CSC12107 – Information Systems for Business Intelligence

Chapter 6
BI - mining

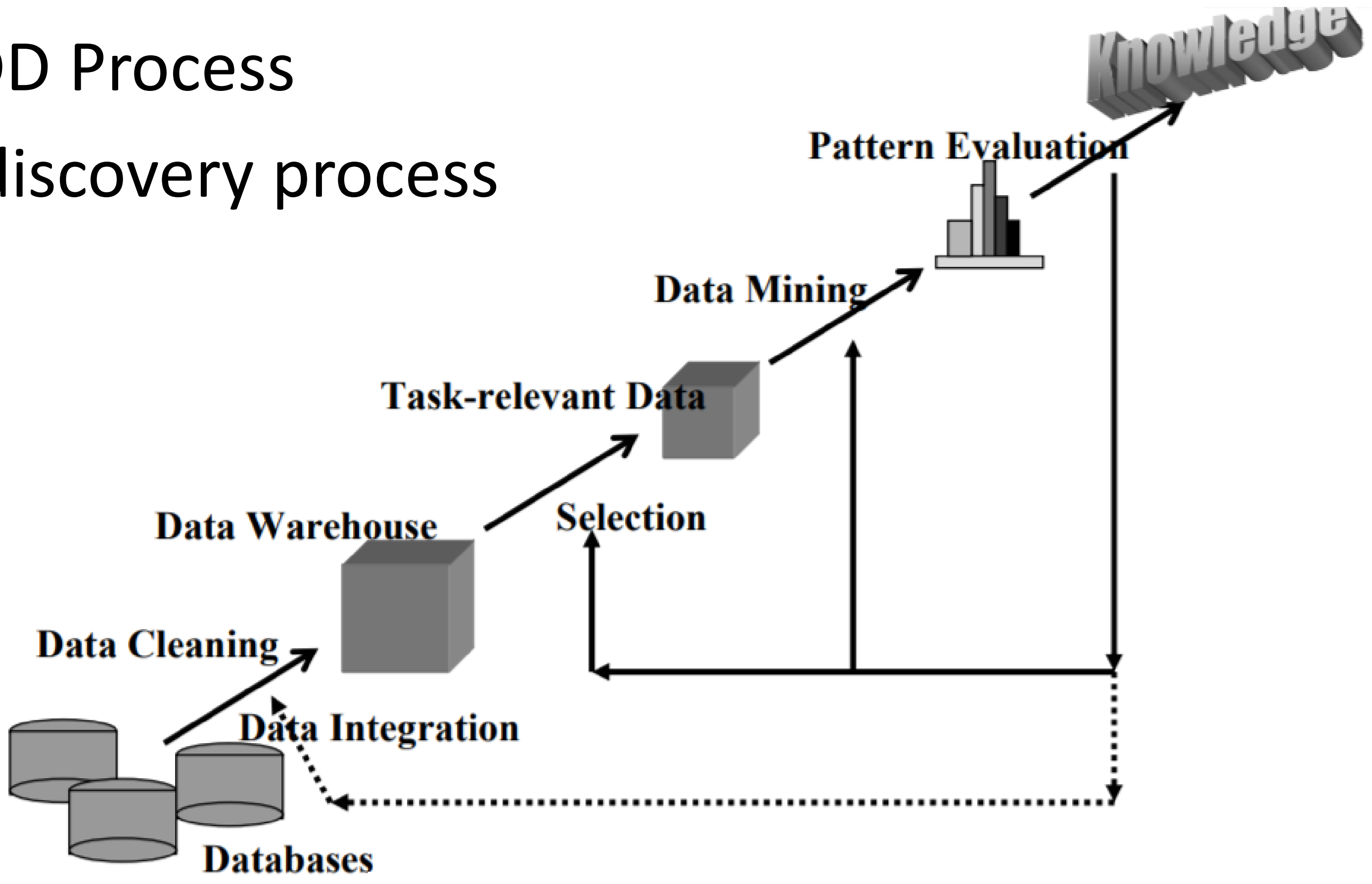
KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Learning Objectives

- Define data mining as an enabling technology for business intelligence
- Describe the objectives and benefits of business analytics and data mining
- Describe some algorithms that are applied to some specific senarios
- Design and implement an integrated data mining solution by using SQL Server Analysis Services

Introduction to data mining

- Data Mining as a step in A KDD Process
- The core step of knowledge discovery process



Introduction to Data mining

- **Data Mining: Concepts and Techniques**
  - *"Data mining, also popularly referred to as* ***knowledge discovery*** *from data (KDD), is the automated or convenient extraction of patterns representing knowledge implicitly stored or captured in large databases, data warehouses, the Web, other massive information repositories or data streams."*
- **Data Mining: Practical Machine Learning Tools and Techniques**
  - *"Data mining is defined as the process of* ***discovering patterns*** *in data. The process must be automatic or (more usually) semiautomatic. The patterns discovered must be meaningful in that hey lead to some advantage, usually an economic one. The data is invariably present in substantial quantities."*

Introduction to Data mining

- Data mining: what for?
  - To look for interesting structures such as:
    - Patterns from statistics
    - Predictive models
    - Hidden relationship

Introduction to data mining

- **Patterns**: Valid, Novel, Potentially useful, Understandable to the users.
- **Types of patterns**
  - Association
  - Prediction
  - Cluster (segmentation)
  - Sequential (or time series) relationships

Introduction to data mining

- These patterns and trends can be collected and defined as a ***data mining model***. Mining models can be applied to specific scenarios, such as:
  - Banking: loan/credit card approval
    - predict good customers based on old customers
  - Customer relationship management:
    - identify those who are likely to leave for a competitor.
  - Targeted marketing:
    - identify likely responders to promotions
  - Fraud detection: telecommunications, financial transactions
    - from an online stream of event identify fraudulent events

