**Analysis of Fine Art Painting Dataset using WEKA**

ITCS 6155/8155 Knowledge Based System Course Project Spring 2017

**Project Team**

Rucha Kshirsagar

Prateek Mahendrakar

Nasheen Nur

Satish Kumar

**Instructor**

Dr. Zbigniew W. Ras

University of North Carolina at Charlotte, NC, USA

**Contents**

[Introduction 2](#_Toc480927404)

[Saatchi Art Dataset 2](#_Toc480927405)

[Columns Introduced 2](#_Toc480927406)

[Initial Data Set (Without Filter) 3](#_Toc480927407)

[Data Mining using WEKA 3](#_Toc480927408)

[Data Preparation 3](#_Toc480927409)

[Processed Data Set 3](#_Toc480927410)

[Process 4](#_Toc480927411)

[Classification 8](#_Toc480927412)

[Classifier Outputs 8](#_Toc480927413)

[a. Decision Table on Price 8](#_Toc480927414)

[b. Decision Table on no. of Views 9](#_Toc480927415)

[c. Random Forest on Views 11](#_Toc480927416)

[d. Random Forest on Price 12](#_Toc480927417)

[e. J48 on Price 13](#_Toc480927418)

[f. J48 on no. of Views 15](#_Toc480927419)

[g. JRip on no. of Views 17](#_Toc480927420)

[h. JRip on Price 18](#_Toc480927421)

[Interpretation of the Results 20](#_Toc480927422)

[Conclusion 2](#_Toc480927423)1

[References 2](#_Toc480927424)1

**Introduction**

The objective of this project is to build the database of minimum 260 fine art paintings listed on SaatchiArt website. The Exploratory Data Analysis is then performed in the dataset through pre-processing, evaluation of potential attributes, application of classification and clustering algorithms. We have proposed 4 new features (attributes) to describe these paintings in addition to the 9 features that were provided.

In this report, we will discuss how the database was built, how these attributes describe a painting alongside talking about the four newly added attributes, followed by a brief overview of WEKA tool, pre-processing, classification, and clustering techniques employed in the study with the corresponding result.

**Saatchi Art Dataset**

The Saatchi Art marketplace contains original paintings, photography, drawings, and sculpture by approximately 65,000 artists from 100 countries worldwide. Saatchi Art is selling and shipping artworks to collectors in over 80 countries. The company's online selection of artworks is curated by [Rebecca Wilson](https://en.wikipedia.org/wiki/Rebecca_Wilson_(curator)), Chief Curator of Saatchi Art and Vice President of Saatchi Art's Art Advisory service.[[1]](https://en.wikipedia.org/wiki/Saatchi_Art)

Web Site: <https://www.saatchiart.com/>

Data extracted from <https://www.saatchiart.com/paintings/fine-art>

The classification columns provided were

1. Name of the painting
2. Artist name
3. Size: The surface area of the painting = Height x Width
4. Price
5. Number of views
6. Number of favourites
7. Date
8. Subject
9. Medium

## **Columns Introduced**

We extended this dataset by adding four new features mentioned below;

1. Orientation of the paintings
2. Number of followers of the artist
3. Number of exhibitions plus events by the artist
4. Number of artworks of the artist.

The data we collected has 260 records varying over these 13 columns.

## **Initial Data Set (Without Filter)**

Below attached excel file has the initial data set we had collected from SaatchiArt web site



**Data Mining using WEKA**

Data mining is one approach that identifies the patterns in data and helps in making decisions by analysis of huge data sets[[3]](http://www.cs.waikato.ac.nz/~ml/weka/index.html" \o "Weka official site). **Weka** (Waikato Environment for Knowledge Analysis) is a popular suite of machine learning software written in Java, developed at the [University of Waikato](https://en.wikipedia.org/wiki/University_of_Waikato), [New Zealand](https://en.wikipedia.org/wiki/New_Zealand)). It is a workbench[[2](https://en.wikipedia.org/wiki/Weka_(machine_learning)" \l "cite_note-1" \o "Weka Wikipedia)]that contains a collection of visualization tools and algorithms for data analysis and [predictive modelling](https://en.wikipedia.org/wiki/Predictive_modeling), together with graphical user interfaces for easy access to these functions.

**Advantages of Weka include:**

* Free availability under the GNU Public License.
* Portability, since it is fully implemented in the [Java programming language](https://en.wikipedia.org/wiki/Java_programming_language) and thus runs on almost any modern computing platform.
* A comprehensive collection of data [pre-processing](https://en.wikipedia.org/wiki/Preprocessing) and modelling techniques.
* Ease of use due to its graphical user interfaces.

We have worked on data mining techniques using the Weka software. The overall goal of Weka is to build a state-of-the-art facility for developing machine learning (ML) techniques and allow people to apply them to real-world data mining problems.

**Data Preparation**

Listed below are common steps in the data preparation process.[[2]](https://en.wikipedia.org/wiki/Weka_(machine_learning))

* Removal of special characters in the dataset for it to be accepted in WEKA.
* Replaced the size of the painting which was in Height x Width x Thickness by product of Height and Width
* The values for newly added attributes are normalized.
* The attribute price is our decision feature which is discretized into three intervals.

