



# Machine Learning and Weka

Tong Liu, 10/12/2018

Dip. Informatica, Mura Anteo Zamboni 7

**BOLOGNA BUSINESS SCHOOL**  
Alma Mater Studiorum Università di Bologna

# Outline

- Overview: machine learning (ML) in AI
- What is ML ?
- Supervised ML vs. Unsupervised ML
- Types of prediction problems (Supervised ML)
- Learning workflow (supervised & unsupervised)
- Classical ML methods vs. Deep Learning (DL)
- Get started with Weka
- Demo

# Overview: ML in AI

## Artificial Intelligence

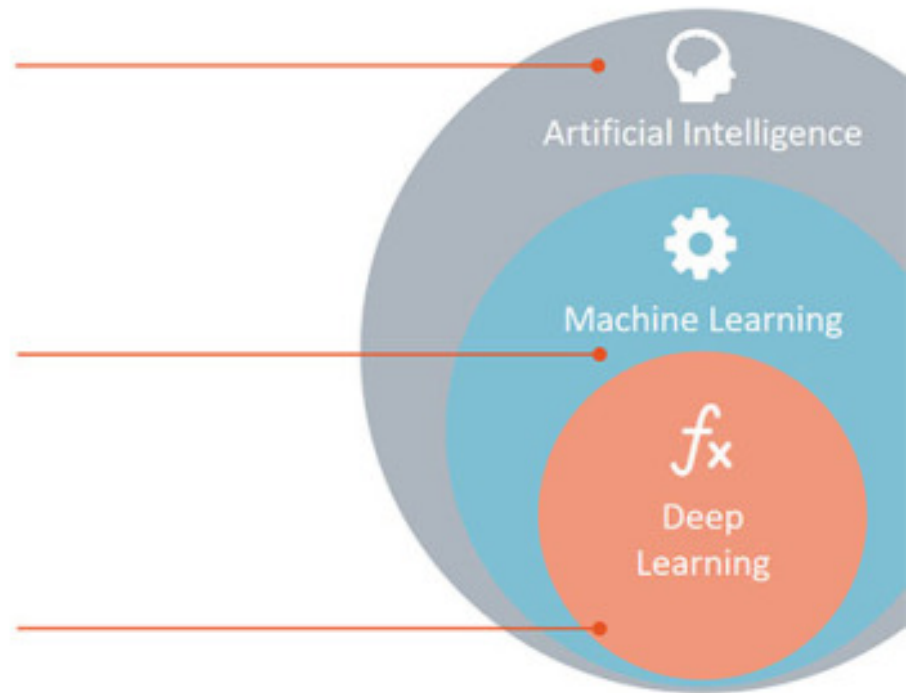
Any technique which enables computers to mimic human behavior.

## Machine Learning

Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

## Deep Learning

Subset of ML which make the computation of multi-layer neural networks feasible.



[Rapidminer]

# What is the ML ?

- Alan Turing (1912-1954)
- “what we want is a **machine** that can learn from *experience*.”



[The Imitation Game]

# Experience for machines ?



[The Imitation Game]

# Dataset - a kind of experience

- A dataset with attributes (a.k.a features) that can be processed by a computer.

Size of House	Lot Size (acre)	# of Bedrooms	# of Bathrooms	Price of House
950	2.5	2	1	\$127,325
1,535	1.5	2	2	\$156,570
1,605	2.25	3	1.5	\$158,895
1,905	2.5	2	1.5	\$200,025
2,057	2.25	3	2	\$230,384
2,227	2.75	3	2	\$233,835
3,150	1	4	2	\$261,420
3,620	3	4	3	\$433,500

House prices dataset

# Dataset - supervised vs. unsupervised

- Supervised ML: data are labelled and an interest is given - learn with a purpose
- Unsupervised ML: no specific interest is given - exploratory analysis

Size of House	Lot Size (acre)	# of Bedrooms	# of Bathrooms	Price of House
950	2.5	2	1	\$127,325
1,535	1.5	2	2	\$156,570
1,605	2.25	3	1.5	\$158,895
1,905	2.5	2	1.5	\$200,025
2,057	2.25	3	2	\$230,384
2,227	2.75	3	2	\$233,835
3,150	1	4	2	\$261,420
3,620	3	4	3	\$433,500

House prices dataset

# Types of Prediction problems (supervised learning)

- Regression: the output variable takes continuous values.
- Classification: the output variable takes class labels.

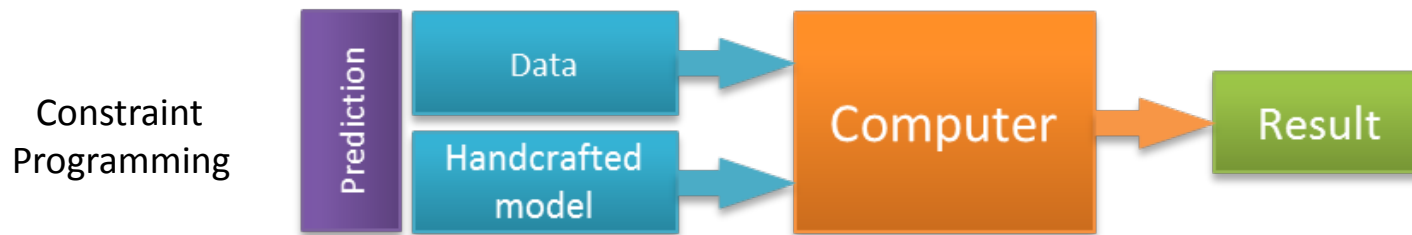
Size of House	Lot Size (acre)	# of Bedrooms	# of Bathrooms	Price of House
950	2.5	2	1	\$127,325
1,535	1.5	2	2	\$156,570
1,605	2.25	3	1.5	\$158,895
1,905	2.5	2	1.5	\$200,025
2,057	2.25	3	2	\$230,384
2,227	2.75	3	2	\$233,835
3,150	1	4	2	\$261,420
3,620	3	4	3	\$433,500