Introduction to data mining

Table 1.1 Contact Lens Data

Age	Spectacle Prescription	Astigmatism	Tear Production Rate	Recommended Lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none

- **A set of rules learned from this information**
  - **If** tear production rate = reduced **then** recommendation = none
  - **if** age = young and astigmatic = no **then** recommendation = soft

Introduction to data mining

- **Attribute** (or dimension, feature, variable) is a data field, representing a characteristic or a feature of a data object.
- A collection of attributes describe an object

attributes

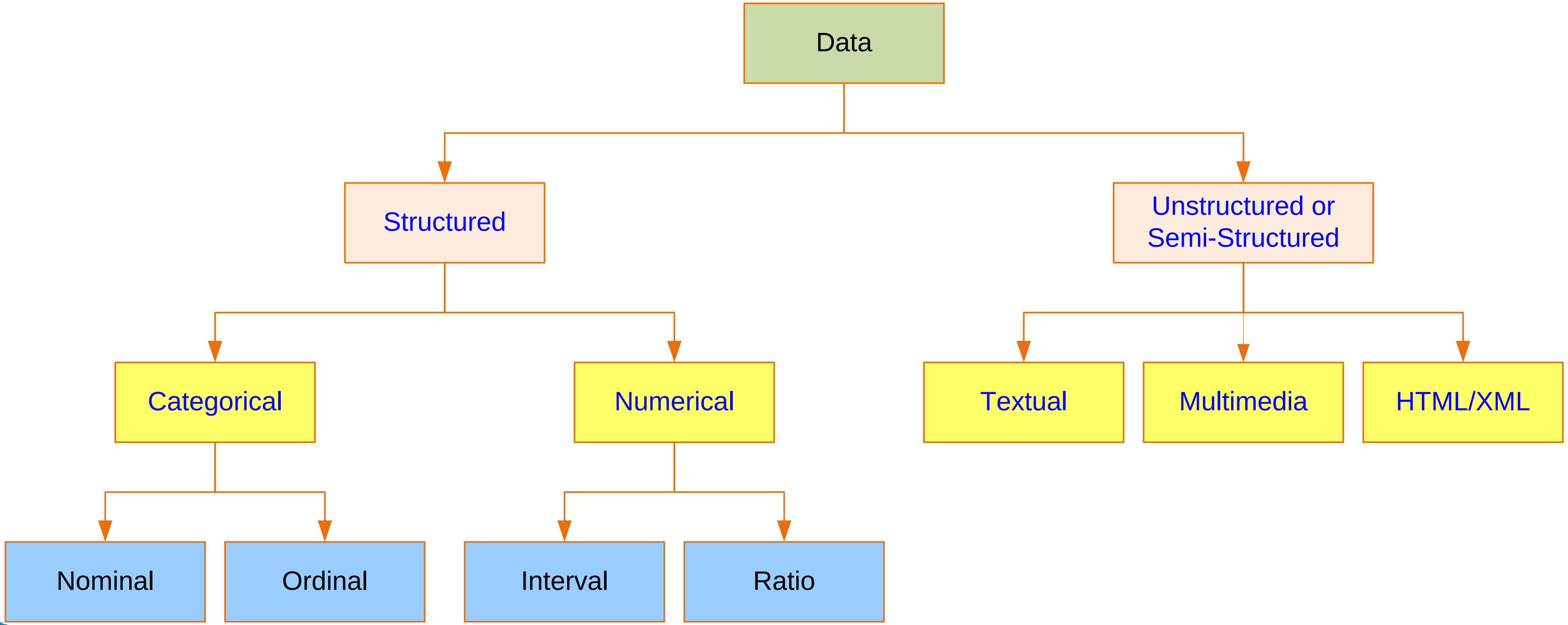
object

Table 1.1 Contact Lens Data

Age	Spectacle Prescription	Astigmatism	Tear Production Rate	Recommended Lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none

Data in Data mining

- Data may consist of numbers, words, images, …



Data in data mining

- **Nominal** are used to label variables without any quantitative value (categories, state, name of things…)
    - Hair_color = {black, brown, blond, red, grey, white}
    - marital status, occupation, ID numbers, zip codes
- **Binary**
    - Nominal attribute with only 2 states (0 and 1)
    - Symmetric binary: both outcomes equally important
    - e.g., gender
- **Ordinal**
    - Values have a meaningful order (ranking) but magnitude between successive values is not known.
    - Size = {small, medium, large}, socio economic status (“low income”,”middle income”,”high income”)

Data in data mining

- **Interval scales**
  - are numeric scales
  - know both the order and the exact differences between the values.
  - don’t have a “true zero.”
    - Ex: Celsius temperature, zero doesn’t mean the absence of value, 20 degrees C is not twice as hot as 10 degrees C
- **Ratio**
  - have a clear definition of zero
  - can be meaningfully added, subtracted, multiplied, divided (ratios)
  - Ex: weight, height

Data in data mining

- **Discrete Attribute**
  - Has only a finite or countably infinite set of values
  - Often represented as integer variables
- **Continuous Attribute**
  - Has real numbers as attribute values
  - Examples: . height, weight, length, temperature and speed

https://link.springer.com/chapter/10.1007%2F978-1-84628-766-4_7
https://www.geeksforgeeks.org/understanding-data-attribute-types-qualitative-and-quantitative/

Data in data mining

Provides:	Nominal	Ordinal	Interval	Ratio
The "order" of values is known		✔	✔	✔
"Counts," aka "Frequency of Distribution"	✔	✔	✔	✔
Mode	✔	✔	✔	✔
Median		✔	✔	✔
Mean			✔	✔
Can quantify the difference between each value			✔	✔
Can add or subtract values			✔	✔
Can multiple and divide values				✔
Has "true zero"				✔

https://www.mymarketresearchmethods.com/types-of-data-nominal-ordinal-interval-ratio/