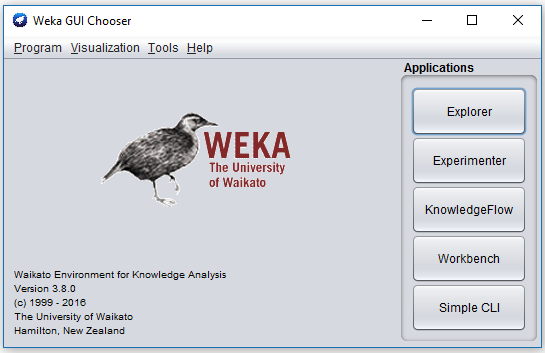
## **Processed Data Set**

Below is the dataset that we used to run on Weka:

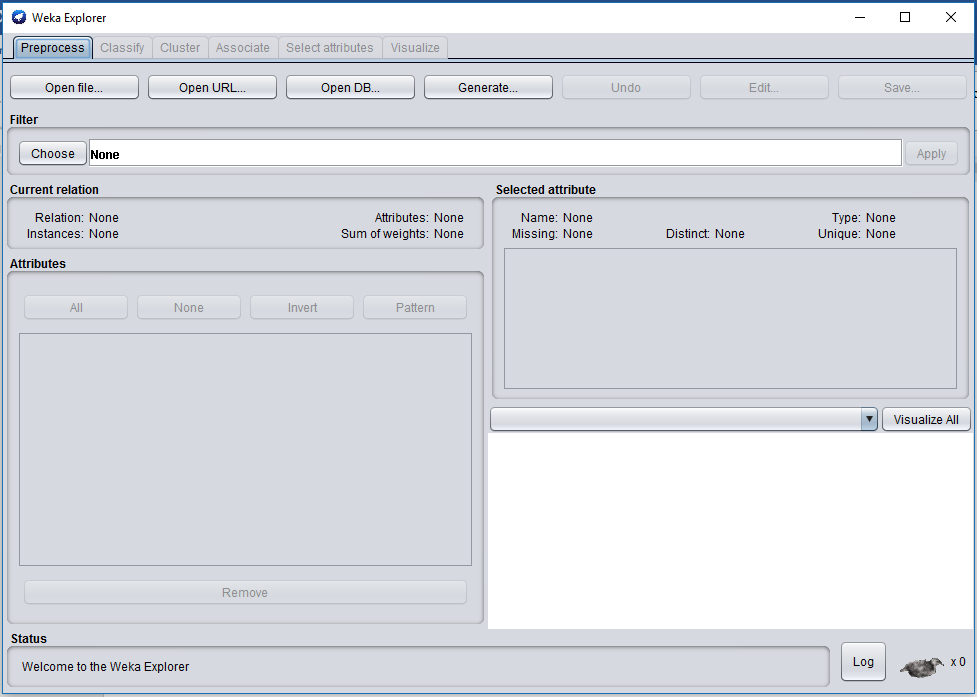


**Process**

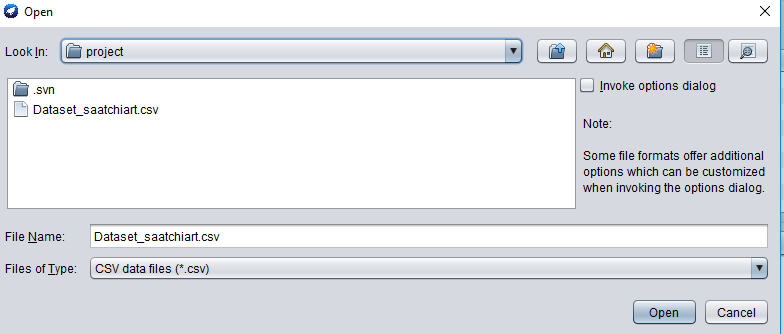