House prices dataset

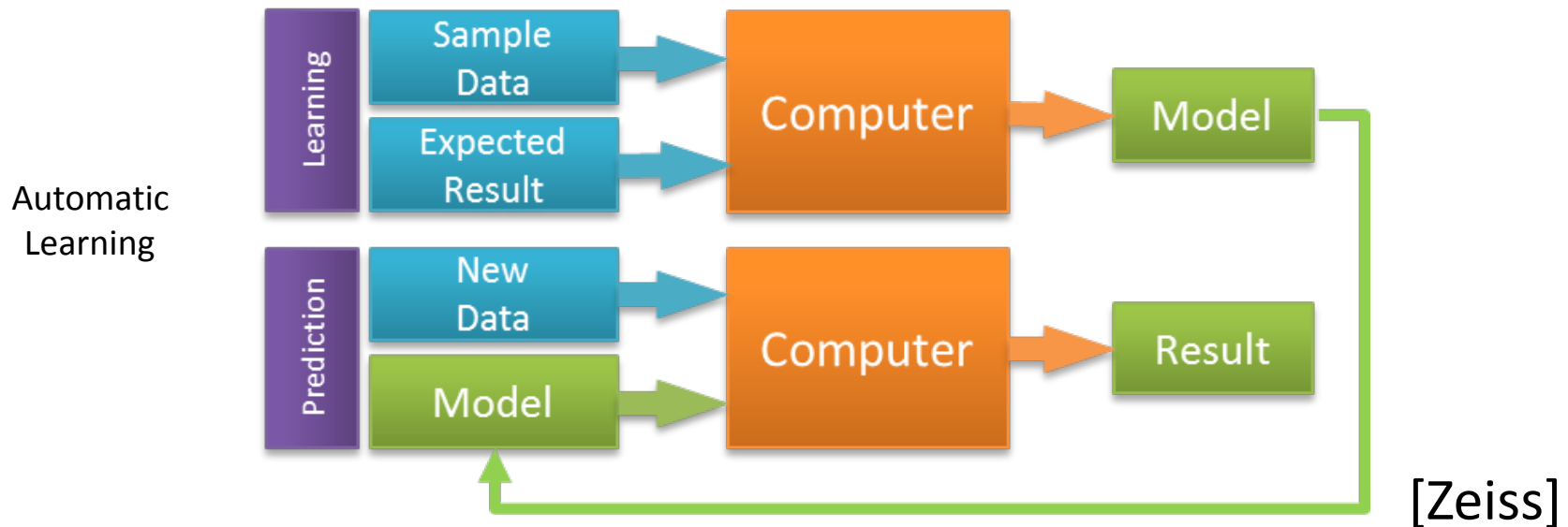


# How we work with dataset?

## Traditional modeling:

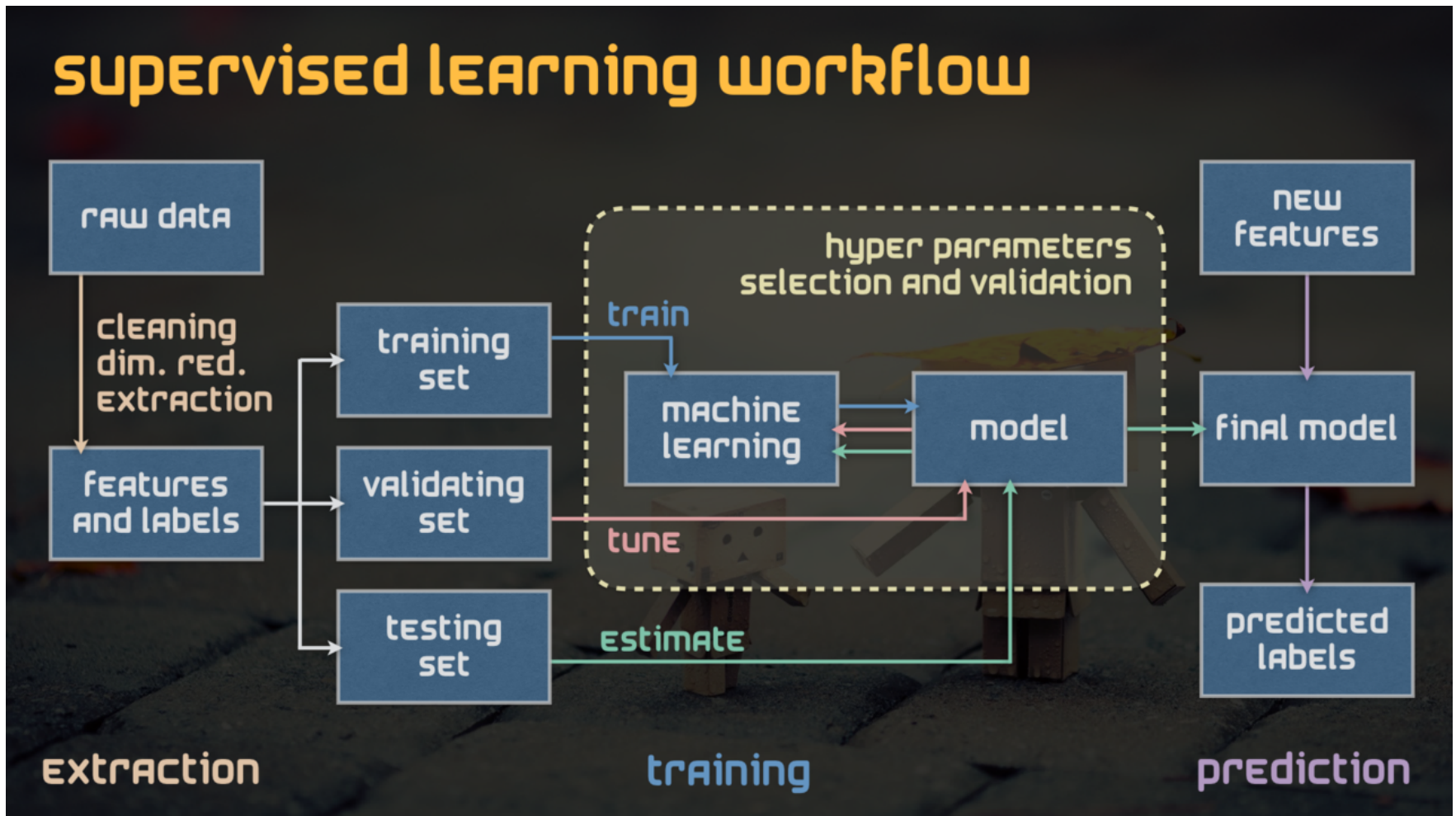


## Machine Learning:



[Zeiss]

# Supervised ML workflow



[BidMotion]

# Classical ML vs. Deep Learning

## CLASSIC MACHINE LEARNING

How do you engineer the best features?



### FEATURES

- ▶ Roundness of face;
- ▶ Distance between eyes;
- ▶ Nose width;
- ▶ Eye socket depth;
- ▶ Cheek bone structure;
- ▶ Jaw line length...

### CLASSIFIER ALGORITHMS

- ▶ SVM;
- ▶ Random Forest;
- ▶ Naïve Bayes;
- ▶ Decision Tree;
- ▶ Regression
- ▶ ...many more...

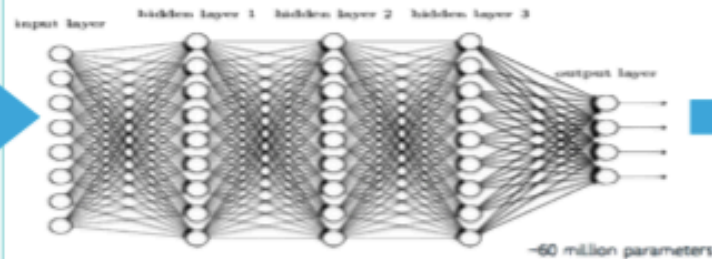
Jim Carrey

## DEEP LEARNING

How do you guide the model to find the best features?



