

Data warehouse & Data mining lab

Assignment 4b

Submitted by :
Shiv Kumar
2016UIT2563
Semester 8
IT2

Experiment 12 : Write a procedure for Clustering Customer data using Simple KMeans Algorithm.

1. Data Loading

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

Filter: Choose None [Apply] [Stop]

Current relation
Relation: customer
Instances: 10
Attributes: 4
Sum of weights: 10

Attributes
[All] [None] [Invert] [Pattern]

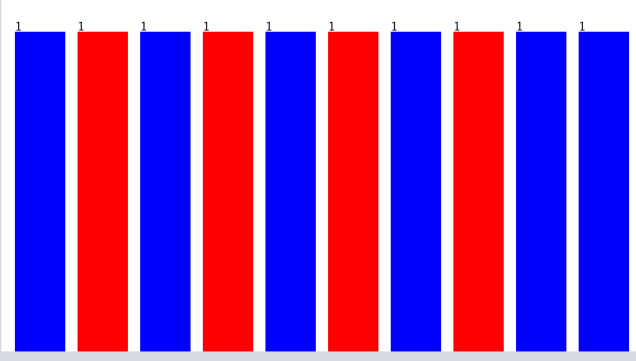
No.	Name
1	<input checked="" type="checkbox"/> name
2	<input type="checkbox"/> age
3	<input type="checkbox"/> income
4	<input type="checkbox"/> class

[Remove]

Selected attribute
Name: name
Missing: 0 (0%)
Distinct: 10
Type: Nominal
Unique: 10 (100%)

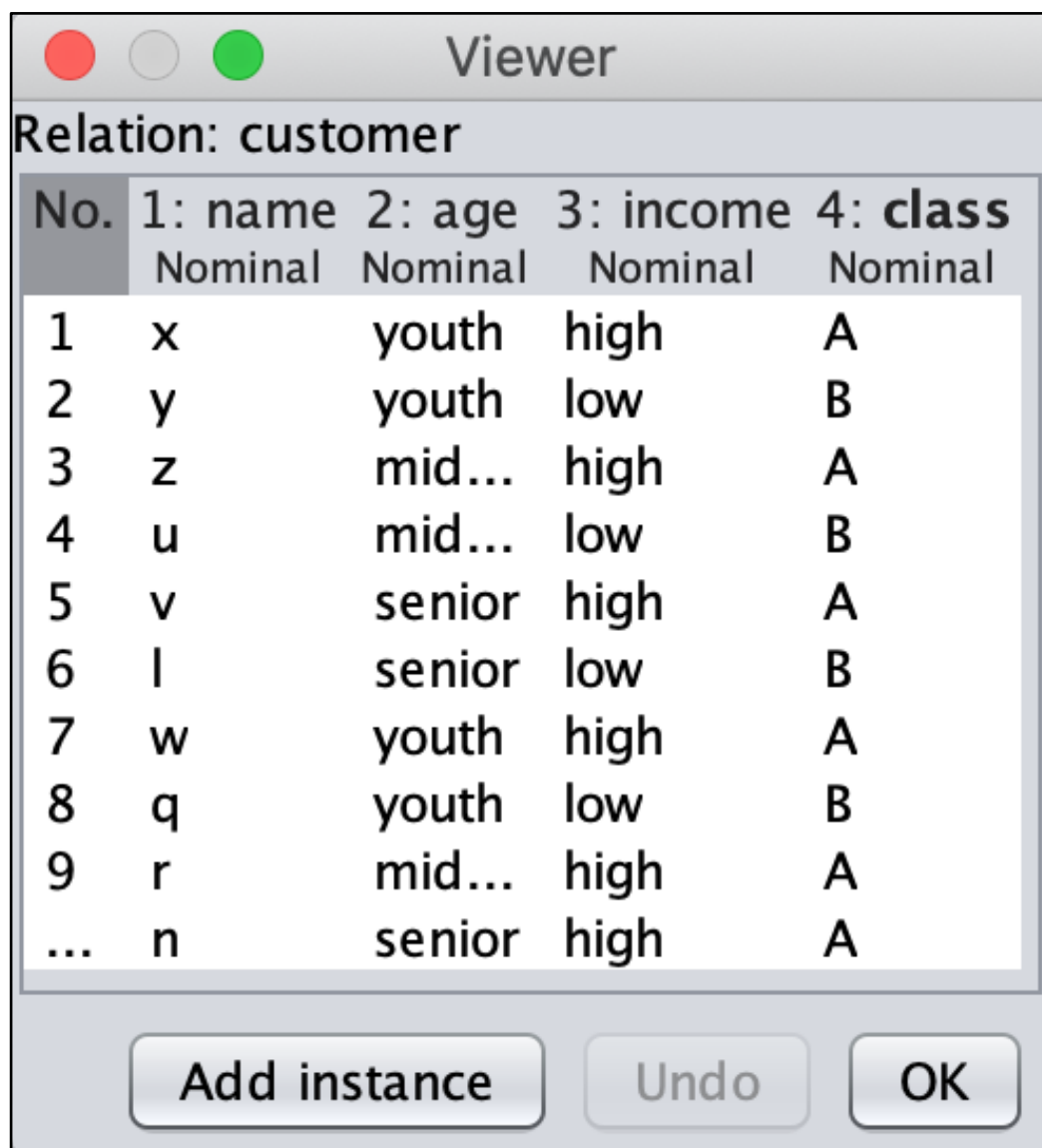
No.	Label	Count	Weight
1	x	1	1.0
2	y	1	1.0
3	z	1	1.0
4	u	1	1.0
5	v	1	1.0
6	l	1	1.0
7	w	1	1.0
8	q	1	1.0
9	r	1	1.0
10	n	1	1.0

