DATA ANALYTICS

Dr. Anil Maheshwari

Professor of Management Information Systems and
Director of Center for Data Analytics
Maharishi University of Management,
Fairfield, Iowa, USA.

Mc Graw Hill Education

McGraw Hill Education (India) Private Limited
CHENNAI

McGraw Hill Education Offices
Chennai New York St Louis San Francisco Auckland Bogotá Caracas
Kuala Lumpur Lisbon London Madrid Mexico City Milan Montreal
San Juan Santiago Singapore Sydney Tokyo Toronto

Section 2

SECTION 3

To answer this question, one should look at the past experiences, and see what decision was made in a similar instance, if such an instance exists. One could look up the database of past decisions to find the answer. Dataset 6.1 shows a list of the decisions taken in 14 instances of past soccer game situations. (Dataset courtesy: Witten, Frank, and Hall, 2010)

Dataset 6.1

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

If there was a row for Sunny/Hot/Normal/Windy condition in the data table, it would match the current problem; the decision from that row could be used to answer the current problem. However, there is no such past instance in this case. There are three disadvantages of looking up the data table

1. As mentioned earlier, how to decide if there isn't a row that corresponds to the exact situation today? If there is no exact matching instance available in the database, the past experience cannot guide the decision.
2. Searching through the entire past database may be time consuming, depending on the number of variables and the organization of the database.
3. What if the data values are not available for all the variables? In this instance, if the data for humidity variable was not available, looking up the past data would not help.

A better way of solving the problem is to abstract the knowledge from the past data into decision tree or rules. These rules can be represented in a decision tree, and then that tree can be used to make the decisions. The decision tree may not need values for all the variables.

DECISION TREE CONSTRUCTION

A decision tree is a hierarchically branched structure. What should be the first question asked in creating the tree? One should ask the more important questions first, and the less important questions later. What is the most important question that should be asked to solve the problem? How is the importance of the questions determined? Thus, how should the root node of the tree be determined?

Determining the Root Node of the Tree

In this example, there are four choices based on the four variables. One can begin by asking one of the following questions – what is the outlook, what is the temperature, what is the humidity, and what is the wind speed? A criterion should be used to evaluate these choices. The key criterion would be that, which one of these questions gives the most insight about the situation? Another way to look at it would be the criterion of frugality. That is, which question will provide us the shortest ultimate decision tree? Another way to look at this is that if one is allowed to ask only one question, which one would one ask? In this case, the most important question should be the one that, by itself, helps make the most correct decisions with the fewest errors. The four questions can now be systematically compared, to see which variable by itself will help make the most correct decisions. One should systematically calculate the correctness of decisions based on each question. Then one can select the question with the most correct predictions, or the fewest errors.

Start with the first variable in this case outlook. It can take three values, sunny, overcast, and rainy.

Start with the sunny value of outlook. There are five instances where the outlook is sunny. In 2 of the 5 instances, the play decision was yes, and in the other three, the decision was no. Thus, if the decision rule was that Outlook: sunny → No, then 3 out of 5 decisions would be correct, while 2 out of 5 such decisions would be incorrect. There are 2 errors out of 5. This can be recorded in Row 1.

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	

Similar analysis can be done for other values of the outlook variable. There are four instances where the outlook is overcast. In all the 4 instances, the play decision was yes. Thus, if the decision rule was that Outlook: overcast → Yes, then 4 out of 4 decisions would be correct, while none of decisions would be incorrect. There are 0 errors out of 4. This can be recorded in the next row.

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	
	Overcast → Yes	0/4	

There are five instances where the outlook is rainy. In 3 of the 5 instances, the play decision was yes, and in the other three, the decision was no. Thus, if the decision rule was that Outlook: rainy → Yes, then 3 out of 5 decisions would be correct, while 2 out of 5 decisions would be incorrect. There will be 2 out of 5 errors. This can be recorded in next row.

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	4/14
	Overcast → Yes	0/4	
	Rainy → Yes	2/5	

Adding up errors for all values of outlook, there are 4 errors out of 14. In other words, outlook gives 10 correct decisions out of 14, and 4 incorrect ones.

A similar analysis can be done for the other three variables. At the end of the analytical exercise, the following error table (Dataset 6.2) can be constructed.

Dataset 6.2

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	4/14
	Overcast → Yes	0/4	
	Rainy → Yes	2/5	
Temperature	Hot → No	2/4	5/14
	Mild → Yes	2/6	
	Cool → Yes	1/4	
Humidity	High → No	3/7	4/14
	Normal → Yes	1/7	
Windy	False → Yes	2/8	5/14
	True → No	3/6	

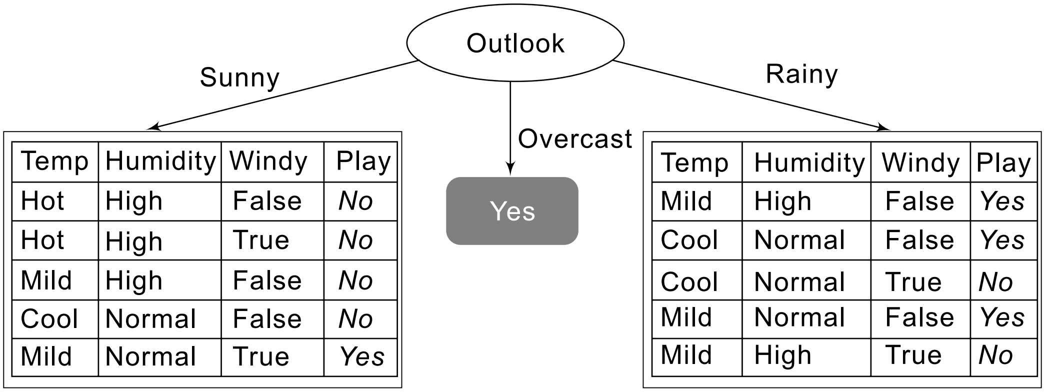
The variable that leads to the least number of errors (and thus the most number of correct decisions) should be chosen as the first node. In this case, two variables have the least number of errors. There is a tie between outlook and humidity, as both have 4 errors out of 14 instances. The tie can be broken using another criterion, the purity of resulting subtrees.

If all the errors were concentrated in few of the subtrees and some of the branches were completely free of error, then that is preferred from a usability perspective. Outlook has one error-free branch, for the overcast value, while there is no such pure subclass for humidity variable. Thus, the tie is broken in favor of outlook. The decision tree will use outlook as the first node, or the first splitting variable. The first question that should be asked to solve the play problem is, 'What is the value of outlook'?

Splitting the Tree

From the root node, the decision tree will be split into three branches or subtrees, one for each of the three values of outlook. Data for the root node (the entire data) will be divided into three segments, one for each of the value of outlook. The sunny branch will inherit the data for the instances that had 'sunny' as the value of outlook. These will be used for further building of that subtree. Similarly, the rainy branch will inherit data for the instances that had 'rainy' as the value of outlook. These will be used for further building of that subtree. The overcast branch will inherit the data for the instances that had 'overcast' as the outlook. However, there will be no need to build further on that branch. There is a clear decision –Yes, for all instances when outlook value is overcast.

The decision tree will look like as follows (Figure 6.1) after the first level of splitting.



Temp	Humidity	Windy	Play
Hot	High	False	*No*
Hot	High	True	*No*
Mild	High	False	*No*
Cool	Normal	False	*No*
Mild	Normal	True	*Yes*

Temp	Humidity	Windy	Play
Mild	High	False	*Yes*
Cool	Normal	False	*Yes*
Cool	Normal	True	*No*
Mild	Normal	False	*Yes*
Mild	High	True	*No*

FIGURE 6.1

Determining the Next Nodes of the Tree

Similar recursive logic of tree building should be applied to each branch. For the sunny branch on the left, error values will be calculated for the three other variables – temperature, humidity and windy. Final comparison will look like as shown in Dataset 6.3 given below.

Dataset 6.3

Attribute	Rules	Error	Total Error
Temperature	Hot → No	0/2	1/5
	Mild → No	1/2	
	Cool → Yes	0/1	
Humidity	High → No	0/3	0/5
	Normal → Yes	0/2	
Windy	False → No	1/3	2/5
	True → Yes	1/2	

The variable of humidity shows the least amount of error, i.e., zero error. The other two variables have non-zero errors. Thus the Outlook: sunny branch on the left will use humidity as the next splitting variable.

Similar analysis should be done for the 'rainy' value of the tree. The following Dataset 6.4 depicts such analysis.

Dataset 6.4

Attribute	Rules	Error	Total Error
Temperature	Mild → Yes	1/3	2/5
	Cool → Yes	1/2	
Humidity	High → No	1/2	2/5
	Normal → Yes	1/3	
Windy	False → Yes	0/3	0/5
	True → No	0/2	

For the rainy branch, it can similarly be seen that the variable windy gives all the correct answers, while none of the other two variables makes all the correct decisions.

This is how the final decision tree will look like. Here it is produced using Weka open-source data mining platform (Figure 6.2). This is the model that abstracts the knowledge of the past data of decision.

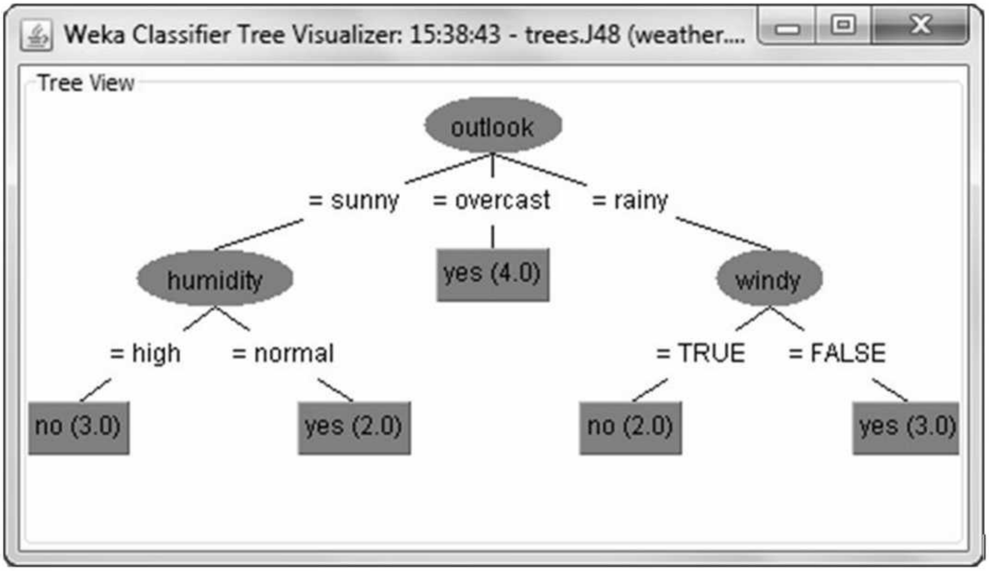


FIGURE 6.2 Decision Tree for the Weather Problem

This decision tree can be used to solve the current problem. Here is the problem again.

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	Normal	True	??

According to the tree, the first question to ask is about outlook. In this problem, the outlook is sunny, so the decision problem moves to the 'sunny' branch of the tree. The node in that subtree is humidity. In the problem, humidity is normal. That branch leads to an answer – Yes. Thus, the answer to the play problem is a yes.

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	Normal	True	Yes

LESSONS FROM CONSTRUCTING TREES

Here are some benefits of using the decision tree compared with looking up the answers from the data table (Table 6.1)

Table 6.1 Comparing Decision Tree with Table Lookup

	Decision Tree	Table Lookup
Accuracy	Varied level of accuracy	100% accurate
Generality	General, applies to all situations	Applies only when a similar case had occurred earlier
Frugality	Only three variables needed	All four variables are needed
Simplicity	Only one, or maximum two variable values are needed	All four variable values are needed
Ease	Logical and easy to understand	Can be cumbersome to look up; no understanding of the logic behind the decision

Here are a few observations about how the trees was constructed

- The final decision tree has zero errors in mapping to the prior data. In other words, the tree has a *predictive accuracy of 100%.* The tree completely fits the data. In real life situations, such perfect predictive accuracy is not possible when making decision trees. When there are larger, complicated datasets, with many more variables, a perfect fit is unachievable. This is especially true in business and social contexts, where things are not always fully clear and consistent.
- The decision tree algorithm *selected the minimum number of variables* that are needed to solve the problem. Thus, one can start with all available data variables, and let the decision-tree algorithm select the ones that are useful, and discard the rest.
- This tree is *almost symmetric* with all branches being of almost similar lengths. However, in real life situations, some of the branches may be much longer than the others, and the tree may need to be pruned to make it more balanced and usable.
- It may be possible to *increase predictive accuracy by making more sub-trees* and making the tree longer. However, the marginal accuracy gained from each subsequent level in the tree will be less, and may not be worth the loss in ease and interpretability of the tree. If the branches are long and complicated, it will be difficult to understand and use. The longer branches may need to be trimmed to keep the tree easy to use.
- A perfectly fitting tree has the *danger of over-fitting the data*, thus capturing all the random variations in the data. It may fit the training data well, but may not do well in predicting the future real instances.
- There was a *single best tree* for this data. There could however be two or more equally efficient decision trees of similar length with similar predictive

accuracy for the same dataset. Decision trees are *based strictly on patterns within the data*, and do not rely on any underlying theory of the problem domain. When multiple candidate trees are available, one could choose whichever is easier to understand, communicate or implement.