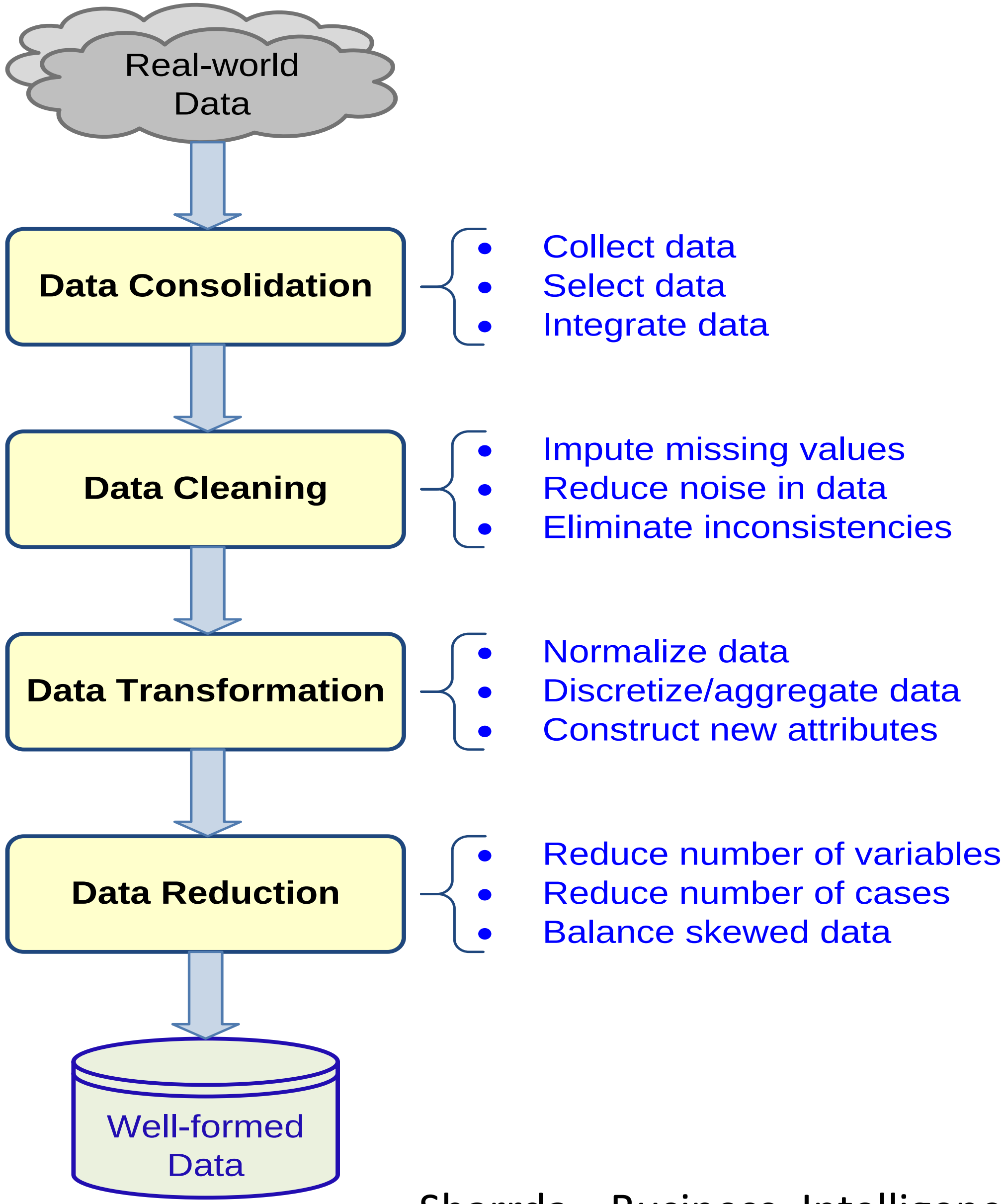
Data mining process - CRISP-DM

Step 1: **Business Understanding**	Set goals for the project Using business objectives and current scenario, define your data mining goals
Step 2: **Data Understanding**	Set the data and data source Check if the available data can meet the objectives of the project and establish how you will meet the objectives
Step 3: **Data Preparation**	The data from different sources should be selected, cleaned, transformed, formatted, anonymized, and constructed Data cleaning & transformation (smoothing noisy data and filling in missing values, aggregation, normalization…)
Step 4: **Model Building**	Execute the algorithms that satisfíes the project objectives Create a scenario to test check the quality and validity of the model. Run the model on the prepared dataset. Results should be assessed by all stakeholders
Step 5: **Testing and Evaluation**	
Step 6: **Deployment**	