Class: class (Nom) [Visualize All]



Status
OK [Log] x 0

2. Review of Loaded Data



The image shows a software window titled "Viewer" with a standard macOS-style title bar (red, yellow, and green buttons). Below the title bar, the text "Relation: customer" is displayed. The main area contains a table with five columns: "No.", "1: name", "2: age", "3: income", and "4: class". Each column has a data type listed below its header: "Nominal", "Nominal", "Nominal", and "Nominal" respectively. The table lists 10 instances, numbered 1 through 10 (the last row is truncated with "..."). The data is as follows:

No.	1: name	2: age	3: income	4: class
	Nominal	Nominal	Nominal	Nominal
1	x	youth	high	A
2	y	youth	low	B
3	z	mid...	high	A
4	u	mid...	low	B
5	v	senior	high	A
6	l	senior	low	B
7	w	youth	high	A
8	q	youth	low	B
9	r	mid...	high	A
...	n	senior	high	A

At the bottom of the window, there are three buttons: "Add instance", "Undo", and "OK".

3. Result after Simple KMeans Algorithm.

The screenshot shows the Weka Explorer interface with the 'Cluster' tab selected. The 'SimpleKMeans' algorithm is chosen with the following command: `-init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 50`. The 'Cluster mode' section has 'Use training set' selected and 'Store clusters for visualization' checked. The 'Result list' shows '01:05:15 - SimpleKMeans' as the selected result. The 'Cluster output' pane displays the following information:

```
=== Run information ===
Scheme:      weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 50
Relation:    customer
Instances:   10
Attributes:  4
              name
              age
              income
              class
Test mode:   evaluate on training data

=== Clustering model (full training set) ===

kMeans
=====
Number of iterations: 2
Within cluster sum of squared errors: 14.000000000000004

Initial starting points (random):
Cluster 0: u,middle,low,B
Cluster 1: w,youth,high,A

Missing values globally replaced with mean/mode

Final cluster centroids:
Attribute      Full Data      Cluster#
              (10.0)      0          1
              (10.0)      (4.0)      (6.0)
=====
name           x             y             x
age            youth        youth        youth
income         high         low          high
class          A            B            A
```

The 'Status' bar at the bottom shows 'OK' and a 'Log' button.

Final cluster centroids:			
Attribute	Full Data	Cluster#	
	(10.0)	0	1
		(4.0)	(6.0)
=====			
name	x	y	x
age	youth	youth	youth
income	high	low	high
class	A	B	A
Time taken to build model (full training data) : 0 seconds			
=== Model and evaluation on training set ===			
Clustered Instances			
0	4 (40%)		
1	6 (60%)		