DECISION TREE ALGORITHMS

As we saw, decision trees employ the divide and conquer method. The data is branched at each node according to certain criteria until all the data is assigned to leaf nodes. It recursively divides a training set until each division consists of examples from one class.

The following is a pseudo code for making decision trees

1. Create a root node and assign all of the training data to it.
2. Select the best splitting attribute according to certain criteria.
3. Add a branch to the root node for each value of the split.
4. Split the data into mutually exclusive subsets along the lines of the specific split.
5. Repeat steps 2 and 3 for each and every leaf node until a stopping criteria is reached.

There are many algorithms for making decision trees. Decision tree algorithms differ on three key elements

Splitting Criteria

(a) Which variable to use for the first split? How should one determine the most important variable for the first branch and subsequently for each subtree?

Algorithms use different measures like least errors, information gain, Gini's coefficient etc., to compute the splitting variable that provides the most benefit. Information gain is a mathematical construct to compute the reduction in information entropy from a prior state to the next state that takes some information as given. The greater the reduction in entropy, the better it is. The Gini coefficient is a statistical concept that measures the inequality among values of a frequency distribution. The lower the Gini's coefficient, the better it is.

(b) What values to use for the split? If the variables have continuous values such as for age or blood pressure, what value-ranges should be used to make bins?

(c) How many branches should be allowed for each node? There could be binary trees, with just two branches at each node. Or there could be more branches allowed.

Stopping Criteria

When to stop building the tree? There are two major ways to make this determination. The tree building can be stopped when a certain depth of the branches has been reached and the tree becomes unreadable after that. The tree can also be stopped when the error level at any node is within predefined tolerable levels.

Pruning

It is the act of reducing the size of decision trees by removing sections of the tree that provide little value. The decision tree could be trimmed to make it more balanced, more general and more easily usable. The symptoms of an over-fitted tree are that it is too deep with too many branches which may reflect anomalies due to random noise or outliers instead of the underlying relationship. Pruning is often done after the tree is constructed. There are two approaches to avoid over-fitting.

- Prepruning means to halt the tree construction early, when certain criteria are met. The downside is that, it is difficult to decide what criteria to use for halting the construction, because we do not know what may happen subsequently if we keep growing the tree.
- Postpruning means removing branches or subtrees from a "fully grown" tree. This method is commonly used. C4.5 algorithm uses a statistical method to estimate the errors at each node for pruning. A validation set may be used for pruning as well.

The most popular decision tree algorithms are C5, CART and CHAID (Table 6.2)

Table 6.2 Comparing Popular Decision Tree Algorithms

DecisionTree	C4.5	CART	CHAID
Full name	Iterative Dichotomizer (ID3)	Classification and Regression Trees	Chi-square Automatic Interaction Detector
Basic algorithm	Hunt's algorithm	Hunt's algorithm	Adjusted significance testing
Developer	Ross Quinlan	Bremman	Gordon Kass

(*contd.*)

When developed	1986	1984	1980
Type of trees	Classification	Classification and Regression trees	Classification and Regression
Serial implementation	Tree growth and Tree pruning	Tree growth and Tree pruning	Tree growth and Tree pruning
Type of data	Discrete and Continuous; Incomplete data	Discrete and Continuous	Non-normal data also accepted
Type of splits	Multi-way splits	Binary splits only; clever surrogate splits to reduce tree depth	Multiway splits as default
Splitting criteria	Information gain	Gini's coefficient, and others	*Chi*-square test
Pruning criteria	Clever bottom-up technique avoids over-fitting	Remove weakest links first	Trees can become very large
Implementation	Publicly available	Publicly available in most packages	Popular in market research for segmentation

Conclusion

Decision trees are the most popular, versatile, and easy to use data mining technique with high predictive accuracy. They are also very useful as communication tools with executives. There are many successful decision tree algorithms. All publicly available data mining software platforms offer multiple decision tree implementations.

Review Questions

1. What is a decision tree? Why are decision trees the most popular classification technique?
2. What is a splitting variable? Describe three criteria for choosing a splitting variable.
3. What is pruning? What are prepruning and postpruning techniques? Why choose one over the other?
4. What are Gini's coefficient and information gain?

Hands-on Exercise

Create a decision tree for the for the data given in Dataset 6.5. The objective is to predict the class category (Loan approved or not).

Dataset 6.5

Age	Job	House	Credit	Loan Approved
Young	False	No	Fair	No
Young	False	No	Good	No
Young	True	No	Good	Yes
Young	True	Yes	Fair	Yes
Young	False	No	Fair	No
Middle	False	No	Fair	No
Middle	False	No	Good	No
Middle	True	Yes	Good	Yes
Middle	False	Yes	Excellent	Yes
Middle	False	Yes	Excellent	Yes
Old	False	Yes	Excellent	Yes
Old	False	Yes	Good	Yes
Old	True	No	Good	Yes
Old	True	No	Excellent	Yes
Old	False	No	Fair	No

Then solve the following problem using the model.

Age	Job	House	Credit	Loan Approved
Young	False	False	Good	??

True/False

1. Decision trees are essentially a hierarchy of if-then statements.
2. Decision trees apply only to strategic decisions made by executives.
3. A good decision tree should be short and ask only a few meaningful questions.
4. A decision tree should be balanced so it can make accurate predictions.
5. Decision trees use statistical techniques to abstract knowledge from data.
6. A good decision tree does not need to be colorful.
7. A decision tree can have any number of branches at any of the nodes.
8. It is desirable to have a 100% predictive accuracy of the tree, even if the tree becomes very long.
9. Making a decision tree is a recursive process.

10. The way of selecting the most important variable in constructing a decision tree is called the splitting criteria.

Liberty Stores Case Exercise: Step 5

Liberty is constantly evaluating requests for opening new stores. They would like to formalize the process for handling many requests, so that the best candidates are selected for detailed evaluation.

Develop a decision tree for evaluating new stores options. Dataset 6.6 shows the training data.

Dataset 6.6

City Size	Average Income	Local Investors	LOHAS Awareness	Decision
Big	High	Yes	High	Yes
Medium	Medium	No	Medium	No
Small	Low	Yes	Low	No
Big	High	No	High	Yes
Small	Medium	Yes	High	No
Medium	High	Yes	Medium	Yes
Medium	Medium	Yes	Medium	No
Big	Medium	No	Medium	No
Medium	High	Yes	Low	No
Small	High	No	High	Yes
Small	Medium	No	High	No
Medium	High	No	Medium	No

Use decision tree to answer the following question.

City Size	Average Income	Local Investors	LOHAS Awareness	Decision
Medium	Medium	No	Medium	??

7 Regression

Learning Objectives

- Understand the basic concept of correlation and regression
- Learn how to do a regression exercise
- Learn how to do a nonlinear regression exercise
- Know about logistic regression as a technique for classification
- Appreciate the advantages and disadvantages of regression

INTRODUCTION

Regression is a well-known statistical technique to model the predictive relationship between several independent variables (DVs) and one dependent variable. The objective is to find the best-fitting curve for a dependent variable in a multidimensional space, with each independent variable being a dimension. The curve could be a straight line, or it could be a nonlinear curve. The quality of fit of the curve to the data can be measured by a coefficient of correlation (r), which is the square root of the amount of variance explained by the curve.

The key steps for regression are simple

1. List all the variables available for making the model.
2. Establish a Dependent Variable (DV) of interest.
3. Examine visual (if possible) relationships between variables of interest.
4. Find a way to predict DV using other variables.

Caselet: Data Driven Prediction Markets

Traditional pollsters still seem to be using methodologies that worked well a decade or two ago. Nate Silver is a new breed of data-based political forecasters who are seeped in big data and advanced analytics. In the 2012 elections, he predicted that Obama would win the election with 291 electoral votes, compared to 247 for Mitt Romney, giving the President a 62% lead and reelection. He stunned the political forecasting world by correctly predicting the Presidential winner in all 50 states, including all nine swing states. He also correctly predicted the winner in 31 of the 33 US Senate races.

Nate Silver brings a different view to the world of forecasting political elections, viewing it as a scientific discipline. State the hypothesis scientifically, gather all available information, analyze the data and extract insights using sophisticated models and algorithms and finally, apply human judgment to interpret those insights. The results are likely to be much more grounded and successful. (Source: The Signal and the Noise: Why Most Predictions Fail but Some Don't, by Nate Silver, 2012)

1. *What is the impact of this story on traditional pollsters and commentators?*

CORRELATIONS AND RELATIONSHIPS

Statistical relationships are about which elements of data hang together and which ones hang separately. It is about categorizing variables that have a relationship with one another and categorizing variables that are distinct and unrelated to other variables. It is about describing significant positive relationships and significant negative differences.

The first and foremost measure of the strength of a relationship is co-relation (or correlation). The strength of a correlation is a quantitative measure that is measured in a normalized range between 0 and 1. A correlation of 1 indicates a perfect relationship, where the two variables are in perfect sync. A correlation of 0 indicates that there is no relationship between the variables.

The relationship can be positive or it can be an inverse relationship, that is, the variables may move together in the same direction or in the opposite direction. Therefore, a good measure of correlation is the correlation coefficient, which is the square root of correlation. This coefficient, called r, can thus range from −1 to +1. An r value of 0 signifies no relationship. An r value of 1 shows perfect relationship in the same direction, and an r value of −1 shows a perfect relationship but moving in opposite directions.

Given two numeric variables x and y, the coefficient of correlation r is mathematically computed by the following equation. $\overline{x}$ (called x-bar) is the mean of x, and $\overline{y}$ (y-bar) is the mean of y.

$$r = \frac{[(x - \overline{x}][y - \overline{y}]}{\sqrt{[(x - \overline{x})^2][(y - \overline{y})^2]}}$$

$$r = \frac{(x - \overline{x})(y - \overline{y})}{\sqrt{[(x - \overline{x})^2][(y - \overline{y})^2]}}$$

VISUAL LOOK AT RELATIONSHIPS

A scatter plot (or scatter diagram) is a simple exercise for plotting all the data points between two variables on a two-dimensional graph. It provides a visual layout of all the data points placed in that two-dimensional space. The scatter plot can be useful for graphically intuiting the relationship between the two variables.

Figure 7.1 shows many possible patterns in scatter diagrams.