Accounts for ~85% of total project time

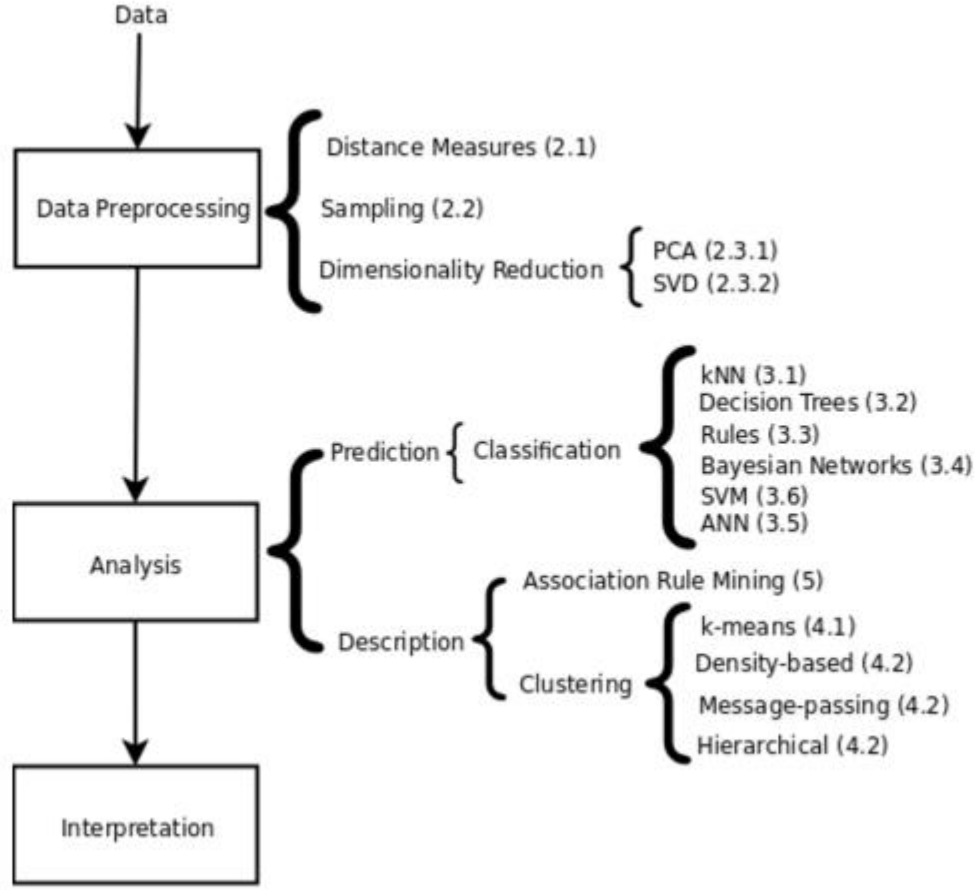
https://towardsdatascience.com/crisp-dm-methodology-leader-in-data-mining-and-big-data-467efd3d3781

Data Preparation



Sharrda, Business Intelligence,3rd

Data mining tasks



Recommender Systems Handbook. © Springer Science+Business Media, LLC 2011

Data Mining	Learning Method	Popular Algorithms
Prediction	Supervised	Classification and Regression Trees, ANN, SVM, Genetic Algorithms
Classification	Supervised	Decision trees, ANN/MLP, SVM, Rough sets, Genetic Algorithms
Regression	Supervised	Linear/Nonlinear Regression, Regression trees, ANN/MLP, SVM
Association	Unsupervised	Apriory, OneR, ZeroR, Eclat
Link analysis	Unsupervised	Expectation Maximization, Apriory Algorithm, Graph-based Matching
Sequence analysis	Unsupervised	Apriory Algorithm, FP-Growth technique
Clustering	Unsupervised	K-means, ANN/SOM
Outlier analysis	Unsupervised	K-means, Expectation Maximization (EM)

Sharrda, Business Intelligence,3rd

Suppervised vs unsuppervised

- **Supperviced learning**
  - **All data is labeled** and the algorithms learn to predict the output from the input data.
  - Given input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output: Y = f(X)
- **Unsupervised learning**
  - is where you only have input data (X) and no corresponding output variables.
  - is to model the underlying structure or distribution in the data in order to learn more about the data
  - there is no correct answers and there is no teacher.
  - **All data is unlabeled** and the algorithms learn to inherent structure from the input data.

Classic application

- **"Market basket" data**
  - Purchase(salesID, item)
  - (3, bread)
  - (3, milk)
  - (3, eggs)
  - (3, beer)
  - (4, beer)
  - (4, chips)
  - ....

- **Want to find association rules**:
  {L1,L2,...,Ln} -> R
- **Diễn giải**: "If a customer bought all the items in set {L1, L2, ..., Ln}, he is very likely to also have bought item R"
- **Ex:**
  {bread, milk} -> eggs
  {diapers} -> beer

- Goals of data mining: Quickly find association rules over extremely large data sets (ex: all Wal-Mart sales for a year).)

Classic application

- **Classification trees** (= decision trees)
  - **Buyers(<attributes>, purchase)**
  - Want to predict purchase from <attributes>
- **Clustering**
  - **Buyers(<attributes>)**
  - Automatically group buyers into N similar types
- **Top-N items**
  - **Purchase(salesID, item)**
  - What were the N most often purchased items? (salesID irrelevant)

DM - Classification

- Most frequently used DM method
- Employ supervised learning
- Learn from past data, classify new data
- The output variable is categorical (nominal or ordinal) in nature
- Predicts categorical class labels (discrete or nominal)
- Use labels of the training data to classify new data
- There is a lot of classification algorithms available: Decision Trees:
  - Bayesian Classifiers, Neural Networks, K-Nearest Neighbour, Support Vector Machines, Linear Regression

DM - Classification

- **Example:**

Customer profile

- ❑ A model or classifier is contsructed to predict categorical labels such as {hard, soft, none } for a recommendation lense application.
- ❑ A bank loan officer wants to analyze the data in order to know which customer (loan applicant) are risky or which are safe.

How Does Classification Works

1. **All data is labeled** and the algorithms learn to predict the output from the input data.
2. **Learning Step (Training Phase):** Construction of Classification Model
   - Given input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output: Y = f(X)
   - Different Algorithms are used to build a classifier by making the model learn using the training set available
3. **Classification Step**
   - Model used to predict class labels and testing the constructed model on test data
   - Given an unlabeled observation X, the predict(X) returns the predicted label y.
4. **Evaluate** the classifier model

How Does Classification Works

- **The weather problem**
  - Supposedly the weather concerns the conditions that are suitable for playing some unspecified game
  - ➔ when there has a new case
  - ➔ measure these variables to predict whether to play/not

Outlook	Temperature	Humidity	Windy	Play
Sunny	hot	high	false	??

to predict whether to play or not

Outlook	Temperature	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

Use the variables (outlook, temperature, humidity, windy)