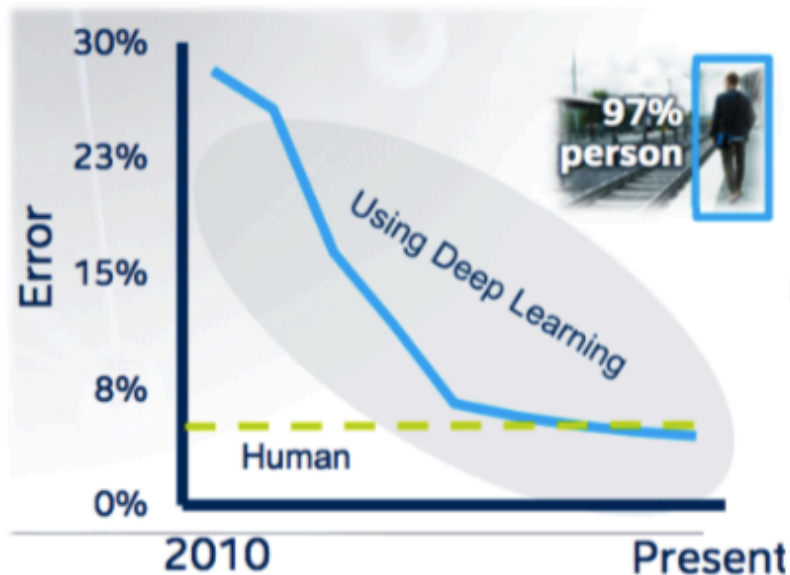
### DEEP NEURAL NETWORK



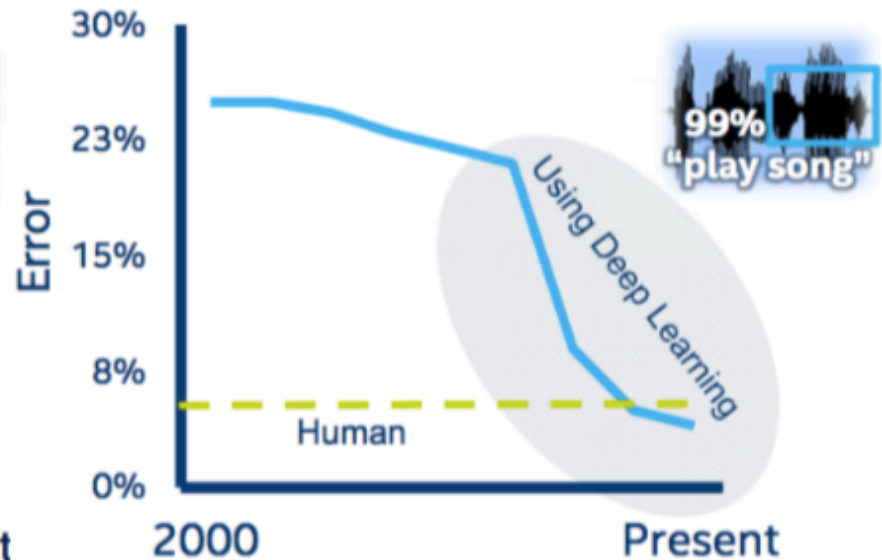
Jim Carrey

# Deep Learning breakthroughs

## IMAGE RECOGNITION



## SPEECH RECOGNITION

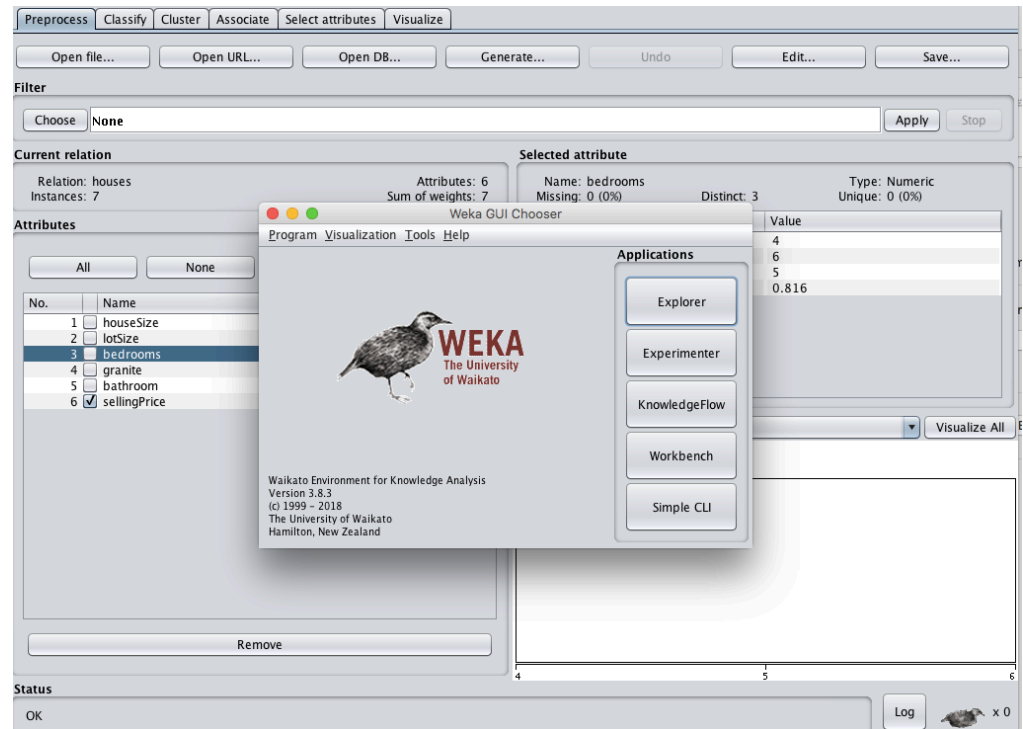


**MACHINES ABLE TO MEET OR EXCEED HUMAN IMAGE & SPEECH RECOGNITION (TO SOME EXTEND...)**

# A simple example of prediction

WEKA is developed by the University of Waikato (New Zealand) under the GNU General Public License (GPL).

It is written in the Java™ object-oriented programming language and provides a GUI for interacting with data files and producing visual results.



# Collected data

House size (square feet)	Lot size	Bedrooms	Granite	Upgraded bathroom?	Selling price
3529	9191	6	0	0	205000
3247	10061	5	1	1	224900
4032	10150	5	0	1	197900
2397	14156	4	1	0	189900
2200	9600	4	0	1	195000
3536	19994	6	1	1	325000
2983	9365	5	0	1	230000
3198	9669	5	1	1	????

# Collected data

Training Data

House size (square feet)	Lot size	Bedrooms	Granite	Upgraded bathroom?	Selling price
3529	9191	6	0	0	205000
3247	10061	5	1	1	224900
4032	10150	5	0	1	197900
2397	14156	4	1	0	189900
2200	9600	4	0	1	195000
3536	19994	6	1	1	325000
2983	9365	5	0	1	230000

3198	9669	5	1	1	????
------	------	---	---	---	------

Test Data

# Loading dataset

## ▶ Preprocess Tab

The screenshot shows the Weka Explorer application window. The 'Preprocess' tab is highlighted with a red box and a red arrow. The interface includes a menu bar with options like 'Preprocess', 'Classify', 'Cluster', 'Associate', 'Select attributes', and 'Visualize'. Below the menu bar are buttons for 'Open file...', 'Open URL...', 'Open DB...', 'Generate...', 'Undo', 'Edit...', and 'Save...'. The 'Filter' section shows 'None' selected. The 'Current relation' section displays 'Relation: houses' and 'Instances: 7'. The 'Attributes' section has buttons for 'All', 'None', 'Invert', and 'Pattern', and a list of attributes: 'houseSize', 'lotSize', 'bedrooms', 'granite', 'bathroom', and 'sellingPrice'. The 'Selected attribute' section shows 'Name: houseSize', 'Missing: 0 (0%)', 'Distinct: 7', and 'Type: Numeric'. A table of statistics for 'houseSize' is shown: Minimum (2200), Maximum (4032), Mean (3132), and StdDev (655.121). The 'Class' is set to 'sellingPrice (Num)'. The status bar at the bottom shows 'OK' and 'Log'.

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

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

Filter  
None Apply Stop

Current relation  
Relation: houses Attributes: 6  
Instances: 7 Sum of weights: 7

Attributes  
All None Invert Pattern

No.	Name
1	houseSize
2	lotSize
3	bedrooms
4	granite
5	bathroom
6	sellingPrice

Remove

Selected attribute  
Name: houseSize Type: Numeric  
Missing: 0 (0%) Distinct: 7 Unique: 7 (100%)