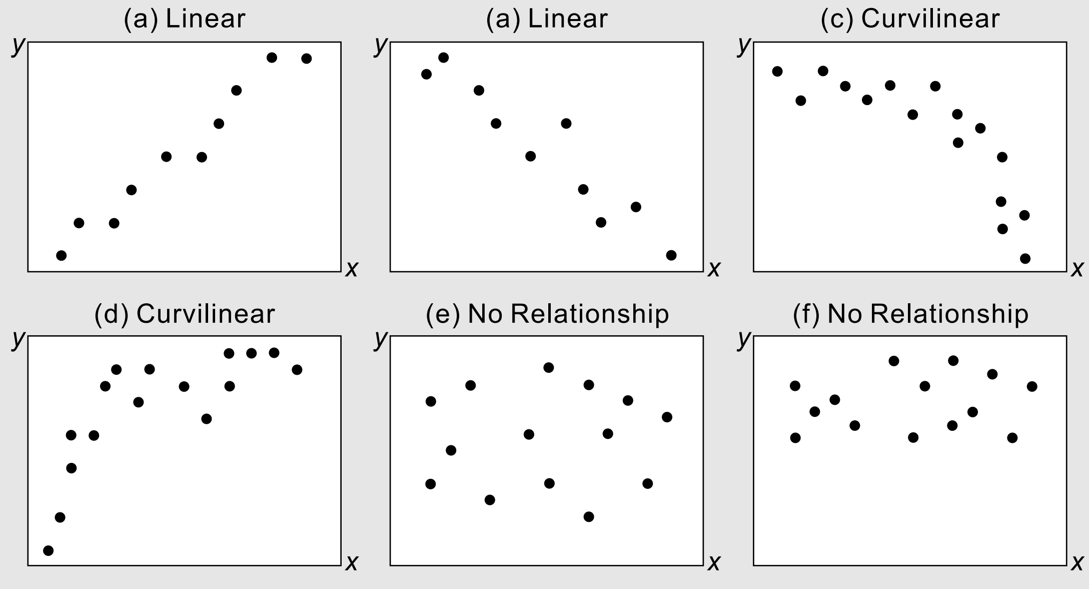


FIGURE 7.1 Scatter Plots showing Types of Relationships among Two Variables (*Source:* Groebner et al., 2013)

Chart (a) shows a very strong linear relationship between the variables x and y. This means the value of y increases proportionally with x. Chart (b) also shows a strong linear relationship between the variables x and y. Here, it is an inverse relationship. That means the value of y decreases proportionally with x.

Chart (c) shows a curvilinear relationship. It is an inverse relationship, which means that the value of y decreases proportionally with x. However, it seems a relatively well-defined relationship, like an arc of a circle, which can be represented by a simple quadratic equation (quadratic means the power of two, that is, using terms like x^2 and y^2). Chart (d) shows a positive curvilinear relationship. However, it does not seem to resemble a regular shape, and thus would not be a strong relationship. Charts (e) and (f) show no relationship,that means variables x and y are independent of each other.

Charts (a) and (b) are good candidates that model a simple linear regression model (the terms regression model and regression equation can be used interchangeably). Chart (c) too could be modeled with a little more complex, quadratic regression equation. Chart (d) might require an even higher order polynomial regression equation to represent the data. Charts (e) and (f) have no relationship, thus, they cannot be modeled together, by regression or using any other modeling tool.

Regression Exercise

The regression model is described as a linear equation that follows. y is the dependent variable, that is, the variable being predicted. x is the independent variable, or the predictor variable. There could be many predictor variables (such as x_1, x_2, …) in a regression equation. However, there can be only one dependent variable (y) in the regression equation.

$$y = \beta_0 + \beta_1 x + \varepsilon$$

Where β_0 and β_1 are the constant, and the co-efficient for the x variable; and ε is the random error variable.

A simple example of a regression equation would be to predict a house price from the size of the house. Dataset 7.1 shows a sample house prices data:

Dataset 7.1

House Price ($)	Size (Sq ft)
229,500	1850
273,300	2190
247,000	2100
195,100	1930
261,000	2300
179,700	1710
168,500	1550
234,400	1920
168,800	1840
180,400	1720
156,200	1660
288,350	2405
186,750	1525
202,100	2030
256,800	2240

The two dimensions (one predictor and one outcome variable) of the data can be plotted on a scatter diagram. A scatter plot with a best-fitting line looks like the graph that follows (Figure 7.2).

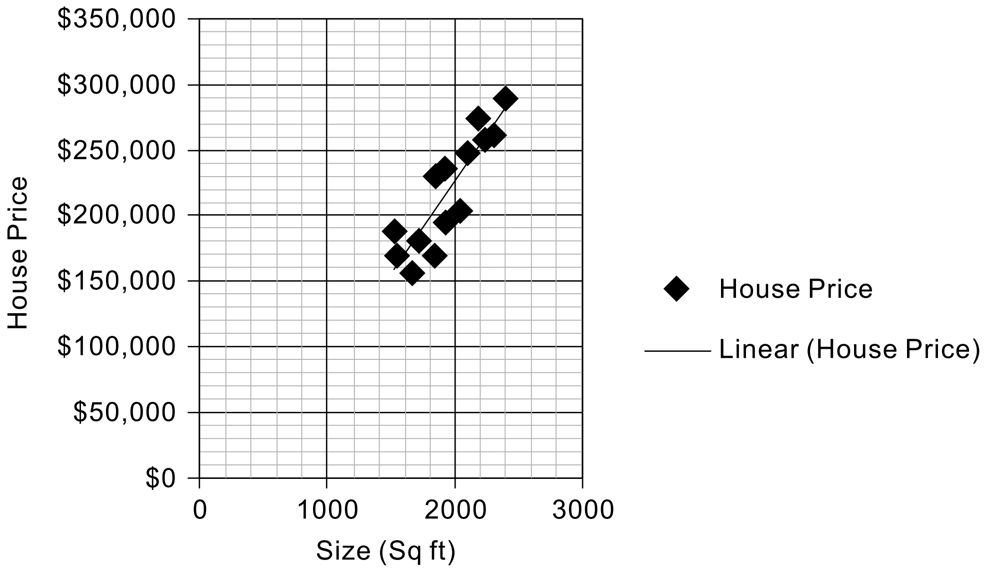


FIGURE 7.2 Scatter Plot and Regression Equation between House Price and House Size

Visually, one can see a positive correlation between house price and size (Sq ft). However, the relationship is not perfect. Running a regression model between the two variables produces the following output (truncated).

Regression Statistics	
R	0.891
r^2	0.794
	Coefficients
Intercept	–54191
Size (Sq ft)	139.48

It shows the coefficient of correlation to be 0.891. r^2, the measure of total variance explained by the equation, is 0.794 or 79%. That means the two variables are moderately and positively correlated. Regression coefficients help create the following equation for predicting house prices.

House Price ($) = 139.48 * Size (Sq ft) – 54191

This equation explains only 79% of the variance in house prices. Suppose other predictor variables are made available, such as the number of rooms in the house. It might help improve the regression model.

The house data will now look like as shown in Dataset 7.2 given below.

Dataset 7.2

House Price ($)	Size (Sq ft)	No. of Rooms
229,500	1850	4
273,300	2190	5
247,000	2100	4
195,100	1930	3
261,000	2300	4
179,700	1710	2
168,500	1550	2
234,400	1920	4
168,800	1840	2
180,400	1720	2
156,200	1660	2
288,350	2405	5
186,750	1525	3
202,100	2030	2
256,800	2240	4

While it is possible to make a three dimensional scatter plot, one can alternatively examine the correlation matrix among the variables.

	House Price	Size (Sq ft)	No. of Rooms
House Price	1		
Size (Sq ft)	0.891	1	
Rooms	0.944	0.748	1

It shows that the house price has a strong correlation with number of rooms (0.944) as well. Thus, it is likely that adding this variable to the regression model will add to the strength of the model.

Running a regression model between these three variables produces the following output (truncated).

Regression Statistics	
r	0.984
r^2	0.968

	Coefficients
Intercept	12923
Size (Sq ft)	65.60
Rooms	23613

It shows that the coefficient of correlation of this regression model is 0.984. R^2, the total variance explained by the equation, is 0.968 or 97%. That means the variables are positively and very strongly correlated. Adding a new relevant variable has helped improve the strength of the regression model.

Using the regression coefficients helps create the following equation for predicting house prices.

House Price ($) = 65.6 * Size (Sq ft) + 23613 * Rooms + 12924

This equation shows a 97% goodness of fit with the data, which is very good for business and economic data. There is always some random variation in naturally occurring business data, and it is not desirable to over fit the model to the data.

This predictive equation should be used for future transactions. Given a situation as below, it will be possible to predict the price of the house with 2000 Sq ft and 3 rooms.

House Price	Size (Sq ft)	#No. of Rooms
??	2000	3

House Price ($) = 65.6 * 2000 (Sq ft) + 23613 * 3 + 12924 = $214,963

The predicted values should be compared with the actual values to see how close the model is able to predict the actual value. As new data points become available, there are opportunities to fine-tune and improve the model.

NON-LINEAR REGRESSION EXERCISE

The relationship between the variables may also be curvilinear. For example, given past data from electricity consumption (kWh) and temperature (K), the objective is to predict the electrical consumption from the temperature value. Dataset 7.3 shows a dozen past observations.

Dataset 7.3

Kwatts	Temp (F)
12530	46.8
10800	52.1
10180	55.1
9730	59.2
9750	61.9
10230	66.2
11160	69.9
13910	76.8
15690	79.3
15110	79.7
17020	80.2
17880	83.3

In two dimensions (one predictor and one outcome variable), data can be plotted on a scatter diagram. A scatter plot with a best-fitting line looks like the graph on next page (Figure 7.3).

It is visually clear that the first line does not fit the data well. The relationship between temperature and Kwatts follows a curvilinear model, where it hits bottom at a certain value of temperature. The regression model confirms the relationship since R is only 0.77 and Rsquare is also only 60%. Thus, only 60% of the variance is explained.

This regression model can be enhanced by introducing a non linear variable (such as a quadratic variable Temp^2) in the equation. The second line is the relationship between kWh and Temp^2. The scatter plot shows that energy consumption has a strong linear relationship with Temp^2. Computing the regression model after adding the Temp^2 variable leads to the following results

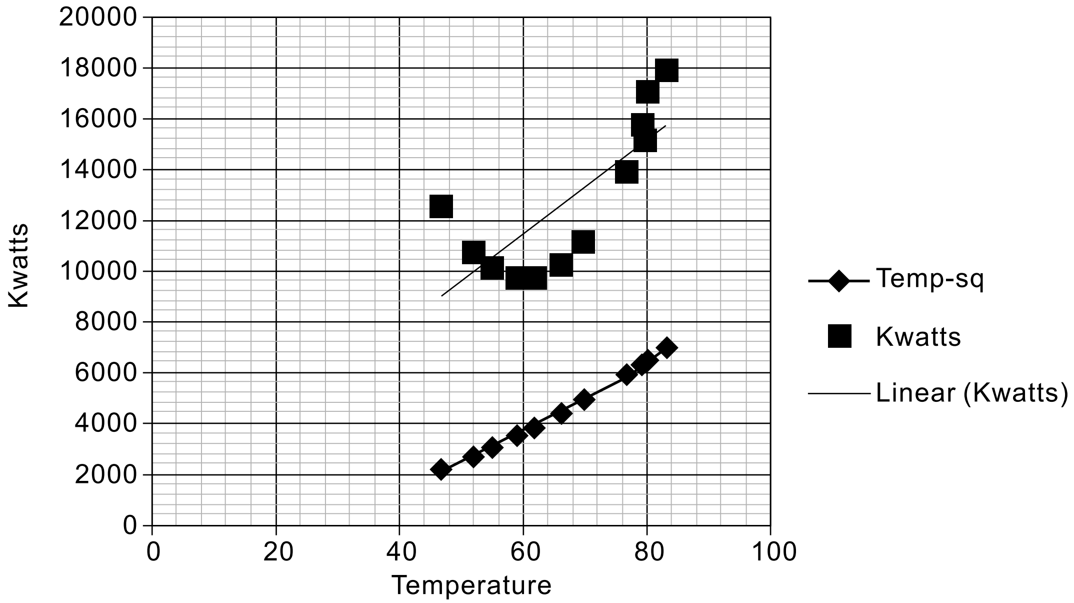


FIGURE 7.3 Scatter Plots showing Regression between (a) Kwatts and Temperature, and (b) Kwatts and Temperature Square

Regression Statistics	
r	0.992
r^2	0.984
	Coefficients
Intercept	67245
Temp (F)	–1911
Temp Sq	15.87

It shows that the coefficient of correlation of the regression model is now 0.99. R^2, the total variance explained by the equation is 0.985 or 98.5%. That means the variables are very strongly and positively correlated. The regression coefficients help create the following equation

Energy Consumption (Kwatts) = 15.87 * Temp^2 –1911 * Temp + 67245

This equation shows a 98.5% fit which is very good for business and economic contexts. Now one can predict the Kwatts value when the temperature is 72 degree.

Energy consumption = (15.87 * 72 * 72) – (1911 * 72) + 67245 = 11923 Kwatts

LOGISTIC REGRESSION

Regression models traditionally work with continuous numeric value data for dependent and independent variables. Logistic regression models can, however, work with dependent variables that have categorical values, such as whether a loan is approved or not. Logistic regression measures the relationship between a categorical dependent variable and one or more independent variables. For example, logistic regression might be used to predict whether a patient has a given disease (e.g., diabetes), based on observed characteristics of the patient (age, gender, body mass index, results of blood tests, etc.).