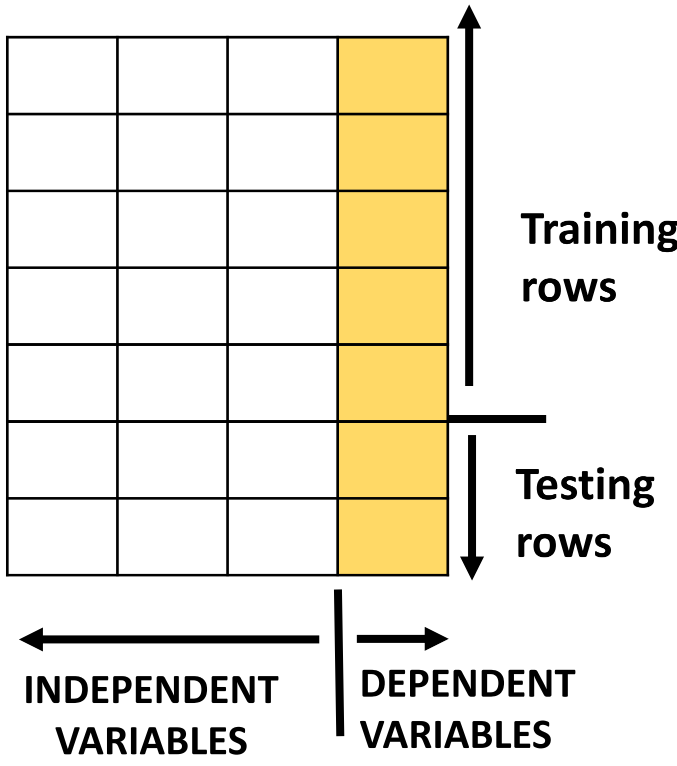
How Does Classification Works

The weather problem

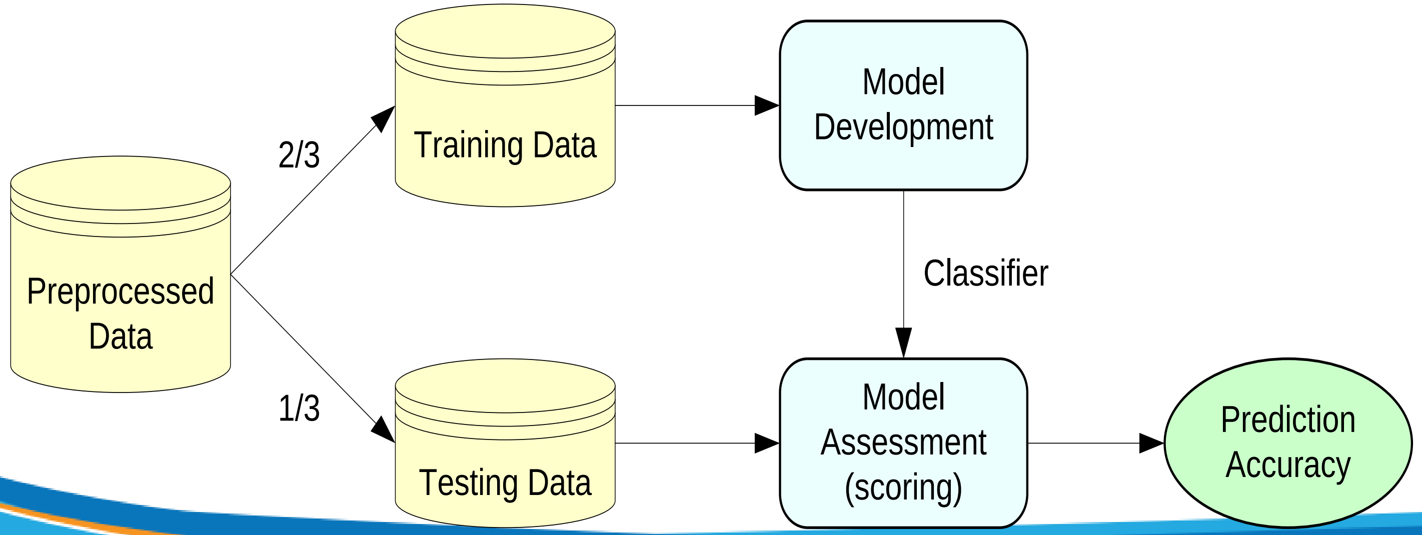
1. <u>Labeled data</u>:

- Attribute/feature:
  - Outlook { sunny, overcast, rainy}
  - Temperature { hot, mild, cool}
  - Humidity { high, normal}
  - Windy { true, false}
- Attribute values: symbolic categories
- The outcome is: play or not play
  - A predefine class label is assigned to every sample tuple or object

How Does Classification Works



- **Learning Step (Training Phase):**
  - Randomly split the loaded dataset into two (70%-30%)
  - Perform the model training on the training set
  - Use the test set for validation purpose
  - Model usage



How Does Classification Works

- **Learning Step (Training Phase):**
  - **Randomly split the loaded dataset**

Outlook	Temperature	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

Training rows

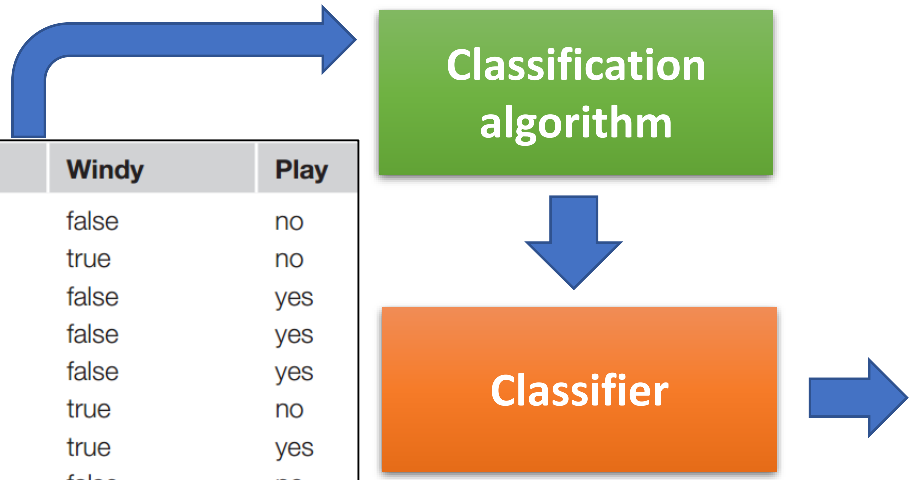
Testing rows

How Does Classification Works

- **Learning Step (Training Phase):**
  - **Perform the model training on the training set**