Statistic	Value
Minimum	2200
Maximum	4032
Mean	3132
StdDev	655.121

Class: sellingPrice (Num) Visualize All

Status  
OK Log x 0



# Loading dataset

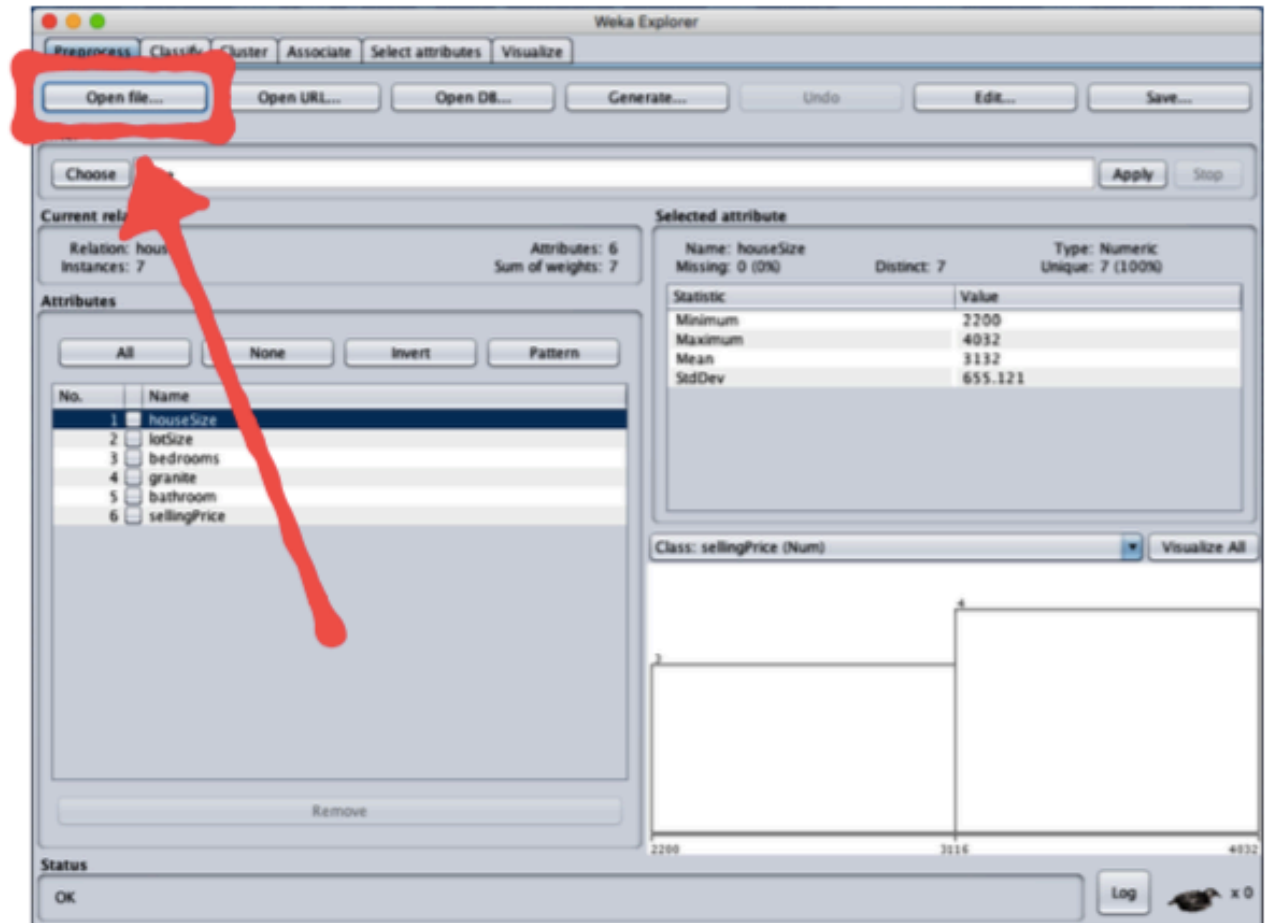
▶ Preprocess Tab

▶ Open File:

▶ CSV format

▶ XLS format

▶ ARFF format



# Loading dataset

- ▶ **Numerical variables**  
data with value representable with numbers
- ▶ **Visualize All**  
shows all graphics at once

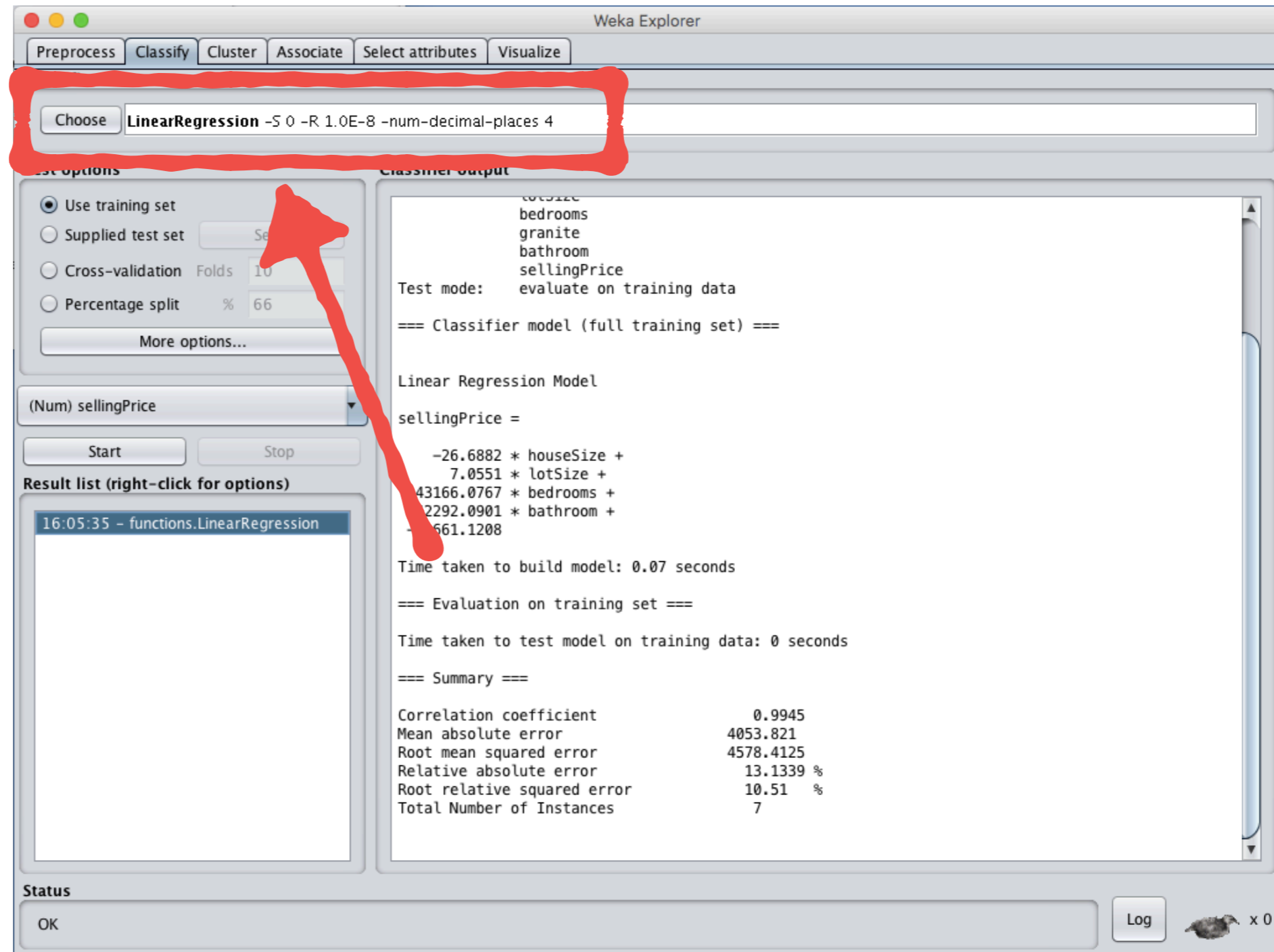
The screenshot shows the Weka Explorer window with the 'Visualize' tab selected. The interface includes a menu bar with 'Preprocess', 'Classify', 'Cluster', 'Associate', 'Select attributes', and 'Visualize'. Below the menu bar are buttons for 'Open file...', 'Open URL...', 'Open DB...', 'Generate...', 'Undo', 'Edit...', and 'Save...'. The 'Filter' section has a 'Choose' dropdown set to 'None' and 'Apply' and 'Stop' buttons. The 'Current relation' section shows 'Relation: houses' and 'Instances: 7'. The 'Attributes' section has buttons for 'All', 'None', 'Invert', and 'Pattern'. A list of attributes is shown, with 'houseSize' selected and highlighted by a red box. The 'Selected attribute' section shows 'Name: houseSize', 'Missing: 0 (0%)', 'Distinct: 7', and 'Type: Numeric'. A table of statistics is displayed: 

Statistic	Value
Minimum	2200
Maximum	4032
Mean	3132
StdDev	5.121

. The 'Class: sellingPrice (Num)' section has a 'Visualize All' button highlighted by a red box. The status bar at the bottom shows 'OK' and 'Log' buttons.

# CLASSIFY THE DATA WITH WEKA LINEAR REGRESSION MODEL

## ► Classify Tab



The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The 'Choose' dropdown is set to 'LinearRegression' with parameters '-S 0 -R 1.0E-8 -num-decimal-places 4'. The 'Use training set' option is selected. The 'Start' button has been clicked, and the classifier output is displayed in the 'Classifier output' pane. A red arrow points from the 'Start' button to the output text.