Logistical regression models use probability scores as the predicted values of the dependent variables. Logistic regression takes the natural logarithm of the probability of the dependent variable being a case (referred to as the logit function), and creates a continuous criterion as a transformed version of the dependent variable. Thus, the logit transformation is used in logistic regression as the dependent variable. The net effect is that although the dependent variable in logistic regression is binomial (or categorical, i.e., has only two possible values), the logit is the continuous function upon which linear regression is conducted. Here is the general logistic function with independent variable on the horizontal axis and the logit dependent variable on the vertical axis (Figure 7.4).

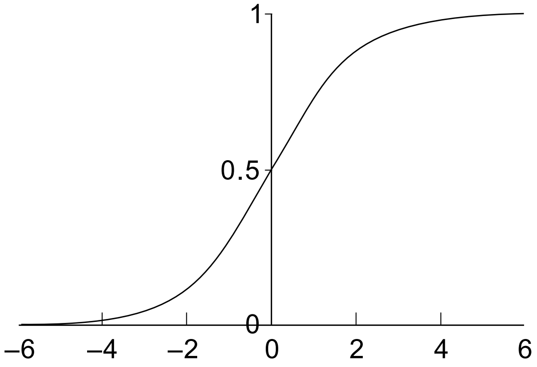


FIGURE 7.4 General Logit Function

All popular data mining platforms provide support for regular multiple regression models, as well as options for Logistic Regression.

ADVANTAGES AND DISADVANTAGES OF REGRESSION MODELS

Regression models are very popular because they offer many advantages. Few are as follows

- Regression models are easy to understand as they are built upon basic statistical principles such as correlation and least square error.
- Regression models provide simple algebraic equations that are easy to understand and use.
- The strength (or the goodness of fit) of the regression model is measured in terms of the correlation coefficients, and other related statistical parameters that are well understood.
- Regression models can match and beat the predictive power of other modeling techniques.
- Regression models can include all the variables that one wants to include in the model.
- Regression modeling tools are pervasive. They are found in statistical packages as well as data mining packages. MS-Excel spreadsheets can also provide simple regression modeling capabilities.

Regression models can however prove inadequate under many circumstances.

- Regression models cannot cover for poor data quality issues. If the data is not prepared well to remove missing values, or is not well-behaved in terms of a normal distribution, the validity of the model suffers.
- Regression models suffer from collinearity problems (meaning strong linear correlations among some independent variables). If the independent variables have strong correlations among themselves, then they will eat into each other's predictive power and the regression coefficients will lose their ruggedness. Regression models will not automatically choose between highly collinear variables, although some packages attempt to do that.
- Regression models can be unwieldy and unreliable if a large number of variables are included in the model. All variables entered into the model will be reflected in the regression equation, irrespective of their contribution to the predictive power of the model. There is no concept of automatic pruning of the regression model.
- Regression models do not automatically take care of nonlinearity. The user needs to imagine the kind of additional terms that might be needed to be added to the regression model to improve its fit.
- Regression models work only with numeric data and not with categorical variables. There are ways to deal with categorical variables though by creating multiple new variables with a yes or no value.

Conclusion

Regression models are simple, versatile, visual/graphical tools with high predictive ability. They include nonlinear as well as binary predictions. Regression models should be used in conjunction with other data mining techniques to confirm the findings.

Review Questions

1. What is a regression model?
2. What is a scatter plot? How does it help?
3. Compare and contrast decision trees with regression models.
4. Using the data given in Dataset 7.4 as shown below, create a regression model to predict the Test2 from Test1 score. Then predict the score for the one who got a 46 in Test1.

Dataset 7.4

Test1	Test2
59	56
52	63
44	55
51	50
42	66
42	48
41	58
45	36
27	13
63	50
54	81
44	56
50	64
47	50

True/False

1. Regression is an artificial intelligence technique.
2. In regression, a dependent variable is predicted using many independent variables.
3. Correlation coefficient (R) can take only positive values from zero to 1.

4. The best-fitting regression line can be straight or a curved one.
5. Regression model can automatically adjust the model to take into account any nonlinear relationship.
6. There can be only one dependent variable in one regression equation.
7. Regression models can be used for time-series analysis.
8. Regression models traditionally work with continuous numeric data.
9. Regression model is uniquely determined by the data.
10. Regression modeling tools are found in almost all statistical as well as data mining packages.

Liberty Stores Case Exercise: Step 6

Liberty wants to forecast its sales for next year for financial budgeting. The following Dataset 7.5 depicts its data.

Dataset 7.5

Year	Global GDP Index Per Capita	No. of Customer Service Calls ('000)	No. of Employees ('000)	No. of Items ('000)	Revenue ($M)
1	100	25	45	11	2000
2	112	27	53	11	2400
3	115	22	54	12	2700
4	123	27	58	14	2900
5	122	32	60	14	3200
6	132	33	65	15	3500
7	143	40	72	16	4000
8	126	30	65	16	4200
9	166	34	85	17	4500
10	157	47	97	18	4700
11	176	33	98	18	4900
12	180	45	100	20	5000

1. *Check the correlations. Which variables are strongly correlated?*
2. *Create a regression model that best predicts the revenue.*

8 Artificial Neural Networks

Learning Objectives

- Understand what are ANNs and why they are useful
- Know the design principles of ANN
- Learn about representation of the elements of an ANN
- Know the many architectures of ANNs
- Understand how ANNs are developed and trained
- Appreciate the many advantages and disadvantages of ANNs

INTRODUCTION

Artificial Neural Networks (ANNs) are inspired by the information processing model of the brain. The human brain consists of billions of neurons that link with one another in an intricate pattern. Every neuron receives information from many other neurons, processes it, gets excited or not, and passes its state information to other neurons.

Just like the brain is a multipurpose system, so also the ANNs are very versatile systems. They can be used for many kinds of pattern recognition and prediction. They are also used for classification, regression, clustering, association, and optimization activities. They are used in finance, marketing, manufacturing, operations, information systems applications, and so on.

ANNs are composed of a large number of highly interconnected processing elements (neurons) working in multilayered structures that receive inputs, process the inputs, and produce an output. An ANN is designed for a specific application, such as pattern recognition or data classification, and trained through a learning process. Just like in biological systems, ANNs make adjustments to the synaptic connections with each learning instance.

ANNs are like a black box trained into solving a particular type of problem, and they can develop high predictive powers. Their intermediate synaptic parameter values evolve as the system obtains feedback on its predictions, and thus an ANN learns from more training data (Figure 8.1).

Input
Artificial Neural Network (ANN)
Output

FIGURE 8.1 General ANN Model

Caselet: IBM Watson – Analytics in Medicine

The amount of medical information available is doubling every five years and much of this data is unstructured. Physicians simply don't have time to read every journal that can help them keep up to date with the latest advances. Mistakes in diagnosis are likely to happen and clients have become more aware of the evidence. Analytics will transform the field of medicine into evidence-based medicine. How can healthcare providers address these problems?

IBM's Watson cognitive computing system can analyze large amounts of unstructured text and develop hypotheses based on that analysis. Physicians can use Watson to assist in diagnosing and treating patients. First, the physician might describe symptoms and other related factors to the system. Watson can then identify the key pieces of information and mine the patient's data to find relevant facts about family history, current medications and other existing conditions. It combines this information with current findings from tests, and then forms and tests a hypotheses by examining a variety of data sources—treatment guidelines, electronic medical record data and doctors' and nurses' notes, as well as peer-reviewed research and clinical studies. From here, Watson can provide potential treatment options and its confidence rating for each suggestion.

Watson has been deployed at many leading healthcare institutions to improve the quality and efficiency of healthcare decisions, to help clinicians uncover insights from its patient information in electronic medical records (EMR),among other benefits.

1. *How would IBM Watson change medical practices in the future?*
2. *In what other industries and functions could this technology be applied?*

BUSINESS APPLICATIONS OF ANN

Neural networks are used most often when the objective function is complex, and where there exists plenty of data and the model is expected to improve over a period of time. A few sample applications are as follows

- They are used in stock price prediction where the rules of the game are extremely complicated, and a lot of data needs to be processed very quickly.
- They are used for character recognition, as in recognizing hand-written text, or damaged or mangled text. They are used in recognizing finger prints.

These are complicated patterns and are unique for each person. Layers of neurons can progressively clarify the pattern leading to a remarkably accurate result.

- They are also used in traditional classification problems, like approving a financial loan application.

DESIGN PRINCIPLES OF AN ARTIFICIAL NEURAL NETWORK

1. A neuron is the basic processing unit of the network. The neuron (or processing element) receives inputs from its preceding neurons (or PEs), does some nonlinear weighted computation on the basis of those inputs, transforms the result into its output value, and then passes on the output to the next neuron in the network (Figure 8.2). x's are the inputs, w's are the weights for each input, and y is the output.

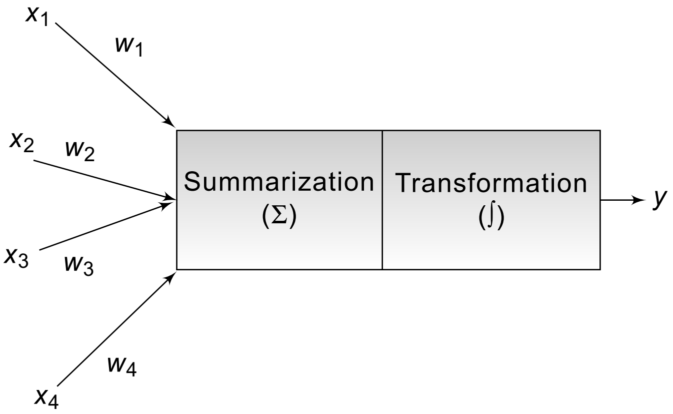


FIGURE 8.2 Model for a Single Artificial Neuron

2. A neural network is a multilayered model. There is at least one input neuron, one output neuron, and at least one processing neuron. An ANN with just this basic structure would be a simple, single-stage computational unit. A simple task may be processed by just that one neuron and the result may be communicated soon. ANNs however, may have multiple layers of processing elements in sequence. There could be many neurons involved in a sequence depending upon the complexity of the predictive action. The layers of PEs could work in sequence or in parallel (Figure 8.3).
3. The processing logic of each neuron may assign different weights to the various incoming input streams. The processing logic may also use nonlinear transformation, such as a sigmoid function, from the processed values to the output value. This processing logic and the intermediate weight and processing functions are just what works for the system as a whole, in its

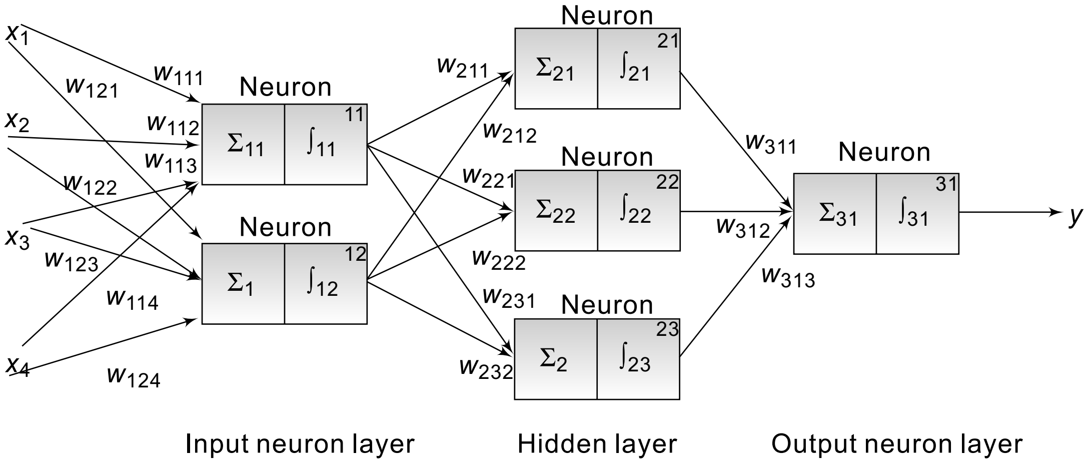


FIGURE 8.3 Model for a Multilayer ANN

objective of solving a problem collectively. Thus, neural networks are considered to be an opaque and a black-box system.