Training data

Outlook	Temperature	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



A set of rules learned from this information
- If outlook = sunny and humidity = high then play = no
- If outlook = rainy and windy = true then play = no
- If outlook = overcast then play = yes
- ....

How Does Classification Works

- **Learning Step (Training Phase):**
  - **Use the test set for validation purpose**

Test Classifier → **Accuracy**

Testing data

Outlook	Temperature	Humidity	Windy	Play
Sunny	mild	normal	true	??
Overcast	mild	high	true	??
Overcast	hot	normal	false	??
Rainy	mild	high	true	??

Hide the outcome in the testing data

Outlook	Temperature	Humidity	Windy	Play
Sunny	mild	normal	true	yes
Overcast	mild	high	true	yes
Overcast	hot	normal	false	yes
Rainy	mild	high	true	no

Real outcome

How Does Classification Works

- **Learning Step (Training Phase):**
  - **Use the test set for validation purpose**
  - **If the accuracy is acceptable:**

New data (Unlabeled data)

Outlook	Temperature	Humidity	Windy	Play
Overcast	**Mild**	**High**	**False**	**??**
Overcast	Cold	Normal	True	??

Classifier

Outlook	Temperature	Humidity	Windy	Play
Overcast	**Mild**	**High**	**False**	**yes**
Overcast	Cold	Normal	True	no

How Does Classification Works

Assessment Methods

- To predict the performance of a classifier on new data, we need to assess its **error rate** on a dataset that played no part in the formation of the classifier → test set (independent dataset)
- The test data is not used in any way to create the classifier.
  - If the class prediction is correct → SuccessCount ++
  - if not, it is an error → ErrorCount++
- The error rate = the proportion of errors made over a whole set of instances
- Understanding the accuracy of your model is invaluable because you can begin to tune the parameters of your model to increase its performance.

How Does Classification Works

- **Assessment Methods (cont)**
- In classification problems, the primary source for accuracy estimation is the confusion matrix

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive Count (TP)	False Positive Count (FP)
	Negative	False Negative Count (FN)	True Negative Count (TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

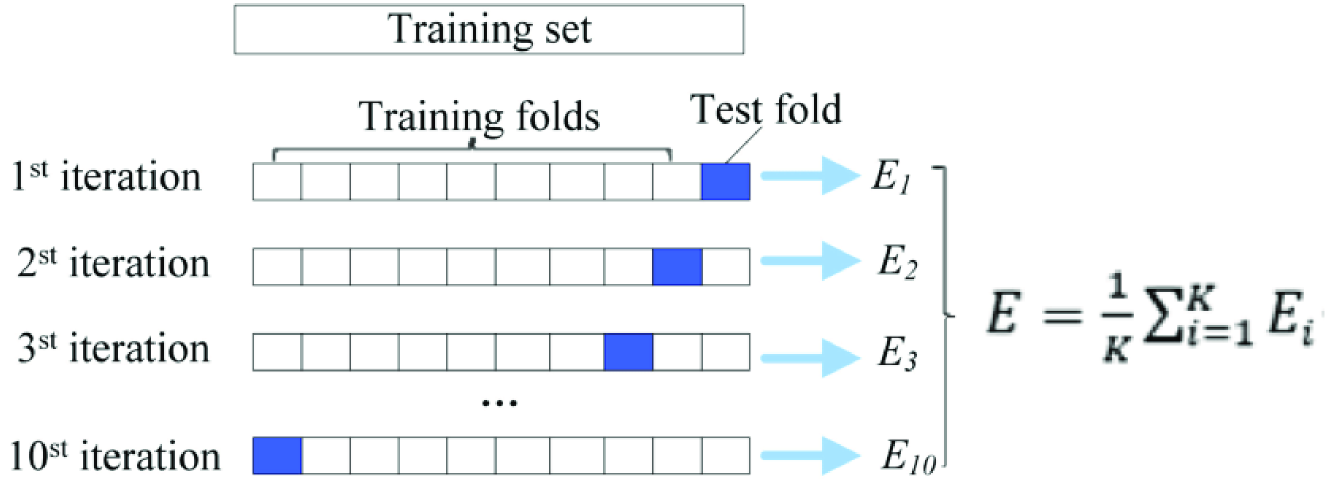
$$True\ Positive\ Rate = \frac{TP}{TP + FN}$$

$$True\ Negative\ Rate = \frac{TN}{TN + FP}$$

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

How Does Classification Works

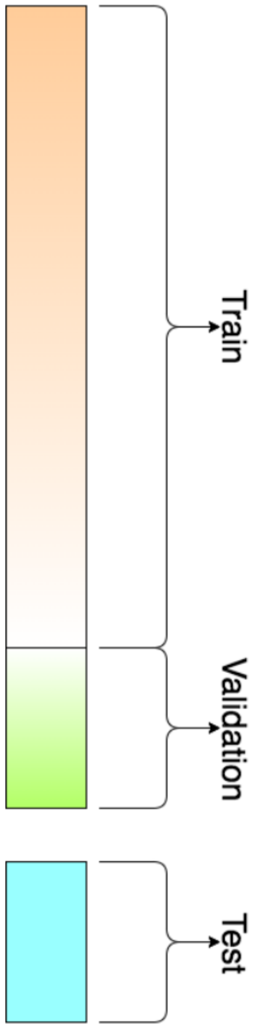
- *k*-Fold Cross validation
  - Split the data into *k* mutually exclusive subsets
  - Use each subset as testing while using the rest of the subsets as training
  - Repeat the experimentation for *k* times
  - Aggregate the test results for true estimation of prediction accuracy training



https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623

How Does Classification Works

- **<u>Training set:</u>**
  - A set of examples used for learning, that is to fit the parameters of the classifier.
- **<u>Validation set</u>**:
  - A set of examples used to tune the parameters of a classifier
- **<u>Test set</u>**:
  - A set of examples used only to assess the performance of a fully-specified classifier.
- *if you have a model with no hyperparameters or ones that cannot be easily tuned, you probably don't need a validation set too!*