**Classifier output**

```
costSize
bedrooms
granite
bathroom
sellingPrice
Test mode: evaluate on training data
=== Classifier model (full training set) ===

Linear Regression Model
sellingPrice =
-26.6882 * houseSize +
7.0551 * lotSize +
43166.0767 * bedrooms +
2292.0901 * bathroom +
-561.1208

Time taken to build model: 0.07 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

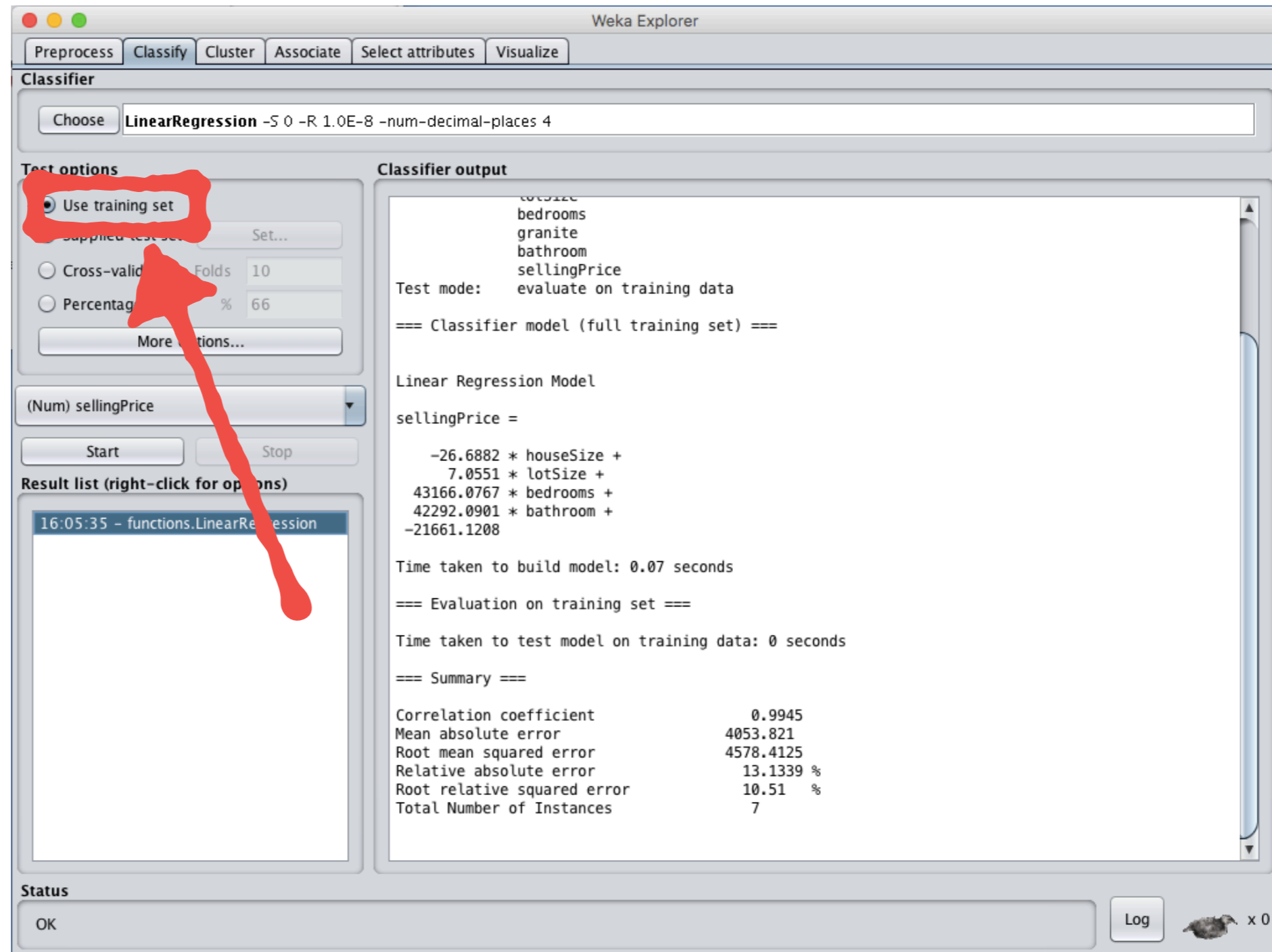
=== Summary ===

Correlation coefficient          0.9945
Mean absolute error             4053.821
Root mean squared error         4578.4125
Relative absolute error         13.1339 %
Root relative squared error     10.51 %
Total Number of Instances       7
```



# CLASSIFY THE DATA WITH WEKA LINEAR REGRESSION MODEL

- ▶ Classify Tab
- ▶ Use training set



Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4

Test options

Use training set

Cross-validation Folds 10

Percentage % 66

More options...

(Num) sellingPrice

Start Stop

Result list (right-click for options)

16:05:35 - functions.LinearRegression

Classifier output

```
lotSize
bedrooms
granite
bathroom
sellingPrice
Test mode: evaluate on training data
=== Classifier model (full training set) ===

Linear Regression Model
sellingPrice =
  -26.6882 * houseSize +
    7.0551 * lotSize +
  43166.0767 * bedrooms +
  42292.0901 * bathroom +
 -21661.1208

Time taken to build model: 0.07 seconds
=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds
=== Summary ===

Correlation coefficient          0.9945
Mean absolute error             4053.821
Root mean squared error        4578.4125
Relative absolute error        13.1339 %
Root relative squared error    10.51 %
Total Number of Instances      7
```

Status

OK

Log x 0



# CLASSIFY THE DATA WITH WEKA LINEAR REGRESSION MODEL

- ▶ Classify Tab;
- ▶ Use training set;
- ▶ Class = sellingPrice;

The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The 'Classifier' dropdown is set to 'LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4'. The 'Test options' section has 'Use training set' selected. The 'Class' dropdown is set to '(Num) sellingPrice'. The 'Classifier output' pane displays the following text:

```
costSize  
bedrooms  
granite  
bathroom  
sellingPrice  
Test mode: evaluate on training data  
=== Classifier model (full training set) ===  
Linear Regression Model  
sellingPrice =  
-26.6882 * houseSize +  
7.0551 * lotSize +  
43166.0767 * bedrooms +  
42292.0901 * bathroom +  
-21661.1208  
Time taken to build model: 0.07 seconds  
=== Evaluation on training set ===  
Time taken to test model on training data: 0 seconds  
=== Summary ===  
Correlation coefficient          0.9945  
Mean absolute error            4053.821  
Root mean squared error        4578.4125  
Relative absolute error         13.1339 %  
Root relative squared error     10.51 %  
Total Number of Instances      7
```

The 'Result list' shows a single entry: '16:05:35 - functions.LinearRegression'. The 'Status' bar at the bottom indicates 'OK'.



# CLASSIFY THE DATA WITH WEKA LINEAR REGRESSION MODEL

- ▶ Classify Tab;
- ▶ Use training set;
- ▶ Class = sellingPrice;
- ▶ Start building the model

The screenshot shows the Weka Explorer interface with the 'Classify' tab selected. The classifier is set to 'LinearRegression' with parameters: `-S 0 -R 1.0E-8 -num-decimal-places 4`. The test options are set to 'Use training set'. The class is set to '(Num) sellingPrice'. The 'Start' button is highlighted with a red box. The classifier output shows the following regression equation:

$$\text{sellingPrice} = -26.6882 * \text{houseSize} + 7.0551 * \text{lotSize} + 43166.0767 * \text{bedrooms} + 42292.0901 * \text{bathroom} - 21661.1208$$

The output also includes the following summary statistics:

=== Summary ===	
Correlation coefficient	0.9945
Mean absolute error	4053.821
Root mean squared error	4578.4125
Relative absolute error	13.1339 %
Root relative squared error	10.51 %
Total Number of Instances	7



# CLASSIFY THE DATA WITH WEKA LINEAR REGRESSION MODEL

- ▶ Classify Tab;
- ▶ Use training set;
- ▶ Class = sellingPrice;
- ▶ Start building the model

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose Linear Regression -S 0 -R 1.0E-8 -num-decimal-places 4

Test options

Use training set  
 Supplied test set  
 Cross-validation Folds 10  
 Percentage split % 66  
More options...