**Step 1:** Launch Weka Explorer

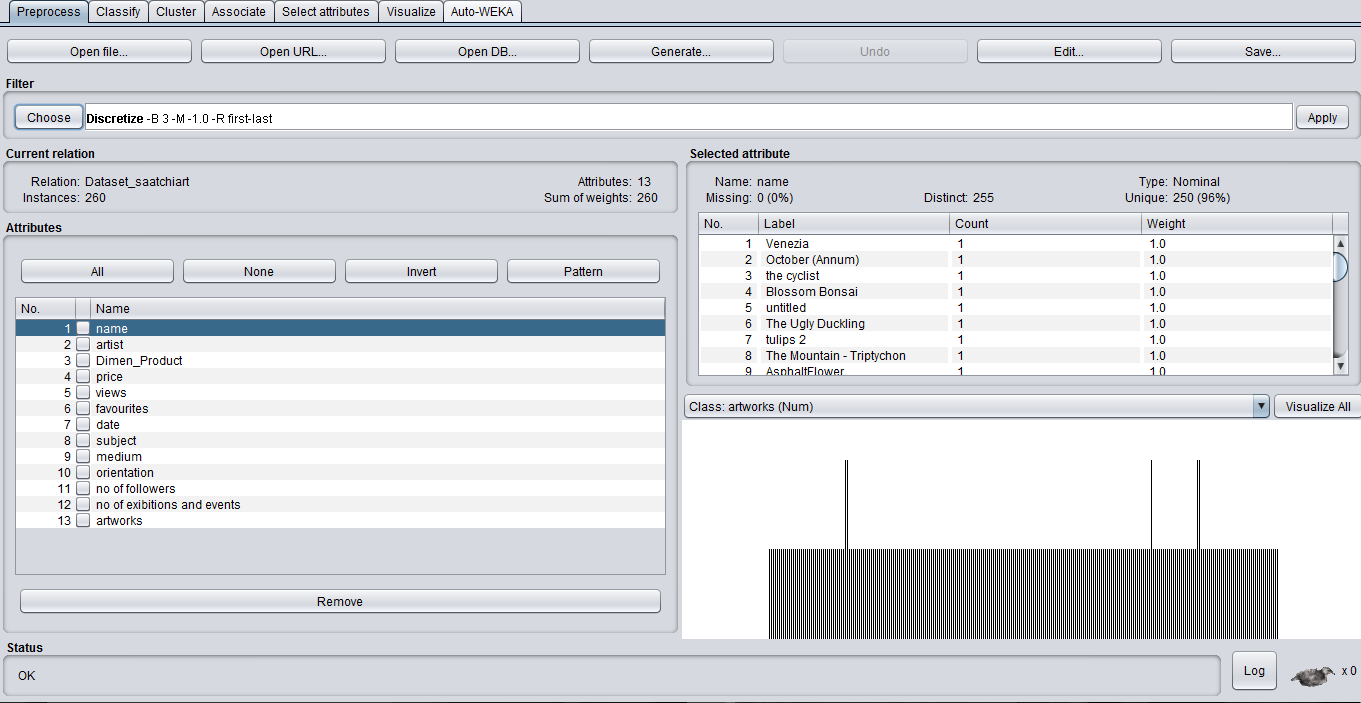


**Step 2:** Explorer Window



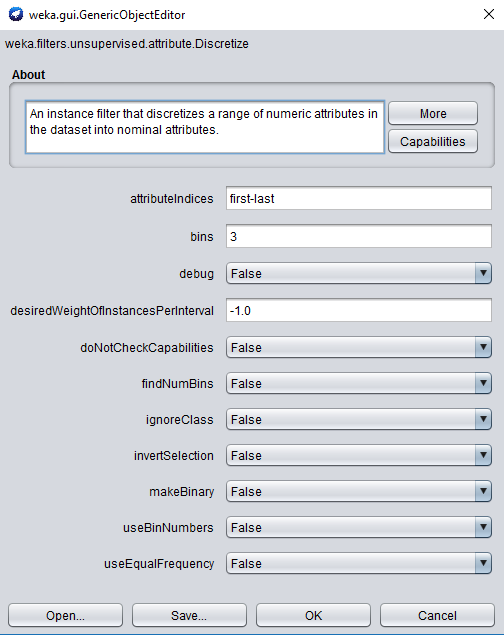
**Step 3:** Click Open File

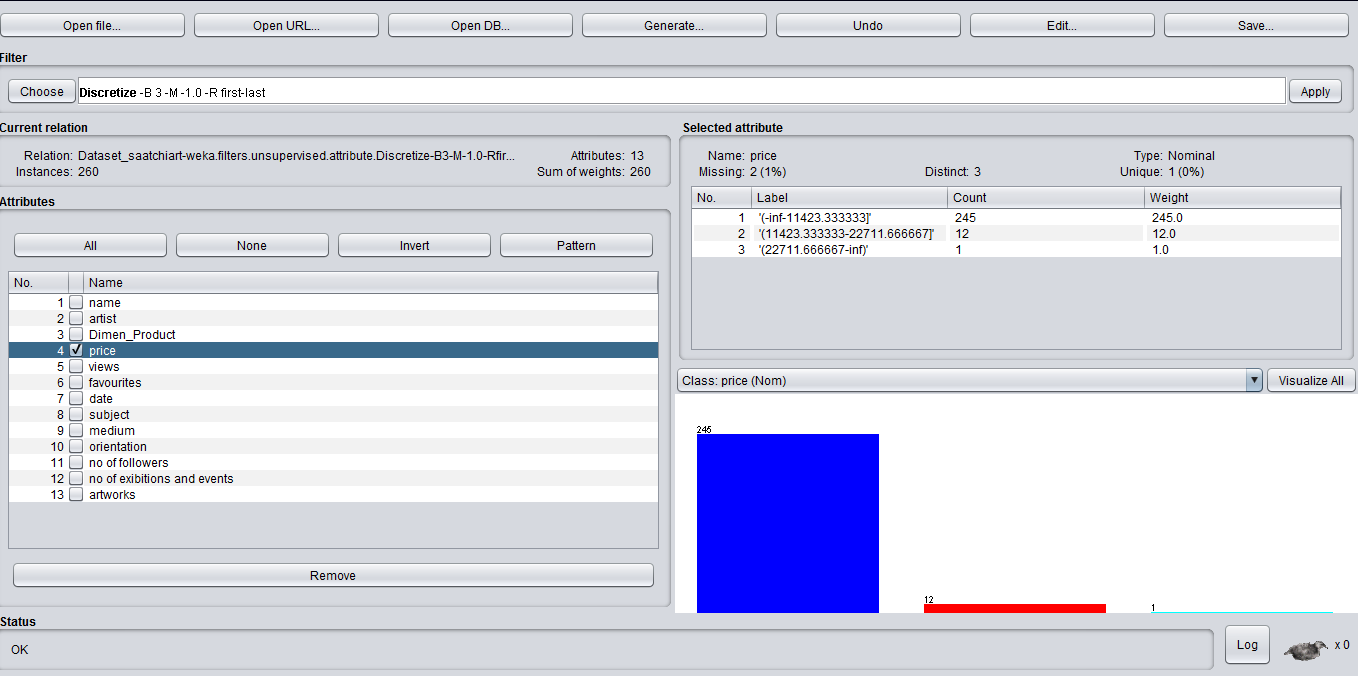