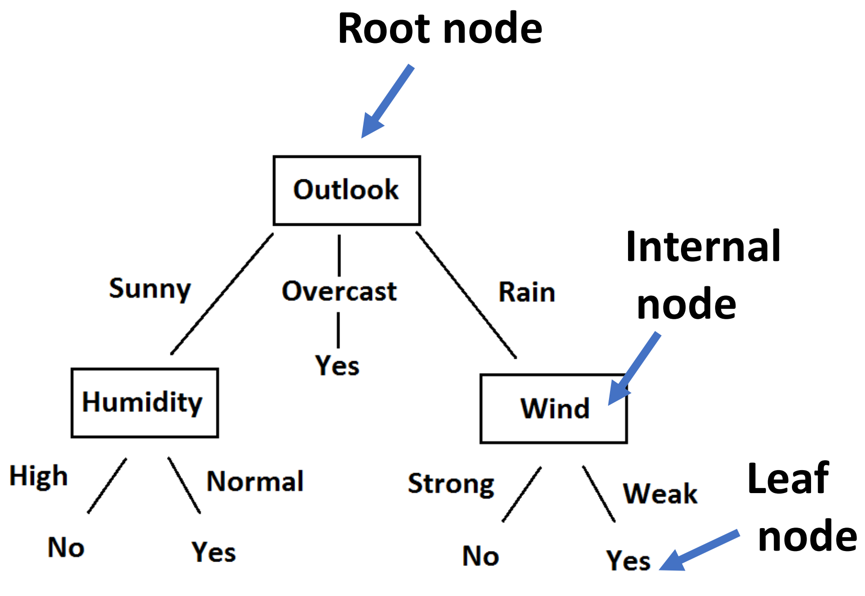
https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7

Classification vs Prediction

- What is prediction?
  - **Classification** models predict **categorical** class labels
  - **prediction** models predict **continuous** valued functions
- **Ex:**
  - Suppose the marketing manager needs to predict how much a given customer will spend during a sale at his company. → **prediction**
  - A bank loan officer wants to analyze the data in order to know which customer (loan applicant) are risky or which are safe → **classification**
  - A marketing manager at a company needs to analyze a customer with a given profile, who will buy a new computer. → **classification**

Classification - **Decision Tree**

- **Definition**:
  - Employs the divide and conquer method
  - Recursively divides a training set until each division consists of examples from one class
  - **Type of node**:
    - **Root node** is an attribute to place at the root node
    - **Internal nodes** (non leaf node) denotes a test on an attribute
    - **Leaf nodes** (terminal nodes) hold class labels



How are decision trees used for classification?

Classification - **Decision Tree**

- **Attribute:**
  - **Input** (indepent variables): **v** feature/attribute = $X_1; X_2; :::; X_v$
    - Each X_j has domain $\mathbf{O_j}$:
    - Category: {high, cold}
    - Numerical: {0,1}
  - **Output** (dependent variable) /class: **C** with domain **Oy**
    - Category: classification
    - Numerical: Regression
- Given a dataset **D**, n row:
  - n example (X_i, C_i); X_i: is a v-dim feature vector
  - $C_i \in$ **Oy** is output variable
- Task:
  - Given an input data vector $\boldsymbol{x}$ predict C

Classification - **Decision Tree**

Idea:

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

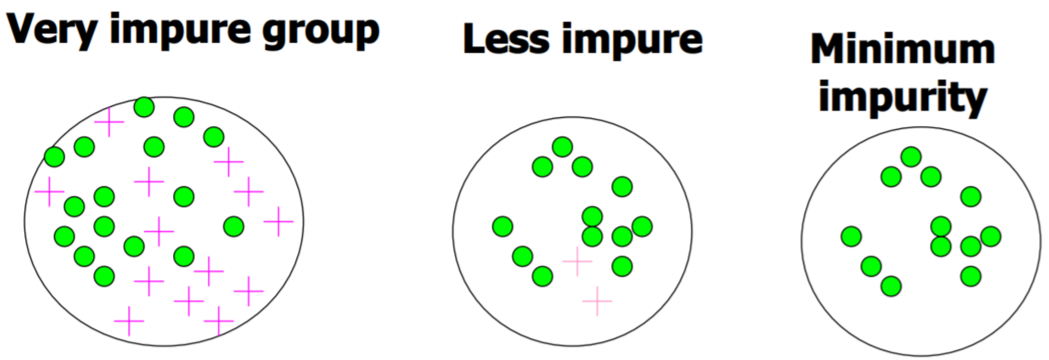
Classification - **Decision Tree**

- **Mainly problems:**
  1. **Splitting criteria**
     - Which variable, what value, etc.
  2. **Stopping criteria**
     - When to stop building the tree
  3. **Pruning** (generalization method)
     - Pre-pruning versus post-pruning
- Most popular DT algorithms include
  - ID3, C4.5, C5; CART; CHAID; M5