4. The neural network can be trained by making similar decisions over and over again with many training cases. It will continue to learn by adjusting its internal computation and communication based on feedback about its previous decisions. Thus, the neural networks become better at making a decision as they handle more and more decisions.

Depending upon the nature of the problem and the availability of good training data, at some point, the neural network will learn enough and begin to match the predictive accuracy of a human expert. In many practical situations, the predictions of ANN, trained over a long period of time with a large number of training data, have begun to decisively become more accurate than human experts. At that point, ANN can begin to be seriously considered for deployment in real situations in real time.

REPRESENTATION OF A NEURAL NETWORK

A neural network is a series of neurons that receive inputs from other neurons. They do a weighted summation function of all the inputs, using different weights (or importance) for each input. The weighted sum is then transformed into an output value using a transfer function.

Learning in ANN occurs when the various processing elements in the neural network adjust the underlying relationship (weights, transfer function, etc.)

between input and outputs, in response to the feedback on their predictions. If the prediction made was correct, then the weights would remain the same, but if the prediction was incorrect, then the parameter values would change.

The Transformation (Transfer) Function is any function suitable for the task at hand. The transfer function for ANNs is usually a nonlinear sigmoid function. Thus, if the normalized computed value is less than some value (say, 0.5) then the output value will be zero. If the computed value is at the cut-off threshold, then the output value will be 1. It could be a nonlinear hyperbolic function in which the output is either –1 or 1. Many other functions could be designed for any or all of the processing elements.

Thus, in a neural network, every processing element can potentially have a different number of input values, a different set of weights for those inputs, and a different transformation function. Those values support and compensate for one another until the neural network as a whole learns to provide the correct output, as desired by the user.

ARCHITECTING A NEURAL NETWORK

There are many ways to architect the functioning of an ANN using fairly simple and open rules with a tremendous amount of flexibility at each stage. The most popular architecture is a feedforward, multilayered perceptron with back-propagation learning algorithm. That means there are multiple layers of PEs in the system and the output of neurons are fed forward to the PEs in the next layers; and the feedback on the prediction is fed back into the neural network for learning to occur. This is essentially what was described in the earlier paragraphs. ANN architectures for different applications are shown in Table 8.1.

Table 8.1 ANN Architecture for Different Applications

Classification	Feedforward networks (MLP), radial basis function and probabilistic
Regression	Feedforward networks (MLP), radial basis function
Clustering	Adaptive resonance theory (ART), Self-organizing maps (SOMs)
Association Rule Mining	Hopfield networks

DEVELOPING AN ANN

It takes resources, training data, skill and time to develop a neural network. Most data mining platforms offer at least the MultiLayerPerceptron (MLP) algorithm to implement a neural network. Other neural network architectures include probabilistic networks and self-organizing feature maps.

The steps required to build an ANN are as follows

1. Gather data and divide into training data and test data. The training data needs to be further divided into training data and validation data.
2. Select the network architecture, such as Feedforward network.
3. Select the algorithm, such as Multi Layer Perception.
4. Set network parameters.
5. Train the ANN with training data.
6. Validate the model with validation data.
7. Freeze the weights and other parameters.
8. Test the trained network with test data.
9. Deploy the ANN when it achieves good predictive accuracy.

Training an ANN requires the training data be split into three parts (Table 8.2)

Table 8.2 ANN Training Datasets

Training Set	This dataset is used to adjust the weights on the neural network (~ 60%).
Validation Set	This dataset is used to minimize overfitting and verifying accuracy (~ 20%).
Testing Set	This dataset is used only for testing the final solution in order to confirm the actual predictive power of the network (~ 20%).
***k*-fold Cross Validation**	This approach means that the data is divided into *k* equal pieces, and the learning process is repeated *k* times with each piece becoming the training set. This process leads to less bias and more accuracy, but is more time consuming.

ADVANTAGES AND DISADVANTAGES OF USING ANNs

There are many benefits of using ANN. Some are given below

- ANNs impose very little restrictions on their use. ANN can deal with (identify/model) highly nonlinear relationships on their own, without much work from the user or analyst. They help find practical data-driven solutions where algorithmic solutions are nonexistent or are too complicated.

- There is no need to program neural networks as they learn from examples. They get better with use, without much programming effort.
- They can handle a variety of problem types, including classification, clustering, associations, etc.
- ANNs are tolerant of data quality issues and they do not restrict the data to follow strict normality and/or independence assumptions.
- They can handle both numerical and categorical variables.
- ANNs can be much faster than other techniques.
- Most importantly, they usually provide better results (prediction and/or clustering) compared to statistical counterparts, once they have been trained enough.

The key disadvantages arise from the fact that they are not easy to interpret or explain or compute.

- They are deemed to be black-box solutions, lacking explainability. Thus they are difficult to communicate about, except through the strength of their results.
- Optimal design of ANN is still an art. It requires expertise and extensive experimentation.
- It could be difficult to handle a large number of variables (especially the rich nominal attributes).
- It takes large datasets to train an ANN.

Conclusion

Artificial neural networks are complex systems that mirror the functioning of the human brain. They are versatile enough to solve many data mining tasks with high accuracy. However, they are like black boxes and they provide little guidance on the intuitive logic behind their predictions.

Review Questions

1. What is a neural network? How does it work?
2. Compare a neural network with a decision tree.
3. What makes a neural network versatile enough for supervised as well as nonsupervised learning tasks?

4. Examine the steps in developing a neural network for predicting stock prices. What kind of objective function and what kind of data would be required for a good stock price predictor system using ANN?

True/False

1. ANN is a machine learning technique.
2. ANNs are like a black box trained into solving a particular type of problem.
3. ANNs can be used for Classification, Clustering, Association Rules, and Optimization activities.
4. The processing logic of each neuron may assign different weights to the various incoming input streams.
5. A neural network is a multilayered model.
6. An ANN should have at least one input neuron, one output neuron, and at least three processing neurons.
7. ANN can deal with (identify/model) nonlinear relationships but with much work from the user or analyst.
8. Learning in ANN occurs when the various processing elements in the neural network adjust the underlying relationship (weights, transfer function, etc.) between inputs and outputs.
9. The most popular ANN architecture is self-organizing maps based learning algorithm.
10. IBM Watson is a form of a neural network.

9 Cluster Analysis

Learning Objectives

- Understand Cluster Analysis and its applications
- Learn the types of clusters and how they are represented
- Learn how the clustering technique works in practice
- Understand the *K*-means technique and its pseudocode
- Appreciate the many advantages and disadvantages of ANNs

INTRODUCTION

Cluster analysis is used for automatic identification of natural grouping of things. It is also known as the segmentation technique. In this technique, data instances that are similar to (or near) each other are categorized into one cluster. Similarly, data instances that are very different (or far away) from each other are moved into different clusters.

Clustering is an unsupervised learning technique as there is no output or dependent variable for which a right or wrong answer can be computed. The correct number of clusters or the definition of those clusters is not known ahead of time. Clustering techniques can only suggest to the user how many clusters would make sense from the characteristics of the data. The user can specify a different, larger or smaller, number of desired clusters based on their making business sense. The cluster analysis technique will then define many distinct clusters from analysis of the data, with cluster definitions for each of those clusters. However, there are good cluster definitions, depending on how closely the cluster parameters fit the data.

Caselet: Cluster Analysis

A national insurance company distributes its personal and small commercial insurance products through independent agents. They wanted to increase their sales by better understanding their customers. They were interested in increasing their market share by doing some direct marketing campaigns, however without creating a channel conflict with the independent agents. They were also interested in examining different customer segments based on their needs, and the profitability of each of those segments.

They gathered attitudinal, behavioral, and demographic data using a mail survey of 2000 U.S. households that own auto insurance. Additional geo-demographic and credit information was added to the survey data. Cluster analysis of the data revealed five roughly equal segments.

- **Non-Traditional** *interested in using the Internet and/or buying insurance at work.*
- **Direct Buyers** *interested in buying via direct mail or telephone.*
- **Budget Conscious** *interested in minimal coverage and finding the best deal.*
- **Agent Loyals** *expressed strong loyalty to their agents and high levels of personal service.*
- **Hassle-Free** *similar to Agent Loyals but less interested in face-to-face service.*

(Source: greenbook.org*)*

1. *Which customer segments would you choose for direct marketing? Will these create a channel conflict?*
2. *Could this segmentation apply to other service businesses? Which ones?*

APPLICATIONS OF CLUSTER ANALYSIS

Cluster analysis is used in almost every field where there is a large variety of transactions. It helps provide characterization, definition, and labels for populations. It can help identify natural grouping of customers, products, patients, and so on. It can also help identify outliers in a specific domain and thus decrease the size and complexity of problems. A prominent business application of cluster analysis is in market research. Customers are segmented into clusters based on their characteristics—wants and needs, geography, price sensitivity, and so on. Here are some examples of clustering

Market Segmentation Categorizing customers according to their similarities, for instance by their common wants and needs, and propensity to pay can help with targeted marketing.

Product Portfolio People of similar sizes can be grouped together to make small, medium and large sizes for clothing items.

Text Mining Clustering can help organize a given collection of text documents according to their content similarities into clusters of related topics.

DEFINITION OF A CLUSTER

An operational definition of a cluster is that, given a representation of n objects, find K groups based on a measure of similarity, such that objects within the same group are alike but the objects in different groups are not alike.

However, the notion of similarity can be interpreted in many ways. Clusters can differ in terms of their shape, size, and density. Clusters are patterns and there can be many kinds of patterns. Some clusters are the traditional types, such as data points hanging together. However, there are other clusters, such as all points representing the circumference of a circle. There may be concentric circles with points of different circles representing different clusters. The presence of noise in the data makes the detection of the clusters even more difficult.

An ideal cluster can be defined as a set of points that is compact and isolated. In reality, a cluster is a subjective entity whose significance and interpretation requires domain knowledge. In the sample data below (Figure 9.1), how many clusters can one visualize?

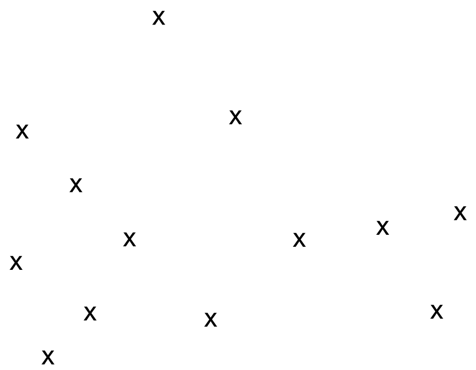


FIGURE 9.1 Visual Cluster Example

It seems like there are two clusters of approximately equal sizes. However, they can be seen as three clusters, depending on how we draw the dividing lines. There is not a truly optimal way to calculate it. Heuristics are often used to define the number of clusters.

REPRESENTING CLUSTERS

The clusters can be represented by a central or modal value. A cluster can be defined as the *centroid* of the collection of points belonging to it. A *centroid* is a measure of central tendency. It is the point from where the sum total of squared distance from all the points is the minimum. A real-life equivalent would be the citycenter as the point that is considered the most easy to use by all constituents of the city. Thus all cities are defined by their centers or downtown areas.

A cluster can also be represented by the most frequently occurring value in the cluster, i.e.,a cluster can be defined by its modal value. Thus, a particular cluster representing a social point of view could be called the ‘soccer moms’, even though not all members of that cluster need currently be a mom with soccer-playing children.

CLUSTERING TECHNIQUES

Cluster analysis is a machine-learning technique. The quality of a clustering result depends on the *algorithm*, the *distance* function, and the *application*. First, consider the distance function. Most cluster analysis methods use a distance measure to calculate the closeness between pairs of items. There are two major measures of distances – Euclidian distance ("as the crow flies" or straight line) is the most intuitive measure; the other popular measure is the Manhattan (rectilinear) distance, where one can go only in orthogonal directions. The Euclidian distance is the hypotenuse of a right triangle, while the Manhattan distance is the sum of the two legs of the right triangle. There are other measures of distance like Jacquard distance (to measure similarity of sets), or Edit distance (similarity of texts), and others.

In either case, the key objective of the clustering algorithm is the same, i.e., inter-cluster distance is maximized and intra-clusters distance is minimized.

There are many algorithms to produce clusters. There are top-down, hierarchical methods that start with creating a given number of best-fitting clusters. There are also bottom-up methods that begin with identifying naturally occurring clusters.