**Step 4:** Data updated in Weka



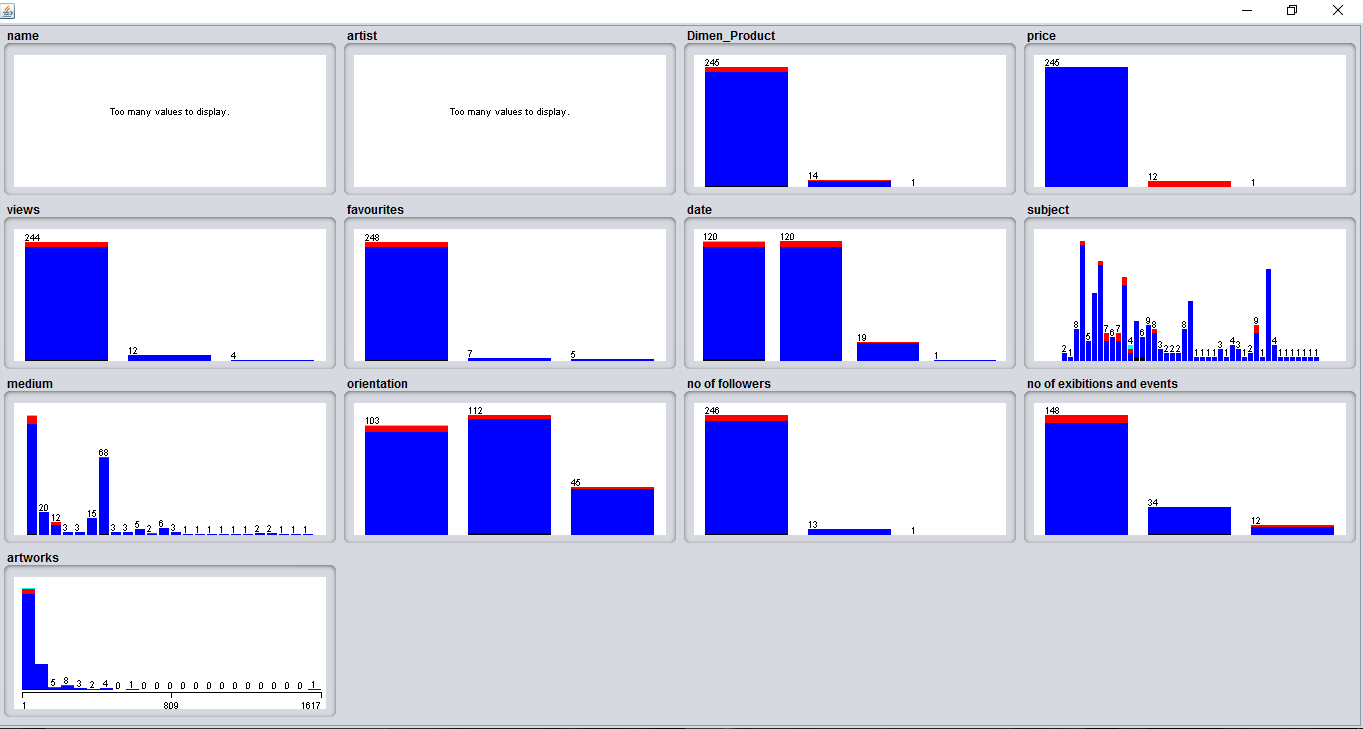
**Step 5:** Filter 🡪 Weka 🡪 Filters 🡪 unsupervised 🡪 attribute 🡪 Discretize

Click on Object editor and choose bins to 3 as we are discretizing with 3 intervals and Apply.