Classifier output

Test mode: evaluate on training data

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

Linear Regression Model

-26.6882 \* houseSize +  
7.0551 \* lotSize +  
43166.0767 \* bedrooms +  
42292.0901 \* bathroom +  
-21661.1208

Time taken to build model: 7 seconds

Correlation coefficient 0.9945  
Mean absolute error 4053.821  
Root mean squared error 4578.4125  
Relative absolute error 13.1339 %  
Root relative squared error 10.51 %  
Total Number of Instances 7

Result list (right-click for options)

16:05:35 - functions.LinearRegression

Status

OK

Log x 0

**Prediction formulae**

# FINAL PREDICTION

$$\begin{aligned} \text{sellingPrice} &= \\ &- 26,68 * [\text{houseSize} = 3198] \\ &+ 7,05 * [\text{lotSize} = 9669] \\ &+ 43.166,07 * [\text{bedrooms} = 5] \\ &+ 42.292,09 * [\text{bathroom} = 1] \\ &- 21.661,12 = \mathbf{219.328,25} \end{aligned}$$



# ANOTHER EXAMPLE OF CLASSIFICATION

## CAR DEALERSHIP

The dealership is **starting a promotional campaign**, whereby it is **trying to push a two-year extended warranty** to its past customers.

The dealership **has done this before** and has gathered **4,500 data points** from **past** sales of extended warranties.

**The attributes in the data set are:**

- ▶ Income bracket [0=\$0-\$30k, 1=\$31k-\$40k, 2=\$41k-\$60k, 3=\$61k-\$75k, 4=\$76k-\$100k, 5=\$101k-\$150k, 6=\$151k-\$500k, 7=\$501k+]
- ▶ Year/month first car bought
- ▶ Year/month most recent car bought
- ▶ **Whether they responded or not** to the extended warranty offer in the past



# PREPROCESS THE DATA WITH WEKA LOAD THE DATASET FOR TRAINING

- ▶ **Nominal variables** – *labelled data*

The screenshot shows the Weka Explorer interface. The 'Current relation' is 'bmwreponses' with 1500 instances and 4 attributes. The 'Selected attribute' is 'IncomeBracket', which is a nominal variable with 8 distinct values. The 'Class' is set to 'responded (Nom)'. A red arrow points to the 'Apply' button in the 'Filter' section, and a red box highlights the 'IncomeBracket' attribute details. A stacked bar chart at the bottom right visualizes the data for the 'IncomeBracket' attribute, with the 'responded' class (red) and the 'not responded' class (blue).

No.	Label	Count	Weight
1	0	361	361.0
2	1	126	126.0
3	2	201	201.0
4	3	143	143.0
5	4	190	190.0
6	5	238	238.0
7	6	131	131.0
8	7	110	110.0

# PREPROCESS THE DATA WITH WEKA LOAD THE DATASET FOR TRAINING

- ▶ **Nominal variables** – *labelled data*
- ▶ **1500 instances**

The screenshot shows the Weka Explorer interface with the 'Preprocess' tab selected. The 'Current relation' section displays 'Relation: bmwrepon' and 'Instances: 1500'. The 'Attributes' list includes 'IncomeBracket', 'FirstPurchase', 'LastPurchase', and 'responded'. The 'Selected attribute' section shows a table for 'IncomeBracket' with 8 distinct values. A bar chart at the bottom visualizes the distribution of these values.

No.	Label	Count	Weight
1	0	361	361.0
2	1	126	126.0
3	2	201	201.0
4	3	143	143.0
5	4	190	190.0
6	5	238	238.0
7	6	131	131.0
8	7	110	110.0

# CLASSIFY THE DATA WITH WEKA DECISION TREE MODEL TRAINING

- ▶ Classify Tab;
- ▶ Use training set;
- ▶ Class = responded;
- ▶ Start building the model

The screenshot shows the Weka Explorer interface. The 'Classify' tab is active. The classifier is set to 'J48 -C 0.25 -M 2'. The 'Test options' section shows 'Use training set' selected. The 'Result list' shows a file named 'trees.J48' selected. The 'Classifier output' pane displays the following information:

```
Size of the tree :      5
Time taken to build model: 0.06 seconds
=== Evaluation on training set ===
Time taken to test model on training data: 0.01 seconds
=== Summary ===
Correctly Classified Instances      852      56.8 %
Incorrectly Classified Instances    648      43.2 %
Kappa statistic                     0.1315
Mean absolute error                  0.4867
Root mean squared error              0.4933
Relative absolute error              97.3854 %
Root relative squared error          98.6841 %
Total Number of Instances           1500
=== Detailed Accuracy By Class ===
              TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
              0.437   0.306   0.578     0.437   0.498     0.135   0.581    0.538     1
              0.694   0.563   0.562     0.694   0.621     0.135   0.581    0.563     0
Weighted Avg.   0.568   0.437   0.570     0.568   0.561     0.135   0.581    0.551
=== Confusion Matrix ===
  a  b  <-- classified as
321 414 |  a = 1
234 531 |  b = 0
```

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



# CLASSIFY THE DATA WITH WEKA DECISION TREE MODEL TESTING

- ▶ Supplied test set;

The screenshot shows the Weka Explorer interface. The 'Classifier' tab is active, and the 'J48 -C 0.25 -M 2' classifier is selected. In the 'Test options' section, the 'Supplied test set' radio button is selected and highlighted with a red box. A red arrow points from this box to the 'Classifier output' pane. The output pane displays the following information:

```
Number of the tree : 5
Time taken to build model: 0.04 seconds
=== Evaluation on test set ===
Time taken to test model on supplied test set: 0.01 seconds
=== Summary ===
Correctly Classified Instances      852      56.8 %
Incorrectly Classified Instances    648      43.2 %
Kappa statistic                    0.1315
Mean absolute error                 0.4867
Root mean squared error             0.4933
Relative absolute error             97.3854 %
Root relative squared error         98.6841 %
Total Number of Instances          1500

=== Detailed Accuracy By Class ===