The most popular clustering algorithm is the *K*-means algorithm. It is a top-down, statistical technique, based on the method of minimizing the least squared distance from the center points of the clusters. Other techniques, such as neural networks, are also used for clustering. Comparing cluster algorithms is a difficult task as there is no single right number of clusters. However, the speed of the algorithm and its versatility in terms of different dataset are important criteria.

Here is the generic pseudocode for clustering

1. Pick an arbitrary number of groups/segments to be created.
2. Start with some initial randomly chosen center values for groups.
3. Classify instances to closest groups.
4. Compute new values for the group centers.
5. Repeat steps 3 and 4 till groups converge.
6. If clusters are not satisfactory, go to step 1 and pick a different number of groups/segments.

The clustering exercise can be continued with a different number of clusters and different location of those points. Clusters are considered good if the cluster definitions stabilize, and the stabilized definitions prove useful for the purpose at hand. Else, repeat the clustering exercise with a different number of clusters and different starting points for group means.

CLUSTERING EXERCISE

Here is a simple exercise to visually and intuitively identify clusters from the data as shown in Dataset 9.1. *X* and *Y* are the two dimensions of interest. The objective is to determine the number of clusters and the center points of those clusters.

Dataset 9.1

X	*Y*
2	4
2	6
5	6
4	7
8	3
6	6
5	2
5	7
6	3
4	4

A scatter plot of 10 items in 2 dimensions shows them distributed fairly randomly. As a bottom-up technique, the number of clusters and their centroids can be intuited (Figure 9.2).

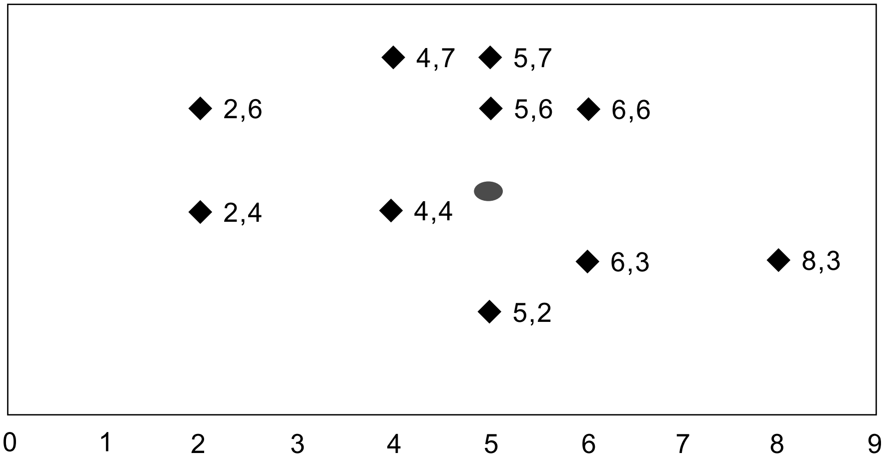


FIGURE 9.2 Initial Data Points and the Centroid (Shown as Thick Dot)

The points are distributed randomly enough such that it is considered as one cluster. The solid circle represents the central point (centroid) of these points.

However, there is a big distance between the points (2, 6) and (8, 3). So, this data can be broken into 2 clusters. The 3 points at the bottom right can form one cluster and the other 7 forms the other cluster. The two clusters look like as follows (Figure 9.3). The two circles are the new centroids.

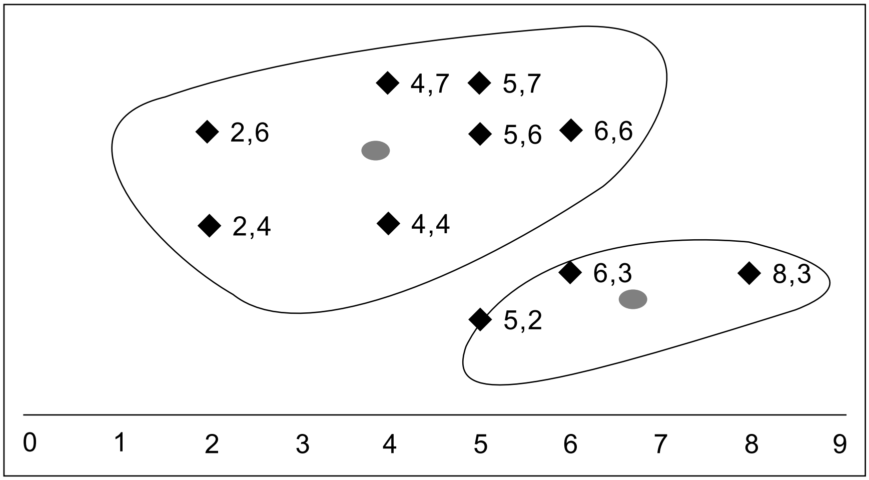


FIGURE 9.3 Dividing into Two Clusters (Centroids Shown as Thick Dots)

The bigger cluster seems too far apart. So, it seems like the 4 points on the top form a separate cluster. The three clusters look like as follows (Figure 9.4).

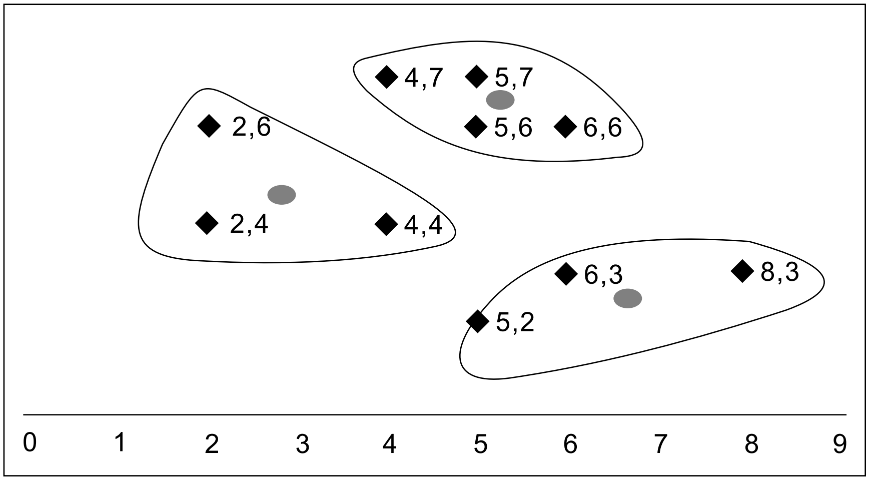


FIGURE 9.4 Dividing into Three Clusters (Centroids Shown as Thick Dots)

This solution has 3 clusters. The cluster on the right is far from the other 2 clusters. However, its centroid is not too close to all the data points. The cluster at the top looks very tight-fitting, with a nice centroid. The third cluster, at the left, is spread out and may not be of much usefulness.

This was a bottom-up exercise in visually producing 3 best-fitting cluster definitions from the given data. The right number of clusters will depend on the data and the application for which the data would be used.

K-MEANS ALGORITHM FOR CLUSTERING

K-means is the most popular clustering algorithm. It iteratively computes the clusters and their centroids. It is a top down approach to clustering. Starting with a given number of *K* clusters, say 3 clusters,that means 3 random centroids will be created as starting points of the centers. The circles are initial cluster centroids (Figure 9.5).

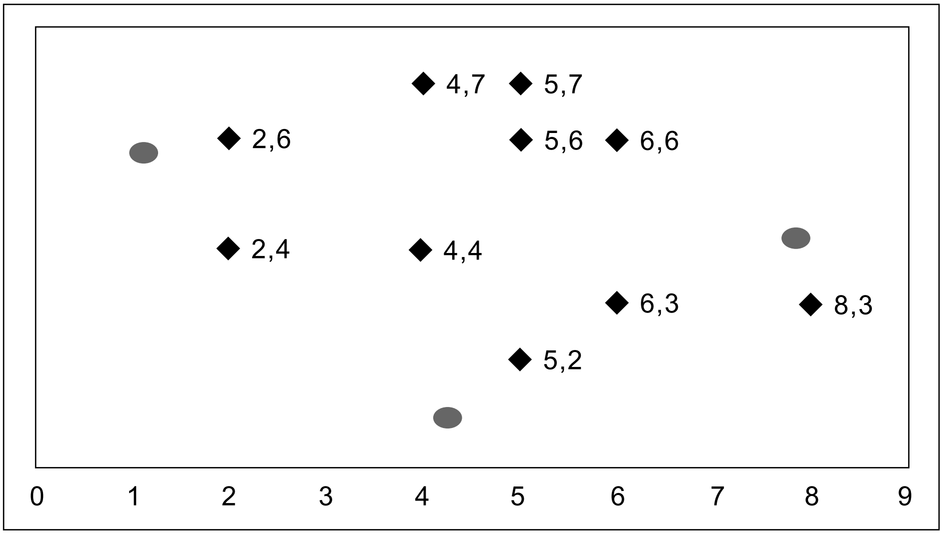


FIGURE 9.5 Randomly Assigning Three Centroids for Three Data Clusters

Step 1 For a data point, distance values will be from each of the three centroids. The data point will be assigned to the cluster with the shortest distance to the centroid. All data points will thus, be assigned to one data point or the other (Figure 9.6). The arrows from each data element show the centroid that the point is assigned to.

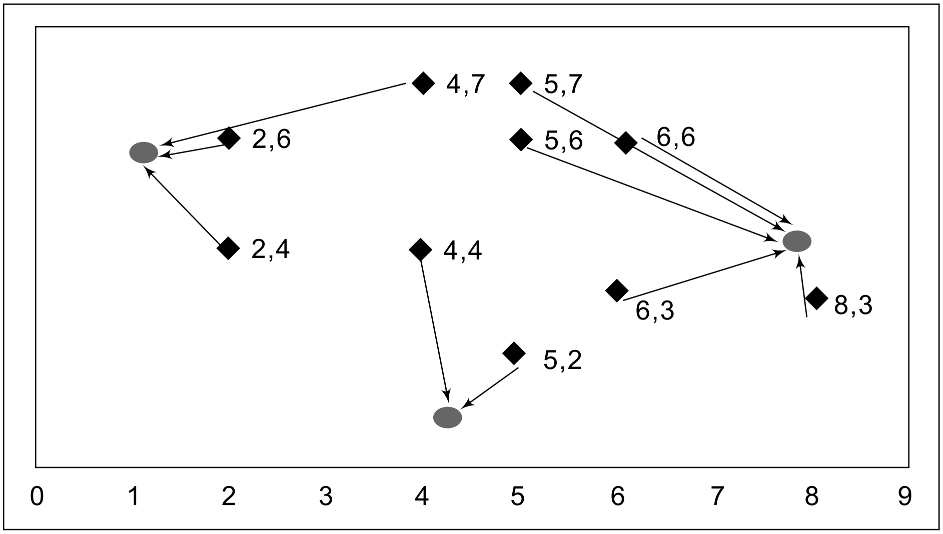


FIGURE 9.6 Assigning Data Points to Closest Centroid

Step 2 The centroid for each cluster will now be recalculated such that it is closest to all the data points allocated to that cluster. The dashed arrows show the centroids being moved from their old (shaded) values to the revised new values (Figure 9.7).

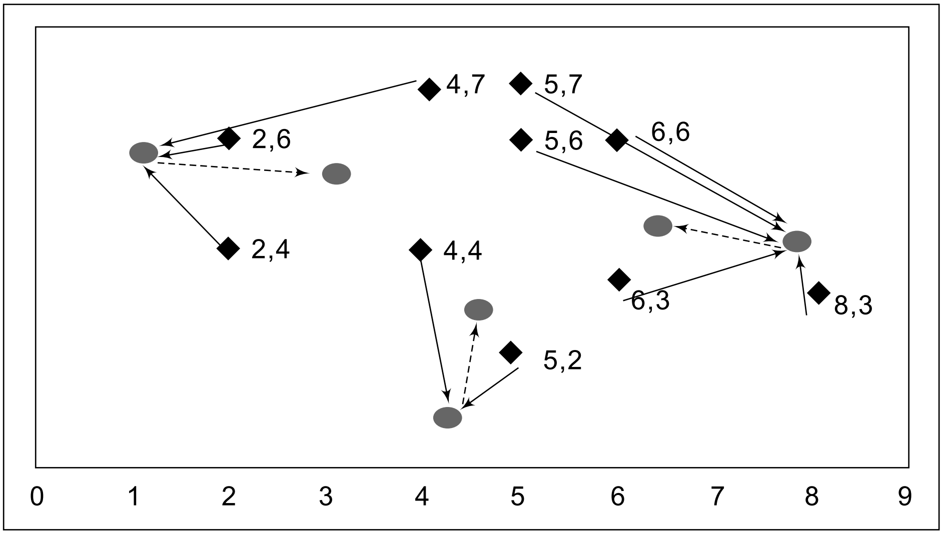


FIGURE 9.7 Recomputing Centroids for Each Cluster

Step 3 Once again, data points are assigned to the three centroids closest to it (Figure 9.8).