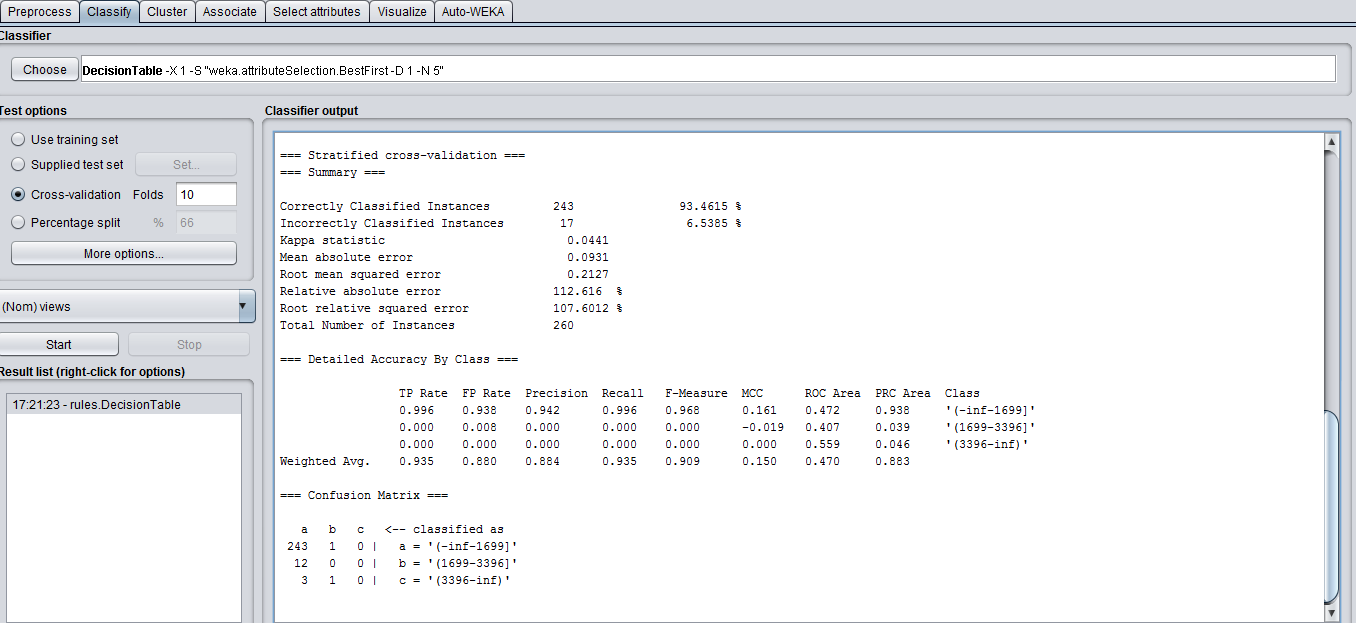
**Visualizing all the other Attributes**



**Step 5:** Go to Classify tab

Choose 🡪 weka 🡪 classifiers 🡪 Choose classifier of interest

Select Cross-Validation with default 10 folds in Test options



**Classification**

Classifiers in WEKA are the models for predicting nominal or numeric quantities. WEKA tool has around 52 different classification/ regression algorithms. For this study, the dataset is classified with the attribute “Views” and “Price” using the Decision Table[[4]](http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/DecisionTable.html) , Random Forest[[5]](http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomForest.html) , J48[[6]](http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/J48.html) and JRip[[7]](http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/JRip.html" \o "Weka documentation for JRip) classifier algorithms. Once the dataset is discretized and normalized, we employed trial and error method to determine the algorithm that produces high percentage of correctly classified instances. Other algorithms we tried include RandomTree , OneR , ZeroR. The decision tree induction uses the top-down recursive divide and conquer approach. JRip is the Repeated Incremental Pruning algorithm that is dedicated towards error reduction. This algorithm operates in different stages including the Building Stage with Grow Phase and Prune Phase and the Optimization Stage. This algorithm is said to implement the propositional rule learner which is an optimized version of IREP. This algorithmic setting included different values for cross validation folds and percentage split.

## **Classifier Outputs**

1. **Decision Table on Price**

=== Run information ===

Scheme: weka.classifiers.rules.DecisionTable -X 1 -S "weka.attributeSelection.BestFirst -D 1 -N 5"

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 260

Number of Rules : 6

Non matches covered by Majority class.

Best first.

Start set: no attributes

Search direction: forward

Stale search after 5 node expansions

Total number of subsets evaluated: 77

Merit of best subset found: 94.231

Evaluation (for feature selection): CV (leave one out)

Feature set: 3,11,5

Time taken to build model: 0.35 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 243 93.4615 %

Incorrectly Classified Instances 17 6.5385 %

Kappa statistic 0.0441

Mean absolute error 0.0931

Root mean squared error 0.2127

Relative absolute error 112.616 %

Root relative squared error 107.6012 %

Total Number of Instances 260

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.938 0.942 0.996 0.968 0.161 0.472 0.938 '(-inf-1699]'

0.000 0.008 0.000 0.000 0.000 -0.019 0.407 0.039 '(1699-3396]'

0.000 0.000 0.000 0.000 0.000 0.000 0.559 0.046 '(3396-inf)'

Weighted Avg. 0.935 0.880 0.884 0.935 0.909 0.150 0.470 0.883

=== Confusion Matrix ===

a b c <-- classified as

243 1 0 | a = '(-inf-1699]'

12 0 0 | b = '(1699-3396]'

3 1 0 | c = '(3396-inf)'

### **b. Decision Table on no. of Views**

=== Run information ===

Scheme: weka.classifiers.rules.DecisionTable -X 1 -S "weka.attributeSelection.BestFirst -D 1 -N 5"

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 260

Number of Rules : 6

Non matches covered by Majority class.