```

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.437	0.306	0.578	0.437	0.498	0.135	0.581	0.538	1
	0.694	0.563	0.562	0.694	0.621	0.135	0.581	0.563	0
Weighted Avg.	0.568	0.437	0.570	0.568	0.561	0.135	0.581	0.551	

```

=== Confusion Matrix ===
  a  b  <-- classified as
321 414 |  a = 1
234 531 |  b = 0

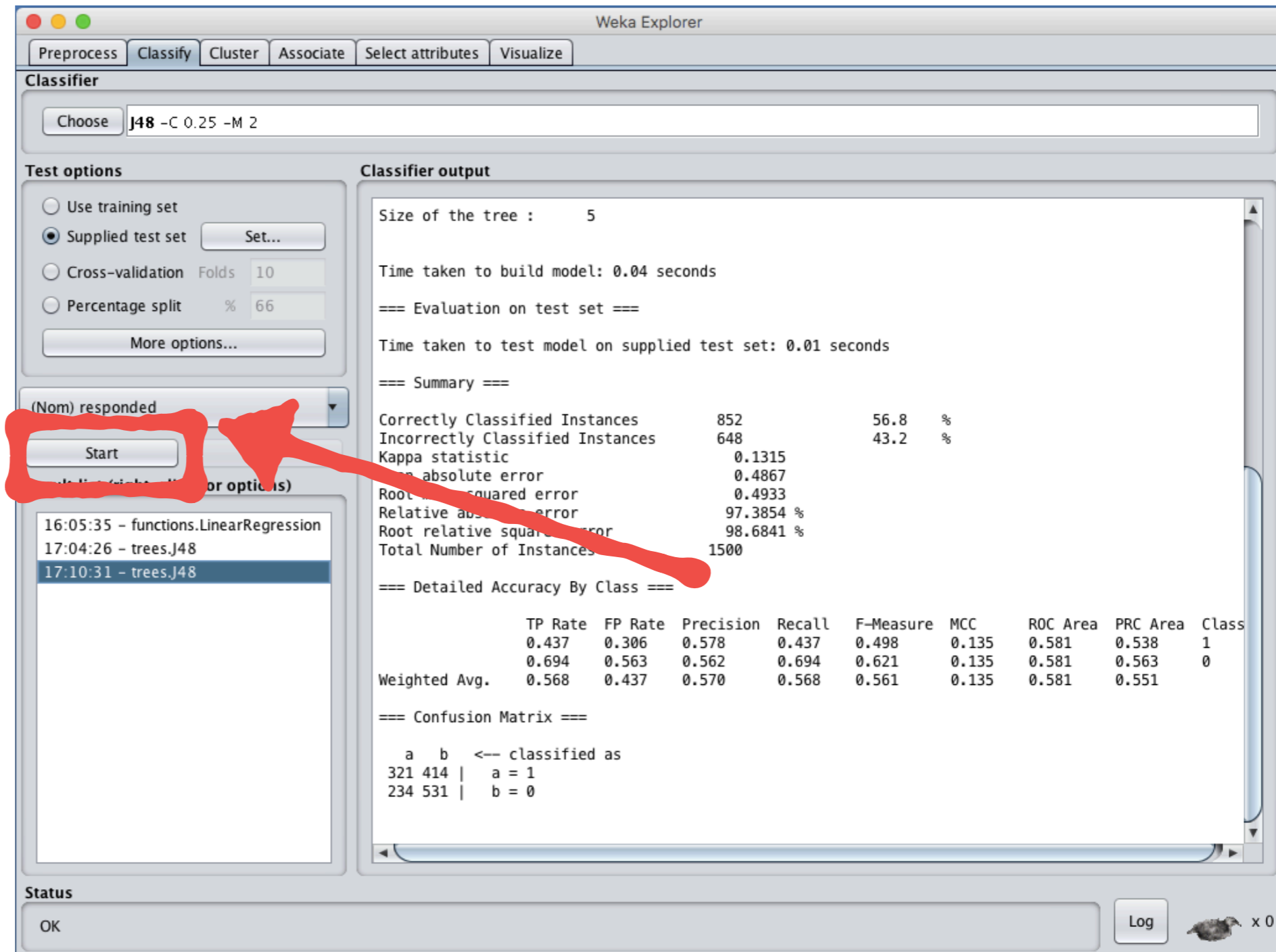
```

The 'Result list' shows the current test run: '17:10:31 - trees.J48'. The status bar at the bottom indicates 'OK'.



# CLASSIFY THE DATA WITH WEKA DECISION TREE MODEL TESTING

- ▶ Supplied test set;
- ▶ Start testing the model;



Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

Use training set

Supplied test set

Cross-validation Folds 10

Percentage split % 66

(Nom) responded

16:05:35 - functions.LinearRegression

17:04:26 - trees.J48

17:10:31 - trees.J48

Classifier output

Size of the tree : 5

Time taken to build model: 0.04 seconds

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds

=== Summary ===

Correctly Classified Instances	852	56.8	%
Incorrectly Classified Instances	648	43.2	%
Kappa statistic	0.1315		
Mean absolute error	0.4867		
Root relative squared error	0.4933		
Relative absolute error	97.3854	%	
Root relative squared error	98.6841	%	
Total Number of Instances	1500		

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.437	0.306	0.578	0.437	0.498	0.135	0.581	0.538	1
	0.694	0.563	0.562	0.694	0.621	0.135	0.581	0.563	0
Weighted Avg.	0.568	0.437	0.570	0.568	0.561	0.135	0.581	0.551	

=== Confusion Matrix ===

a	b	<-- classified as
321	414	a = 1
234	531	b = 0

Status

OK

Log x 0



# CLASSIFY THE DATA WITH WEKA DECISION TREE MODEL TESTING

- ▶ Supplied test set;
- ▶ Start testing the model;
- ▶ Compare models accuracy between train and test.

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

Use training set  
 Supplied test set Set...  
 Cross-validation Folds 10  
 Percentage split % 66  
More options...

(Nom) responded

Start Stop

Result list (right-click for options)

- 16:05:35 - functions.LinearRegression
- 17:04:26 - trees.J48
- 17:10:31 - trees.J48

Classifier output

Size of the tree : 5

Time taken to build model: 0.04 seconds

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.01 seconds

=== Summary ===

Correctly Classified Instances	852	56.8 %
Incorrectly Classified Instances	648	43.2 %
Kappa statistic	0.1315	
Mean absolute error	0.4867	
Root mean squared error	0.4933	
Relative absolute error	97.3854 %	
Root relative squared error	98.6841 %	
Total Number of Instances	1500	

==== Accuracy By Class ====

	TP	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.437		0.578	0.437	0.49	0.135	0.581	0.538	1
	0.694	0.563	0.562	0.694	0.67	0.135	0.581	0.563	0
Weighted Avg.	0.568	0.437	0.576	0.568	0.58	0.135	0.581	0.551	

==== Confusion Matrix ====

a	b	<-- classified as
321	414	a = 1
234	531	b = 0

Status

OK Log x 0

# AN EXAMPLE OF CLUSTERING CAR DEALERSHIP BEHAVIOUR ANALYSIS

The dealership has **kept track of how people walk through the dealership** and the showroom, what cars they look at, and how often they ultimately make purchases.

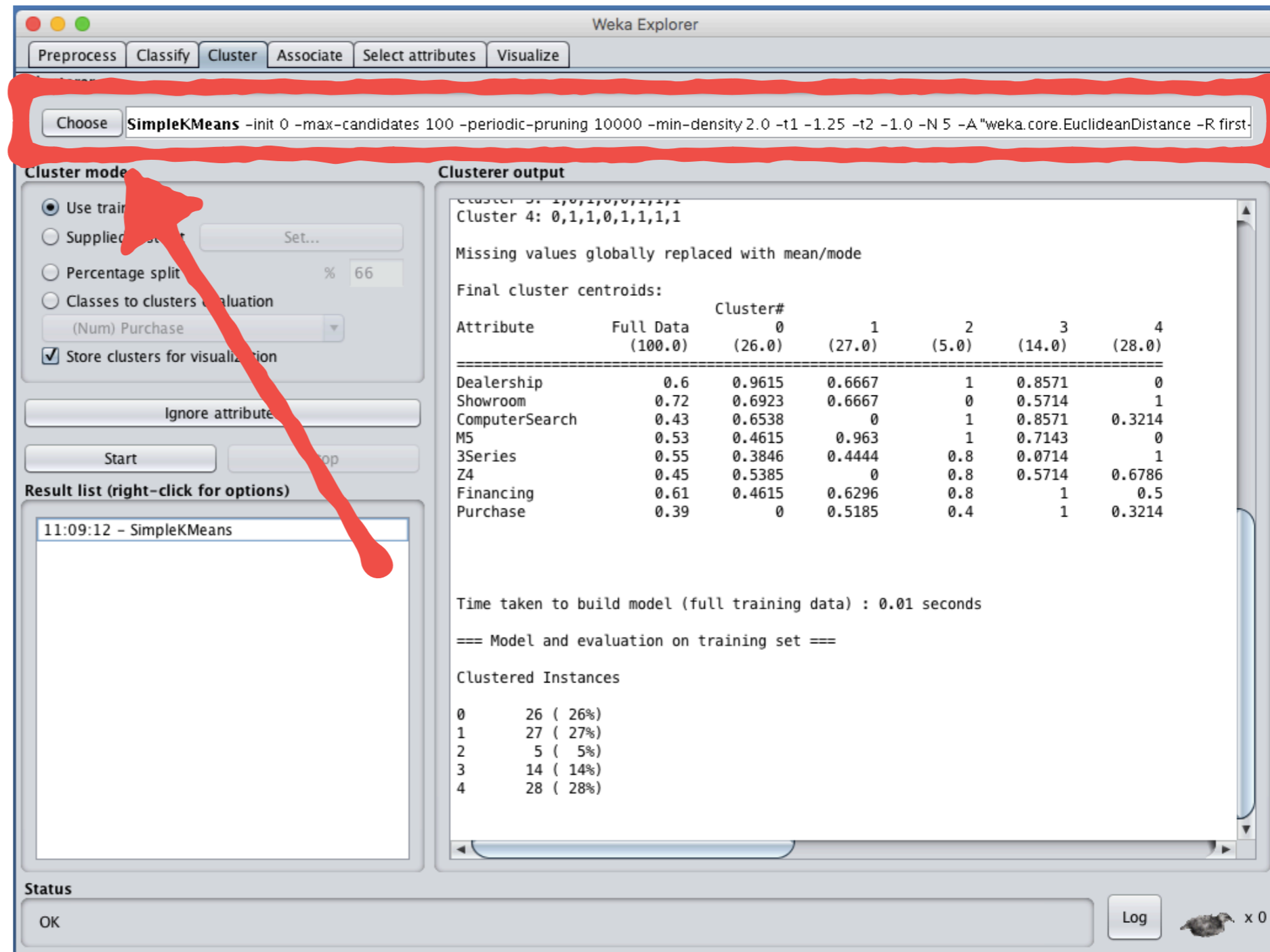
They are hoping to **mine this data by finding patterns** in the data and by using clusters to determine **if certain behaviours in their customers emerge**.