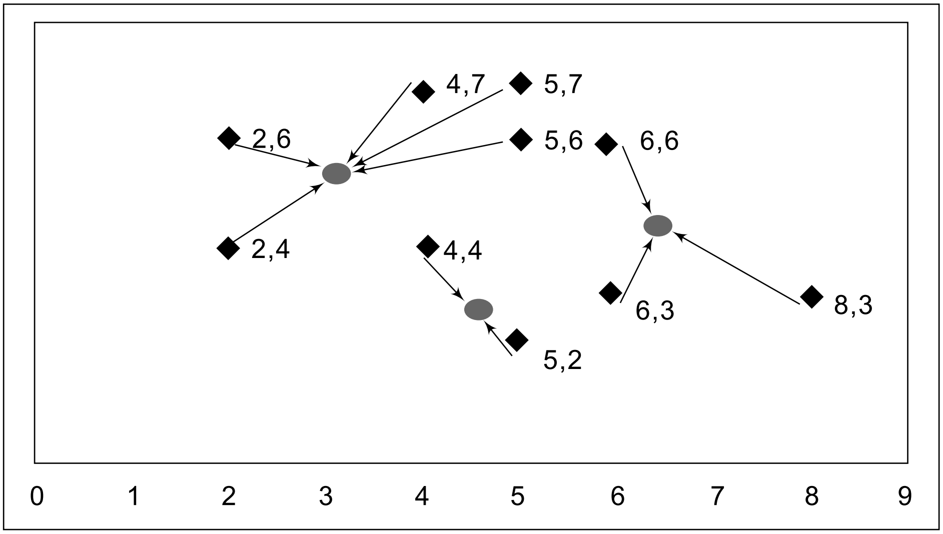


FIGURE 9.8 Assigning Data Points to Recomputed Centroids

The new centroids will be computed from the data points in the cluster until finally, the centroids stabilize in their locations. These are the three clusters computed by this algorithm.

The three clusters shown are – a 3-datapoints cluster with centroid (6.5, 4.5), a 2-datapoint cluster with centroid (4.5, 3) and a 5-datapoint cluster with centroid (3.5, 3) (Figure 9.9).

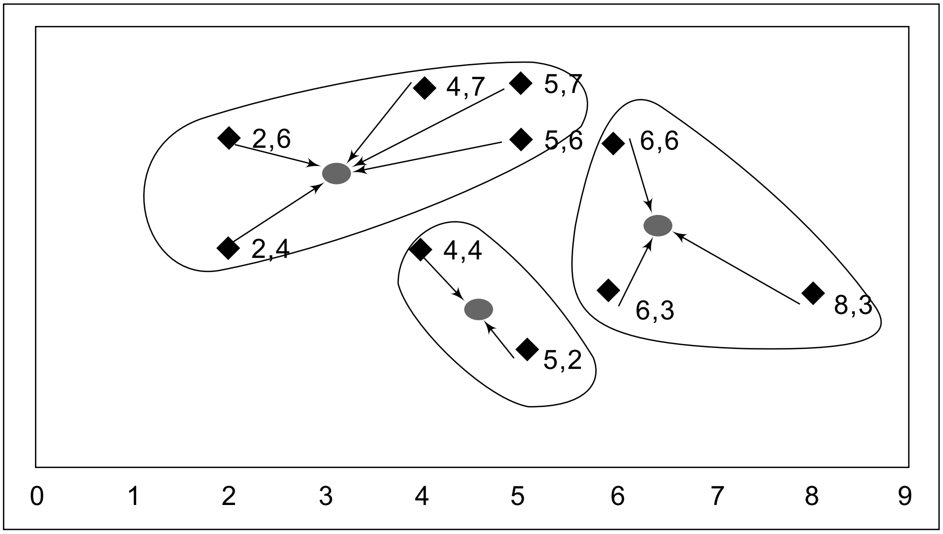


FIGURE 9.9 Recomputing Centroids for Each Cluster till Clusters Stabilize

These cluster definitions are different from the ones derived visually. This is a function of the random starting centroid values. The centroid points used earlier in the visual exercise were different from that chosen with the *K*-means clustering algorithm. The *K*-means clustering exercise should therefore, be run again with this data, but with new random centroid starting values. With many runs, the cluster definitions are likely to stabilize. If the cluster definitions do not stabilize, that may be a sign that the number of clusters chosen is too high or too low. The algorithm should also be run with different values of *K*.

Here is the pseudocode for implementing a *K*-means algorithm.

```
Algorithm $K$-Means ($K$ number of clusters, $D$ list of data points)
1. Choose $K$ number of random data points as initial centroids (cluster-centers)
2. Repeat till cluster-centers stabilize
    (a) {Allocate each point in $D$ to the nearest of $K$ centroids;
    (b) Compute centroid for the cluster using all points in the cluster}
```

SELECTING THE NUMBER OF CLUSTERS

The correct choice of the value of *K* is often ambiguous. It depends on the shape and scale of the distribution points in a dataset and the desired clustering resolution of the user. Heuristics are needed to pick the right number. One can graph the percentage of variance explained by the clusters against the number of clusters (Fig. 9.10). The first cluster will add more information (explain a lot of variance), but at some point the marginal gain in variance will fall, giving a sharp angle to the graph, looking like an elbow. Beyond that elbow point, adding more clusters will not add much incremental value. That would be the desired *K*.

To engage with the data and to understand the clusters better, it is often better to start with a small number of clusters such as 2 or 3, depending upon the dataset and the application domain. The number can be increased subsequently, as needed from an application point of view. This helps understand the data and the clusters progressively better.

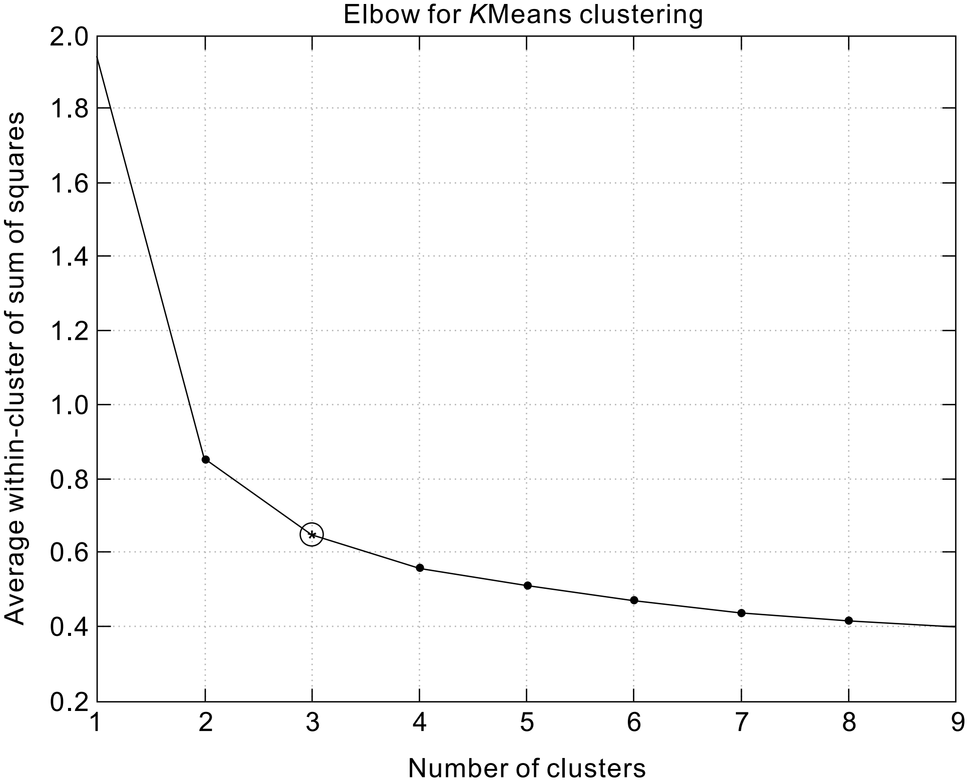


FIGURE 9.10 Elbow Method for Determining Number of Clusters in a Dataset

ADVANTAGES AND DISADVANTAGES OF *K*-MEANS ALGORITHM

There are many advantages of the *K*-means algorithm.

- *K*-means algorithm is simple, easy to understand and easy to implement.
- It is also efficient, in that, the time taken to cluster *K*-means rises linearly with the number of data points.
- No other clustering algorithm performs better than *K*-means, in general.

There are a few disadvantages too

- The user needs to specify an initial value of *K*.
- The process of finding the clusters may not converge.
- It is not suitable for discovering cluster shapes that are not hyperellipsoids (or hyperspheres).

Neural networks can also be deployed for clustering, using the appropriate objective function. The neural network will produce the appropriate cluster centroids and cluster population for each cluster.

Conclusion

Cluster analysis is a useful, unsupervised learning technique that is used in many business situations to segment the data into meaningful small groups. *K*-means algorithm is an easy statistical technique to iteratively segment the data. However, there is only a heuristic technique to select the right number of clusters.

Review Questions

1. What is unsupervised learning? When is it used?
2. Describe three business applications in your industry where cluster analysis will be useful.
3. Data about height and weight for a few volunteers is available. Create a set of clusters for the following data shown in Dataset 9.2, to decide how many sizes of T-shirts should be ordered.

Dataset 9.2

Height	Weight
71	165
68	165
72	180
67	113
72	178
62	101
70	150
69	172
72	185
63	149
69	132
61	115

True/False

1. Cluster analysis is used for market segmentation.
2. A good set of clusters should be of significantly different sizes.

3. Usually the clusters can each be represented by a central point, or the modal point, also called the 'centroid'.
4. A decision tree can help in determining the likely number of clusters in a given dataset.
5. The key objective of all clustering algorithms is the same, i.e.,inter-cluster distance is maximized and intra-cluster distance is minimized.
6. There are multiple ways of measuring the distance between data points.
7. An unlimited number of variables can be used for cluster analysis.
8. *K*-means is the most popular clustering algorithm.
9. *K*-means automatically produces the right number of clusters in the data.
10. Amazon uses clustering technique to recommend new products for its customers to buy.

Liberty Stores Case Exercise: Step 7

Liberty wants to find a suitable number of segments for its customers, for targeted marketing. Dataset 9.3 shows a list of representative customers.

Dataset 9.3

Customer	No. of transactions	Total Purchase ($)	Income ($ K)
1	5	450	90
2	10	800	82
3	15	900	77
4	2	50	30
5	18	900	60
6	9	200	45
7	14	500	82
8	8	300	22
9	7	250	90
10	9	1000	80
11	1	30	60
12	6	700	80

1. *What is the right number of clusters for Liberty?*
2. *What are their centroids for the clusters?*

10 Association Rule Mining

Learning Objectives

- Understand Association Rule Mining and its business applications
- Learn how association rules are represented
- Know the Apriori algorithm and its pseudocode
- Learn how the association rule technique works in practice

INTRODUCTION

Associate Rule Mining is a popular, unsupervised learning technique, used in businesses to help identify shopping patterns. It is also known as Market Basket Analysis. It helps find interesting relationships (affinities) between variables (items or events). Thus, it can help cross-sell related items and increase the size of a sale.

All data used in this technique is of categorical type. There is no dependent variable. It uses machine learning algorithms. The fascinating relationship between 'sales of diapers and beers' is how it is often explained in popular literature. This technique accepts the raw point-of-sale transaction data as input. The output produced is the description of the most frequent affinities among the items. An example of association rule would be, "a customer who bought flight tickets and hotel reservation also bought a rental car plan 60 percent of the time."

Caselet: Netflix – Data Mining in Entertainment

Netflix suggestions and recommendation engines are powered by a suite of algorithms using data of millions of customer ratings about thousands of movies. Most of these algorithms are based on the premise that similar viewing patterns represent similar user tastes. This suite of algorithms, called CineMatch, instructs Netflix's servers to process information from its databases to determine what movies a customer is likely to enjoy. The algorithm takes into account many factors about the films themselves, the customers' ratings, and the combined ratings of all Netflix users. The company estimates that a whopping 75 percent of viewer activity is driven by recommendations. According to Netflix, these predictions were valid around 75 percent of the time and half of Netflix users who rented CineMatch, recommended movies and gave them a five-star rating.