Best first.

Start set: no attributes

Search direction: forward

Stale search after 5 node expansions

Total number of subsets evaluated: 77

Merit of best subset found: 94.231

Evaluation (for feature selection): CV (leave one out)

Feature set: 3,11,5

Time taken to build model: 0.08 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 243 93.4615 %

Incorrectly Classified Instances 17 6.5385 %

Kappa statistic 0.0441

Mean absolute error 0.0931

Root mean squared error 0.2127

Relative absolute error 112.616 %

Root relative squared error 107.6012 %

Total Number of Instances 260

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.938 0.942 0.996 0.968 0.161 0.472 0.938 '(-inf-1699]'

0.000 0.008 0.000 0.000 0.000 -0.019 0.407 0.039 '(1699-3396]'

0.000 0.000 0.000 0.000 0.000 0.000 0.559 0.046 '(3396-inf)'

Weighted Avg. 0.935 0.880 0.884 0.935 0.909 0.150 0.470 0.883

=== Confusion Matrix ===

a b c <-- classified as

243 1 0 | a = '(-inf-1699]'

12 0 0 | b = '(1699-3396]'

3 1 0 | c = '(3396-inf)'

### **c.** **Random Forest on Views**

Random Forest Algorithm:

When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This out-of-bag data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance. After each tree is built, all the data are run down the tree, and proximities are computed for each pair of cases. If two cases occupy the same terminal node, their proximity is increased by one. At the end of the run, the proximities are normalized by dividing by the number of trees. Proximities are used in replacing missing data, locating outliers, and producing illuminating low-dimensional views of the data.

=== Run information ===

Scheme: weka.classifiers.rules.DecisionTable -X 1 -S "weka.attributeSelection.BestFirst -D 1 -N 5"

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

Decision Table:

Number of training instances: 260

Number of Rules : 6

Non matches covered by Majority class.

Best first.

Start set: no attributes

Search direction: forward

Stale search after 5 node expansions

Total number of subsets evaluated: 77

Merit of best subset found: 94.231

Evaluation (for feature selection): CV (leave one out)

Feature set: 3,11,5

Time taken to build model: 0.09 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 243 93.4615 %

Incorrectly Classified Instances 17 6.5385 %

Kappa statistic 0.0441

Mean absolute error 0.0931

Root mean squared error 0.2127

Relative absolute error 112.616 %

Root relative squared error 107.6012 %

Total Number of Instances 260

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.938 0.942 0.996 0.968 0.161 0.472 0.938 '(-inf-1699]'

0.000 0.008 0.000 0.000 0.000 -0.019 0.407 0.039 '(1699-3396]'

0.000 0.000 0.000 0.000 0.000 0.000 0.559 0.046 '(3396-inf)'

Weighted Avg. 0.935 0.880 0.884 0.935 0.909 0.150 0.470 0.883

=== Confusion Matrix ===

a b c <-- classified as

243 1 0 | a = '(-inf-1699]'

12 0 0 | b = '(1699-3396]'

3 1 0 | c = '(3396-inf)'

### **d. Random Forest on Price**

=== Run information ===

Scheme: weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner

weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities

Time taken to build model: 0.06 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 245 94.9612 %

Incorrectly Classified Instances 13 5.0388 %

Kappa statistic 0

Mean absolute error 0.0598

Root mean squared error 0.1632

Relative absolute error 86.7307 %

Root relative squared error 91.1083 %

Total Number of Instances 258

Ignored Class Unknown Instances 2

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 1.000 0.950 1.000 0.974 0.000 0.741 0.972 '(-inf-11423.333333]'

0.000 0.000 0.000 0.000 0.000 0.000 0.740 0.369 '(11423.333333-22711.666667]'

0.000 0.000 0.000 0.000 0.000 0.000 0.048 0.004 '(22711.666667-inf)'

Weighted Avg. 0.950 0.950 0.902 0.950 0.925 0.000 0.738 0.941

=== Confusion Matrix ===

a b c <-- classified as

245 0 0 | a = '(-inf-11423.333333]'

12 0 0 | b = '(11423.333333-22711.666667]'

1 0 0 | c = '(22711.666667-inf)'

### **e. J48 on Price**

J48 Decision Trees Algorithm:

To classify a new item, it first needs to create a decision tree based on the attribute values of the available training data. So, whenever it encounters a set of items (training set) it identifies the attribute that discriminates the various instances most clearly. This feature that can tell us most about the data instances so that we can classify them the best is said to have the highest information gain. Now, among the possible values of this feature, if there is any value for which there is no ambiguity, that is, for which the data instances falling within its category have the same value for the target variable, then we terminate that branch and assign to it the target value that we have obtained. For the other cases, we then look for another attribute that gives us the highest information gain. Hence, we continue in this manner until we either get a clear decision of what combination of attributes gives us a target value, or we run out of attributes. If we run out of attributes, or if we cannot get an unambiguous result from the available information, we assign this branch a target value that most the items under this branch possess.