# CLUSTERING THE DATA WITH WEKA

## K-MEANS BEHAVIOUR ANALYSIS

- ▶ Cluster Tab;
- ▶ Use training set;
- ▶ No Class;
- ▶ Start



Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Choose SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 5 -A "weka.core.EuclideanDistance" -R first

Cluster mode

- Use training set
- Supplier set
- Percentage split % 66
- Classes to clusters evaluation
- Store clusters for visualization

Ignore attribute

Start

Result list (right-click for options)

11:09:12 - SimpleKMeans

Clusterer output

Cluster 0: 1,0,1,0,0,1,1,1  
Cluster 4: 0,1,1,0,1,1,1,1

Missing values globally replaced with mean/mode

Final cluster centroids:

Attribute	Full Data (100.0)	Cluster# 0 (26.0)	1 (27.0)	2 (5.0)	3 (14.0)	4 (28.0)
Dealership	0.6	0.9615	0.6667	1	0.8571	0
Showroom	0.72	0.6923	0.6667	0	0.5714	1
ComputerSearch	0.43	0.6538	0	1	0.8571	0.3214
M5	0.53	0.4615	0.963	1	0.7143	0
3Series	0.55	0.3846	0.4444	0.8	0.0714	1
Z4	0.45	0.5385	0	0.8	0.5714	0.6786
Financing	0.61	0.4615	0.6296	0.8	1	0.5
Purchase	0.39	0	0.5185	0.4	1	0.3214

Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0	26 ( 26%)
1	27 ( 27%)
2	5 ( 5%)
3	14 ( 14%)
4	28 ( 28%)

Status

OK

Log x 0



# CLUSTERING THE DATA WITH WEKA

## K-MEANS BEHAVIOUR ANALYSIS

- ▶ Cluster Tab;
- ▶ Use training set;
- ▶ No Class;
- ▶ Start;
- ▶ Evaluate patterns.

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Clusterer

Choose SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 5 -A "weka.core.EuclideanDistance -R first-

Cluster mode

Use training set  
 Supplied test set Set...  
 Percentage split % 66  
 Classes to clusters evaluation  
(Num) Purchase  
 Store clusters for visualization

Ignore attributes

Start Stop

Result list (right-click for options)

11:09:12 - SimpleKMeans

Clusterer output

Cluster 0: 1,0,1,0,0,1,1,1  
Cluster 4: 0,1,1,0,1,1,1,1

Missing values globally replaced with mean/mode

Print cluster centroids:

Attribute	Full Data (100.0)	Cluster# 0 (26.0)	1 (27.0)	2 (5.0)	3 (14.0)	4 (28.0)
Dealership	0.6	0.9615	0.6667	1	0.8571	0
Showroom	0.72	0.6923	0.6667	0	0.5714	1
ComputerSearch	0.43	0.6538	0	1	0.8571	0.3214
M5	0.53	0.4615	0.963	1	0.7143	0
3Series	0.55	0.3846	0.4444	0.8	0.0714	1
Z4	0.45	0.5385	0	0.8	0.5714	0.6786
Financing	0.61	0.4615	0.6296	0.8	1	0.5
Purchase	0.39	0	0.5185	0.4	1	0.3214

Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0	26 ( 26%)
1	27 ( 27%)
2	5 ( 5%)
3	14 ( 14%)
4	28 ( 28%)

Status

OK Log x 0

Final cluster centroids

Attribute	Full Data (100.0)	Cluster#				
		0 (26.0)	1 (27.0)	2 (5.0)	3 (14.0)	4 (28.0)
Dealership	0.6	0.9615	0.6667	1	0.8571	0
Showroom	0.72	0.6923	0.6667	0	0.5714	1
ComputerSearch	0.43	0.6538	0	1	0.8571	0.3214
M5	0.53	0.4615	0.963	1	0.7143	0
3Series	0.55	0.3846	0.4444	0.8	0.0714	1
Z4	0.45	0.5385	0	0.8	0.5714	0.6786
Financing	0.61	0.4615	0.6296	0.8	1	0.5
Purchase	0.39	0	0.5185	0.4	1	0.3214

Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

```

0      26 ( 26%)
1      27 ( 27%)
2       5 (  5%)
3      14 ( 14%)
4      28 ( 28%)

```

# EXAMPLE CONCLUSION

## K-MEANS BEHAVIOUR ANALYSIS

- ▶ **Cluster 0**– This group we can call the "Dreamers," as they appear to wander around the dealership, looking at cars parked outside on the lots, but trail off when it comes to coming into the dealership, and worst of all, they don't purchase anything.
- ▶ **Cluster 1**– We'll call this group the "M5 Lovers" because they tend to walk straight to the M5s, ignoring the 3-series cars and the Z4. However, they don't have a high purchase rate – only 52 percent. This is a potential problem and could be a focus for improvement for the dealership, perhaps by sending more salespeople to the M5 section.
- ▶ **Cluster 2**– This group is so small we can call them the "Throw-Aways" because they aren't statistically relevant, and we can't draw any good conclusions from their behaviour. (This happens sometimes with clusters and may indicate that you should reduce the number of clusters you've created).
- ▶ **Cluster 3**– This group we'll call the "BMW Babies" because they always end up purchasing a car and always end up financing it. Here's where the data shows us some interesting things: It appears they walk around the lot looking at cars, then turn to the computer search available at the dealership. Ultimately, they tend to buy M5s or Z4s (but never 3-series). This cluster tells the dealership that it should consider making its search computers more prominent around the lots (outdoor search computers?), and perhaps making the M5 or Z4 much more prominent in the search results. Once the customer has made up his mind to purchase the vehicle, he always qualifies for financing and completes the purchase.
- ▶ **Cluster 4**– This group we'll call the "Starting Out With BMW" because they always look at the 3-series and never look at the much more expensive M5. They walk right into the showroom, choosing not to walk around the lot and tend to ignore the computer search terminals. While 50 percent get to the financing stage, only 32 percent ultimately finish the transaction. The dealership could draw the conclusion that these customers looking to buy their first BMWs know exactly what kind of car they want (the 3-series entry-level model) and are hoping to qualify for financing to be able to afford it. The dealership could possibly increase sales to this group by relaxing their financing standards or by reducing the 3-series prices.



# References

- MOOC: <https://www.youtube.com/user/WekaMOOC>
- The Weka workbench: [https://www.cs.waikato.ac.nz/ml/weka/Witten\\_et\\_al\\_2016\\_appendix.pdf](https://www.cs.waikato.ac.nz/ml/weka/Witten_et_al_2016_appendix.pdf)
- Weka tutorial by Google: <https://www.youtube.com/watch?v=TF1yh5PKaql>
- dataset: <https://www.kaggle.com/datasets>
- dataset: <https://archive.ics.uci.edu/ml/datasets.html>

Slide courtesy of Dr. Stefano Pio Zingaro: <https://saltgz.github.io>