To make matches, a computer

1. *Searches the CineMatch database for people who have rated the same movie—for example, "The Return of the Jedi".*
2. *Determines which of those people have also rated a second movie, such as "The Matrix".*
3. *Calculates the statistical likelihood that people who liked "Return of the Jedi" will also like "The Matrix".*
4. *Continues this process to establish a pattern of correlations between subscribers' ratings of many different films.*

Netflix launched a contest in 2006 to find an algorithm that could beat CineMatch. The contest, called the Netflix Prize, promised $1 million to the first person or team to meet the accuracy goals for recommending movies based on users' personal preferences. Each of these algorithm submissions was required to demonstrate a 10 percent improvement over CineMatch. Three years later, the $1 million prize was awarded to a team of seven people. (Source: http://electronics.howstuffworks.com*)*

1. *Are Netflix customers being manipulated into seeing what Netflix wants them to see?*
2. *Compare this story with Amazon's personalization engine.*

BUSINESS APPLICATIONS OF ASSOCIATION RULES

In business environments, a pattern or knowledge can be used for many purposes. In sales and marketing, it is used for cross-marketing and cross-selling, catalog design, e-commerce site design, online advertising optimization, product pricing, and sales/promotion configurations. This analysis suggests not to put one item on sale at a time, and instead to create a bundle of products promoted as a package to sell other nonselling items.

In retail environments, it can be used for store design. Strongly associated items can be kept close together for customer convenience. Or they could be placed far from each other so that the customer has to walk the aisles and by doing so is potentially exposed to other items.

In medicine, this technique can be used for relationships between symptoms and illnesses; diagnosis and patient characteristics/treatments; genes and their functions, etc.

REPRESENTING ASSOCIATION RULES

A generic association rule is represented between a set X and Y: $\boldsymbol{X \Rightarrow Y\ [S\%, C\%]}$

$\boldsymbol{X, Y}$ Products and/or services

X Left hand side (LHS)

Y Right hand side (RHS)

S Support – how often ***X*** and ***Y*** go together in the dataset, i.e., $P(\boldsymbol{X} \cup \boldsymbol{Y})$

C Confidence – how often **Y** is found, given ***X***, i.e., $P(\boldsymbol{Y} \mid \boldsymbol{X})$

Example{Hotel booking, Flight booking} ⇒ {Rental Car} [30%, 60%]

[**Note** $P(X)$ is the mathematical representation of the probability or chance of *X* occurring in the dataset]

Computation Example

Suppose there are 1000 transactions in a dataset. There are 300 occurrences of *X* and 150 occurrences of (X, Y) in the dataset.

Support *S* for $\boldsymbol{X} \Rightarrow \boldsymbol{Y}$ will be $P(X \cup Y) = 150/1000 = 15\%$

Confidence for $\boldsymbol{X} \Rightarrow \boldsymbol{Y}$ will be $P(\boldsymbol{Y} \mid \boldsymbol{X})$ or $P(\boldsymbol{X} \cup \boldsymbol{Y})/P(\boldsymbol{X}) = 150/300 = 50\%$

ALGORITHMS FOR ASSOCIATION RULE

Not all association rules are interesting and useful, except those that are strong and occur frequently. In association rule mining, the goal is to find all the rules that satisfy the user-specified *minimum support* and *minimum confidence*. The resulting sets of rules are all the same irrespective of the algorithm used, that is, given a transaction dataset *T*, a minimum support and a minimum confidence, the set of association rules existing in *T* is *uniquely determined*.

Fortunately, there are many algorithms that are available for generating association rules. The most popular algorithms are Apriori, Eclat, FP-Growth, along with various derivatives and hybrids of the three. All the algorithms help identify the frequent itemsets, which are then converted to association rules.

APRIORI ALGORITHM

This is the most popular algorithm used for association rule mining. The objective is to find subsets that are common to at least a minimum number of the itemsets. A frequent itemset is the one whose support is greater than or equal to minimum support threshold. The Apriori property is a downward closure property, which means that any subset of a frequent itemset is also a frequent itemset. Thus, if (A, B, C, D) is a frequent itemset, then any subset such as (A, B, C) or (B, D) is also a frequent itemset.

It uses a bottom-up approach and the size of frequent subsets is gradually increased, from 1-item subsets to 2-item subsets, then 3-item subsets, and so on. Groups of candidates at each level are tested against the data for minimum support.

ASSOCIATION RULES EXERCISE

Dataset 10.1 shows a dozen sales transactions. There are six products being sold—Milk, Bread, Butter, Eggs, Cookies, and Ketchup. Transaction#1 sold Milk, Eggs, Bread and Butter. Transaction#2 sold Milk, Butter, Eggs and Ketchup and so on. The objective is to use this transaction data to find affinities between products, i.e., which products sell together often.

The support level will be set at 33 percent and the confidence level will be set at 50 percent. That means that we have decided to consider rules from only those itemsets that occur at least 33 percent of the time in the total set of transactions. Confidence level means that within those itemsets, the rules of the form $X \rightarrow Y$ should be such that there is at least 50 percent chance of Y occurring based on X occurring.

Dataset 10.1

Transaction List				
1	Milk	Egg	Bread	Butter
2	Milk	Butter	Egg	Ketchup
3	Bread	Butter	Ketchup	
4	Milk	Bread	Butter	
5	Bread	Butter	Cookies	
6	Milk	Bread	Butter	Cookies
7	Milk	Cookies		
8	Milk	Bread	Butter	
9	Bread	Butter	Egg	Cookies
10	Milk	Butter	Bread	
11	Milk	Bread	Butter	
12	Milk	Bread	Cookies	Ketchup

First step is to compute 1-item itemsets, i.e., how often does any product sells individually.

1-itemSets	Frequency
Milk	9
Bread	10
Butter	10
Egg	3
Ketchup	3
Cookies	5

Thus, Milk sells in 9 out of 12 transactions, Bread sells in 10 out of 12 transactions,and so on.

At every point, there is an opportunity to select itemsets of interest, and thus further analysis. Other itemsets that occur infrequently may be removed. If itemsets that occur 4 or more times out of 12 are selected, that corresponds to meeting a minimum support level of 33 percent (4 out of 12). Only 4 items make the cut. The frequent items that meet the support level of 33 percent are

Frequent 1-item Sets	Frequency
Milk	9
Bread	10
Butter	10
Cookies	5

The next step is to go for the next level of itemsets using items selected earlier, i.e., 2-item itemsets.

2-item Sets	Frequency
Milk, Bread	7
Milk, Butter	7
Milk, Cookies	3
Bread, Butter	9
Butter, Cookies	3
Bread, Cookies	4

Thus the sale of (Milk, Bread) is 7 times out of 12, (Milk, Butter) is 7 times, (Bread, Butter) is 9 times, and (Bread, Cookies) is 4 times.

However, only four of these transactions meet the minimum support level of 33 percent.

2-item Sets	Frequency
Milk, Bread	7
Milk, Butter	7
Bread, Butter	9
Bread, Cookies	4

The next step is to list the next higher level of itemsets, i.e., 3-item itemsets.

3-item Sets	Frequency
Milk, Bread, Butter	6
Milk, Bread, Cookies	1
Bread, Butter, Cookies	3

Thus the sale of (Milk, Bread, Butter) is 6 times out of 12 and (Bread, Butter, Cookies) is 3 times out of 12. One 3-item itemset meets the minimum support requirements.

3-item Sets	Frequency
Milk, Bread, Butter	6

There is no room to create a 4-item itemset for this support level.

CREATING ASSOCIATION RULES

The most interesting and complex rules at higher size itemsets start top-down with the most frequent itemsets of higher size-numbers. Association rules are created that meet the support level (>33 percent) and confidence levels (> 50 percent).

The highest level itemset that meets the support requirements is the 3-item itemset. The following itemset has a support level of 50 percent(6 out of 12).

Milk, Bread, Butter	6

This itemset could lead to multiple candidates association rules.
Start with the following rule

(Bread, Butter) → Milk

There are a total of 12 transactions.

X (in this case Bread, Butter) occurs 9 times; *X*, *Y* (in this case Bread, Butter, Milk) occurs 6 times.

The support level for this rule is 6/12 = 50 percent. The confidence level for this rule is 6/9 = 67 percent. This rule meets our thresholds for support (>33 percent) and confidence (>50 percent).

Thus, the first valid association rule from this data is **(Bread, Butter)** → **Milk {*S* = 50%, *C* = 67%}**.

In exactly the same way, other rules can be considered for their validity.

Consider the rule (Milk, Bread) → Butter. Out of total 12 transactions, (Milk, Bread) occurs 7 times and (Milk, Bread, Butter) occurs 6 times.

The support level for this rule is 6/12 = 50 percent. The confidence level for this rule is 6/7 = 86 percent. This rule meets our thresholds for support (>33 percent) and confidence (>50 percent).

Thus, the second valid association rule from this data is
(Milk, Bread) → **Butter {*S* = 50%, *C* = 67%}**.

Consider the rule (Milk, Butter) → Bread. Out of total 12 transactions, (Milk, Butter) occurs 7 times, while (Milk, Butter, Bread) occur 6 times.

The support level for this rule is 6/12 = 50 percent. The confidence level for this rule is 6/7 = 86 percent. This rule meets our thresholds for support (>33 percent) and confidence (>50 percent).

Thus, the next valid association rule is **Milk, Butter** → **Bread{*S* = 50%, *C* = 86%}**.

Thus, there were only three possible rules at the 3-item itemset level and all were found to be valid.

One can get to the next lower level and generate association rules at the 2-item itemset level.
Consider the following rule

> Milk → Bread;out of total 12 transactions, Milk occurs 9 times while (Milk, Bread) occurs 7 times.

The support level for this rule is 7/12 = 58 percent. The confidence level for this rule is 7/9 = 78 percent. This rule meets our thresholds for support (>33 percent) and confidence (>50 percent).

Thus, the next valid association rule is **Milk** → **Bread{58%, 78%}**.

Many such rules could be derived if needed.

Not all such association rules are interesting. The client may be interested in only the top few rules that they want to implement. The number of association rules depends upon business needs. Implementing every rule in business will require some cost and effort, with some potential of gains. The strongest of rules, with the higher support and confidence rates, should be used first, and the others should be progressively implemented later.

Conclusion

Association rules help discover affinities between products in transactions. It helps make cross-selling recommendations much more targeted and effective. Apriori technique is the most popular technique and it is a machine learning technique.

Review Questions

1. What are association rules? How do they help?
2. How many association rules should be used?
3. What are frequent itemsets?
4. How does the Apriori algorithm work?

True/False

1. Also known as Market Basket Analysis, Association Rule Mining is a machine learning algorithm.
2. Association rule mining is an unsupervised learning technique that helps find frequent patterns.
3. Not all association rules are interesting. The client may be interested in implementing only the top few rules.
4. All the data used in association rules technique is of ratio (numeric) type.
5. Given a transaction dataset T, a minimum support and a minimum confidence, the goal is to determine a set of rules that meet those support and confidence conditions.
6. The set of association rules existing in T depends upon the algorithm used.
7. A generic association rule is represented between sets X and Y as $\boldsymbol{X \rightarrow Y}$ **[$\boldsymbol{S}$%, $\boldsymbol{C}$%]**.
8. A frequent itemset can contain any number of items.

9. Apriori is the name of the most popular association rule mining technique.

10. A suite of algorithms called CineMatch helps Netflix determine which movies a customer is likely to enjoy next.

Liberty Stores Case Exercise: Step 8

Dataset 10.2 shows a list of transactions from Liberty's stores. Create association rules for the following data with 33 percent support level and 66 percent confidence.

Dataset 10.2

	Barley	Corn	Gram	Millet	Rice	Wheat
1	1	1	1	1	1	1
2		1		1	1	1
3	1		1		1	1
4		1	1		1	1
5	1		1	1	1	
6			1		1	1
7	1			1	1	1
8				1	1	1
9	1	1	1	1		
10	1	1	1		1	1
11		1		1	1	1
12	1	1		1	1	1

https://hemanthrajhemu.github.io