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree

------------------

: '(-inf-11423.333333]' (258.0/13.0)

Number of Leaves : 1

Size of the tree : 1

Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 245 94.9612 %

Incorrectly Classified Instances 13 5.0388 %

Kappa statistic 0

Mean absolute error 0.0641

Root mean squared error 0.1791

Relative absolute error 92.962 %

Root relative squared error 99.9531 %

Total Number of Instances 258

Ignored Class Unknown Instances 2

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 1.000 0.950 1.000 0.974 0.000 0.412 0.933 '(-inf-11423.333333]'

0.000 0.000 0.000 0.000 0.000 0.000 0.430 0.041 '(11423.333333-22711.666667]'

0.000 0.000 0.000 0.000 0.000 0.000 0.048 0.004 '(22711.666667-inf)'

Weighted Avg. 0.950 0.950 0.902 0.950 0.925 0.000 0.411 0.888

=== Confusion Matrix ===

a b c <-- classified as

245 0 0 | a = '(-inf-11423.333333]'

12 0 0 | b = '(11423.333333-22711.666667]'

1 0 0 | c = '(22711.666667-inf)'

### **f. J48 on no. of Views**

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree

------------------

favourites = '(-inf-40]': '(-inf-1699]' (248.0/7.0)

favourites = '(40-80]'

| subject = Bike: '(1699-3396]' (0.0)

| subject = Bike : '(1699-3396]' (0.0)

| subject = Botanic: '(1699-3396]' (0.0)

| subject = Nature: '(-inf-1699]' (2.0/1.0)

| subject = Business: '(1699-3396]' (0.0)

| subject = Culture: '(1699-3396]' (0.0)

| subject = Beach: '(1699-3396]' (0.0)

| subject = Science/Technology: '(3396-inf)' (1.0)

| subject = Architecture: '(-inf-1699]' (1.0)

| subject = Boat: '(1699-3396]' (0.0)

| subject = Celebrity: '(1699-3396]' (0.0)

| subject = Fashion: '(1699-3396]' (0.0)

| subject = Floral: '(1699-3396]' (0.0)

| subject = Fish: '(1699-3396]' (0.0)

| subject = Love: '(1699-3396]' (0.0)

| subject = Graffiti: '(1699-3396]' (0.0)

| subject = Body: '(1699-3396]' (0.0)

| subject = Recycle: '(1699-3396]' (0.0)

| subject = Travel: '(1699-3396]' (0.0)

| subject = Kids: '(1699-3396]' (0.0)

| subject = Abstract: '(1699-3396]' (0.0)

| subject = Women: '(1699-3396]' (2.0)

| subject = Sport: '(1699-3396]' (0.0)

| subject = Religion: '(1699-3396]' (0.0)

| subject = Society: '(1699-3396]' (0.0)

| subject = Interiors: '(1699-3396]' (0.0)

| subject = Animal: '(1699-3396]' (0.0)

| subject = Man: '(1699-3396]' (0.0)

| subject = Men: '(1699-3396]' (0.0)

| subject = Children: '(1699-3396]' (0.0)

| subject = Wall: '(1699-3396]' (0.0)

| subject = Cartoon: '(1699-3396]' (0.0)

| subject = Technology: '(1699-3396]' (0.0)

| subject = Rural Life: '(1699-3396]' (0.0)

| subject = Automobile: '(1699-3396]' (0.0)

| subject = Calligraphy: '(1699-3396]' (0.0)

| subject = Aerial: '(1699-3396]' (0.0)

| subject = People: '(1699-3396]' (1.0)

| subject = Cities: '(1699-3396]' (0.0)

| subject = Nude: '(1699-3396]' (0.0)

| subject = Aeroplane: '(1699-3396]' (0.0)

| subject = Garden: '(1699-3396]' (0.0)

| subject = Fantasy: '(1699-3396]' (0.0)

favourites = '(80-inf)': '(1699-3396]' (5.0/2.0)

Number of Leaves : 45

Size of the tree : 47

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 244 93.8462 %

Incorrectly Classified Instances 16 6.1538 %

Kappa statistic 0.3342

Mean absolute error 0.0615

Root mean squared error 0.1884

Relative absolute error 74.4455 %

Root relative squared error 95.3071 %

Total Number of Instances 260

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.988 0.625 0.960 0.988 0.974 0.477 0.663 0.954 '(-inf-1699]'

0.250 0.024 0.333 0.250 0.286 0.259 0.626 0.153 '(1699-3396]'

0.000 0.000 0.000 0.000 0.000 0.000 0.282 0.047 '(3396-inf)'

Weighted Avg. 0.938 0.588 0.916 0.938 0.927 0.459 0.655 0.903

=== Confusion Matrix ===

a b c <-- classified as

241 3 0 | a = '(-inf-1699]'

9 3 0 | b = '(1699-3396]'

1 3 0 | c = '(3396-inf)'

### **g. JRip on no. of Views**

JRip Algorithm:

JRip (RIPPER) is one of the basic and most popular algorithms. Classes are examined in increasing size and an initial set of rules for the class is generated using incremental reduced error JRip (RIPPER) proceeds by treating all the examples of a judgment in the training data as a class, and finding a set of rules that cover all the members of that class. Thereafter it proceeds to the next class and does the same, repeating this until all classes have been covered.

