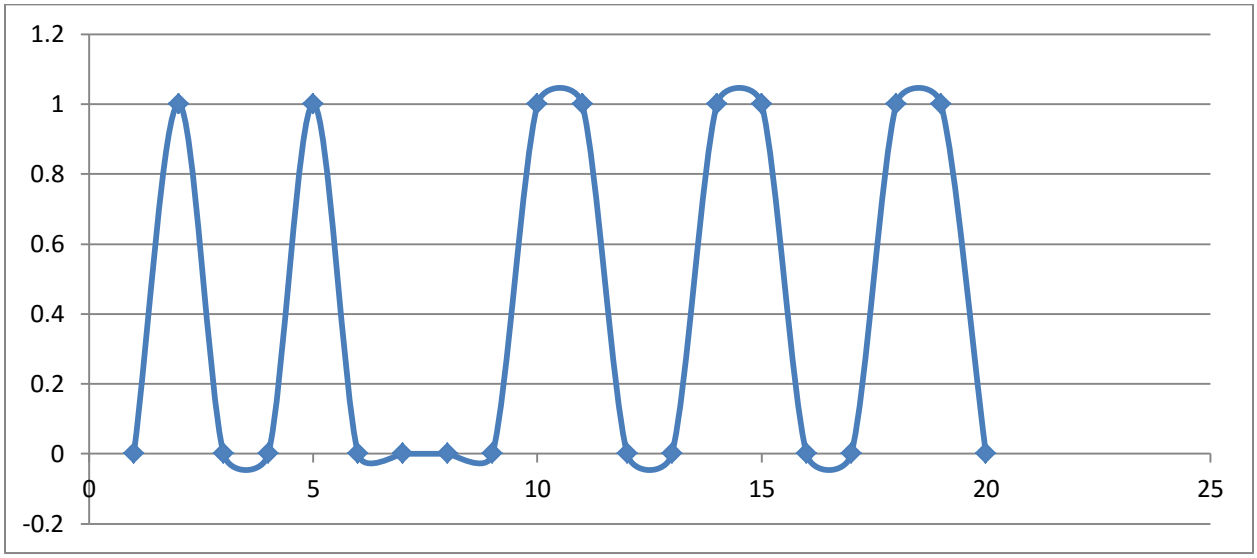



Figure 22. Sample of transformed dataset

| No. | Price Numeric | Season Numeric | NeckLine Nominal | FabricType Nominal | Pattern Type Nominal | **Recommendation** Nominal |
|---|---|---|---|---|---|---|
| 1 | 1.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 1 |
| 2 | 1.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 0 |
| 3 | 2.0 | 2.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 0 |
| 4 | 3.0 | 3.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 1 |
| 5 | 1.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 0 |
| 6 | 1.0 | 1.0 | '(1.875-... | '(-inf-2.2]' | '(-inf-2]' | 0 |
| 7 | 3.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 0 |
| 8 | 3.0 | 2.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 0 |
| 9 | 3.0 | 3.0 | '(1.875-... | '(-inf-2.2]' | '(2-4]' | 1 |
| 10 | 1.0 | 1.0 | '(1.875-... | '(-inf-2.2]' | '(-inf-2]' | 1 |
| 11 | 3.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 0 |
| 12 | 3.0 | 3.0 | '(1.875-... | '(-inf-2.2]' | '(-inf-2]' | 0 |
| 13 | 1.0 | 4.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 1 |
| 14 | 3.0 | 1.0 | '(-inf-1.... | '(2.2-4.4]' | '(-inf-2]' | 1 |
| 15 | 1.0 | 1.0 | '(1.875-... | '(-inf-2.2]' | '(2-4]' | 0 |
| 16 | 1.0 | 4.0 | '(1.875-... | '(2.2-4.4]' | '(-inf-2]' | 0 |
| 17 | 3.0 | 3.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 1 |
| 18 | 4.0 | 2.0 | '(-inf-1.... | '(-inf-2.2]' | '(4-6]' | 1 |
| 19 | 3.0 | 4.0 | '(3.75-5... | '(-inf-2.2]' | '(4-6]' | 0 |
| 20 | 3.0 | 2.0 | '(-inf-1.... | '(-inf-2.2]' | '(-inf-2]' | 1 |
| 21 | 3.0 | 2.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 0 |
| 22 | 1.0 | 1.0 | '(-inf-1.... | '(-inf-2.2]' | '(2-4]' | 0 |
| 23 | 1.0 | 1.0 | '(1.875-... | '(-inf-2.2]' | '(2-4]' | 0 |
| 24 | 3.0 | 1.0 | '(1.875-... | '(-inf-2.2]' | '(2-4]' | 1 |

Figure 25. Example of dataset after discretization and attributes selection

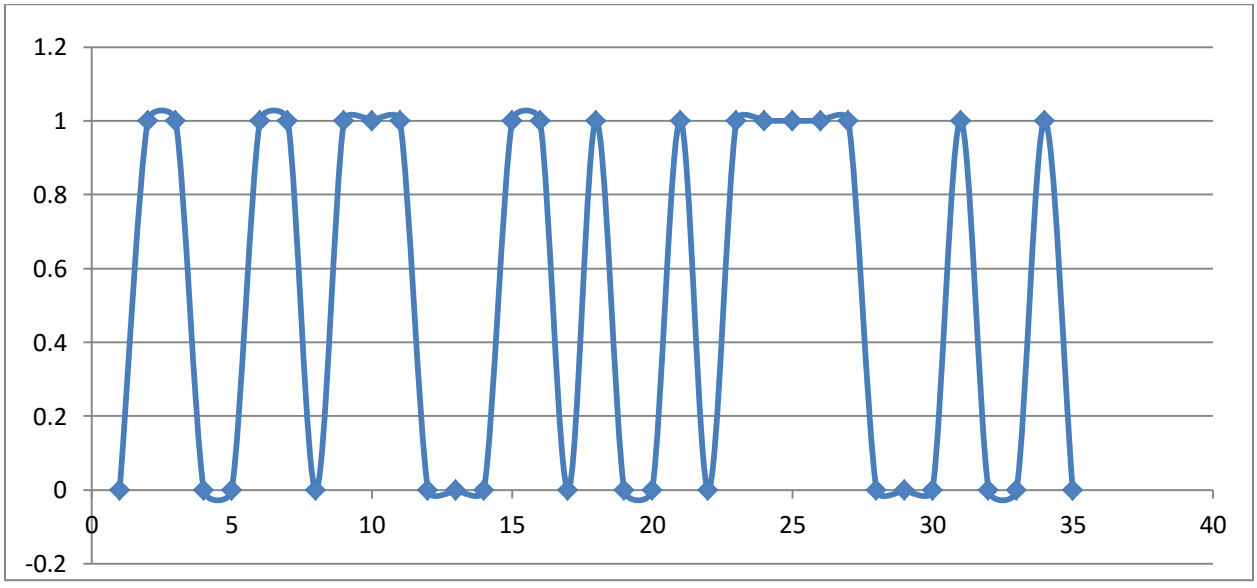



Figure 26. Sample of dataset after discretization and attribute selection