=== Run information ===

Scheme: weka.classifiers.rules.JRip -F 3 -N 2.0 -O 2 -S 1

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

JRIP rules:

===========

(artworks >= 60) and (favourites = '(80-inf)') => views='(1699-3396]' (4.0/1.0)

=> views='(-inf-1699]' (256.0/12.0)

Number of Rules : 2

Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 243 93.4615 %

Incorrectly Classified Instances 17 6.5385 %

Kappa statistic -0.0045

Mean absolute error 0.0735

Root mean squared error 0.2006

Relative absolute error 88.9076 %

Root relative squared error 101.4597 %

Total Number of Instances 260

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 1.000 0.938 0.996 0.966 -0.016 0.478 0.933 '(-inf-1699]'

0.000 0.000 0.000 0.000 0.000 0.000 0.476 0.044 '(1699-3396]'

0.000 0.004 0.000 0.000 0.000 -0.008 0.244 0.013 '(3396-inf)'

Weighted Avg. 0.935 0.939 0.880 0.935 0.907 -0.015 0.474 0.878

=== Confusion Matrix ===

a b c <-- classified as

243 0 1 | a = '(-inf-1699]'

12 0 0 | b = '(1699-3396]'

4 0 0 | c = '(3396-inf)'

### **h. JRip on Price**

=== Run information ===

Scheme: weka.classifiers.rules.JRip -F 3 -N 2.0 -O 2 -S 1

Relation: Dataset\_saatchiart-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-Rfirst-last

Instances: 260

Attributes: 13

name

artist

Dimen\_Product

price

views

favourites

date

subject

medium

orientation

no of followers

no of exibitions and events

artworks

Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

JRIP rules:

===========

(artist = Leo Kadele) => price='(11423.333333-22711.666667]' (2.0/0.0)

(name = David Bowie) => price='(11423.333333-22711.666667]' (2.0/0.0)

=> price='(-inf-11423.333333]' (254.0/9.0)

Number of Rules : 3

Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 243 94.186 %

Incorrectly Classified Instances 15 5.814 %

Kappa statistic -0.0131

Mean absolute error 0.0599

Root mean squared error 0.1857

Relative absolute error 86.8217 %

Root relative squared error 103.6323 %

Total Number of Instances 258

Ignored Class Unknown Instances 2

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.992 1.000 0.949 0.992 0.970 -0.020 0.497 0.944 '(-inf-11423.333333]'

0.000 0.008 0.000 0.000 0.000 -0.020 0.459 0.042 '(11423.333333-22711.666667]'

0.000 0.000 0.000 0.000 0.000 0.000 0.052 0.004 '(22711.666667-inf)'

Weighted Avg. 0.942 0.950 0.901 0.942 0.921 -0.020 0.493 0.898

=== Confusion Matrix ===

a b c <-- classified as

243 2 0 | a = '(-inf-11423.333333]'

12 0 0 | b = '(11423.333333-22711.666667]'

1. 0 0 | c = '(22711.666667-inf)'

**Interpretation of the Results**

* There are about 94 % correctly classified instances and 6% incorrectly classified instances.
* J48 Classifier gave maximum Correctly classified Correctly Classified Instances i.e 94.9612% for Price attribute
* J48 Classifier on Number of Views attributes gives 93.8462% correctly classifies instances
* JRip Classifier on Price attribute gives 94.186% correctly classifies instances
* JRip Classifier on Number of Views attribute gives 93.4615 % correctly classifies instances

**Conclusion**

Considering there are lot of classifiers provided by Weka we deduce that our data set built for fine art paintings yield on average 94.5% of correct classification after discretizing the data set into 3 bins taking price as decision attribute. We noticed that if we increase the no. of bins for discretization the percentage of current classification goes down.

**References**

[1] WikiPedia, Saatchi Art <https://en.wikipedia.org/wiki/Saatchi_Art>

[2] WikiPedia, Weka <https://en.wikipedia.org/wiki/Weka_(machine_learning)>

[3] Weka official web site <http://weka.sourceforge.net/>

[4] Weka Decision table documentation <http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/DecisionTable.html>

[5] Weka Random Forest documentation <http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomForest.html>

[6] Weka J48 Documentation <http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/J48.html>

[7] Weka JRip Documentation <http://weka.sourceforge.net/doc.dev/weka/classifiers/rules/JRip.html>