Classification - **Decision Tree**

- **Alternative splitting criteria**
  - **Gini index** determines the purity of a specific class as a result of a decision to branch along a particular attribute/value
    - Used in CART
  - **Information gain** uses entropy to measure the extent of uncertainty or randomness of a particular attribute/value split
    - Used in ID3, C4.5, C5

Classification - **Decision Tree**



- **Impurity/Entropy (informal)**
  - Measures the level of impurity in a group of examples
  - **Entropy**: a common way to measure impurity
  - The expected information needed to classify a tuple in D is given
  - by:

$$\text{Info (D)} = -\sum_{i=1}^{m} p_i \, log_2\,(pi) \quad ; p_i = \frac{|Ci_{\,D}|}{|D|}$$

- m: the number of classes
- pi: the probability that an arbitrary tuple in D belongs to class Ci estimated by: |Ci,D|/|D|(proportion of tuples of each class)
- Entropy comes from information theory. The higher the entropy the more the information content.

https://homes.cs.washington.edu/~shapiro/EE596/notes/InfoGain.pdf

Decision Tree - Classification

- Example

id	outlook	temperature	humidity	wind	play
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
3	overcast	hot	high	weak	yes
4	rainy	mild	high	weak	yes
5	rainy	cool	normal	weak	yes
6	rainy	cool	normal	strong	no
7	overcast	cool	normal	strong	yes
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
10	rainy	mild	normal	weak	yes
11	sunny	mild	normal	strong	yes
12	overcast	mild	high	strong	yes
13	overcast	hot	normal	weak	yes
14	rainy	mild	high	strong	no

Data Mining: Practical Machine Learning Tools and Techniques – p11

Information gain

- Training set D
  - Attribute X = $\{x_1, x_2, \ldots, x_v\}$
    - Outlook = {Sunny, Overcast, Rainy}
    - Temperature = {Hot, mild, cool}
    - Humidity = {high, normal}
    - Windy = {true, false}
  - Gain (Outlook)?
  - Gain (Temperature)?
  - Gain (Humidity)?
  - Gain (Windy)?

Attribute X

What feature should be used?

What values?

$$\mathrm{Info}_{\mathrm{X}}\,(\mathrm{D}) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|}\, I\,(D_{\mathrm{j}})$$

$$\mathrm{Gain}\,(\mathrm{X}) = \mathrm{Info}\,(\mathrm{D}) - \mathrm{Info}_{\mathrm{A}}\,(\mathrm{D})$$

Example: training set

- **Information gain**
  - 14 tuples: 9 yes (play tennis); 5 No
  - $|D| = 14$
  - $m = 2$
  - C_1 = “Yes”; C_2 = “No”
  - $|C_{1,D}| = 9$; $|C_{2,D}| = 5$

$$\text{Info (D)} = I(9,5) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.94$$

Information Gain

Outlook	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Sunny	2	3	0.971
Overcast	4	0	0
Rainy	3	2	0.971

$$I\,(2,3) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.971$$

$$I\,(4,0) = -\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4} = 0$$

$$I\,(3,2) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.971$$

id	outlook	temperature	humidity	wind	play
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
11	sunny	mild	normal	strong	Yes
3	overcast	hot	high	weak	yes
7	overcast	cool	normal	strong	yes
12	overcast	mild	high	strong	yes
13	overcast	hot	normal	weak	yes
4	rainy	mild	high	weak	yes
5	rainy	cool	normal	weak	yes
6	rainy	cool	normal	strong	no
10	rainy	mild	normal	weak	yes
14	rainy	mild	high	strong	no

$$\mathbf{Info_{Outlook}(D)} = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.693$$

$$\mathbf{Gain(outlook)} = Info(D) - InfoOu_{tlook}(D)$$

$= 0.94$ - 0.693

= 0.25

Information Gain

Temperature	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Hot	2	2	1
mild	4	2	0.9183
cool	3	1	0.811278

$$I\,(2,2) = -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4} = 1$$

$$I\,(4,2) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6} = 0.9183$$

$$I\,(3,1) = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} = 0.811278$$

id	outlook	temperature	humidity	wind	play
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
3	overcast	hot	high	weak	yes
13	overcast	hot	normal	weak	yes
4	rainy	mild	high	weak	yes
8	sunny	mild	high	weak	no
10	rainy	mild	normal	weak	yes
11	sunny	mild	normal	strong	yes
12	overcast	mild	high	strong	yes
14	rainy	mild	high	strong	no
5	rainy	cool	normal	weak	yes
6	rainy	cool	normal	strong	no
7	overcast	cool	normal	strong	yes
9	sunny	cool	normal	weak	yes

$$\mathrm{Info}_{\text{Temperature}}(D) = \frac{4}{14}I(2,2) + \frac{6}{14}I(4,2) + \frac{4}{14}I(3,1) = 0.911$$

$$\mathrm{Gain}(\textbf{Temperature}) = Info(D) - Info_{\text{Temperature}}(D)$$

$= 0.94 - 0.911$

$= 0.029$

Information Gain

Humidity	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
High	3	4	0.985
Normal	6	1	0.592

$$I\,(3,4) = -\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} = 0.985$$

$$I\,(6,1) = -\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7} = 0.592$$

id	outlook	temperature	humidity	wind	play
1	sunny	hot	**high**	weak	no
2	sunny	hot	**high**	strong	no
3	overcast	hot	**high**	weak	**yes**
4	rainy	mild	**high**	weak	**yes**
8	sunny	mild	**high**	weak	no
12	overcast	mild	**high**	strong	**yes**
14	rainy	mild	**high**	strong	no
5	rainy	cool	**normal**	weak	yes
10	rainy	mild	**normal**	weak	yes
9	sunny	cool	**normal**	weak	yes
11	sunny	mild	**normal**	strong	yes
6	rainy	cool	**normal**	strong	no
7	overcast	cool	**normal**	strong	yes
13	overcast	hot	**normal**	weak	yes

$$\mathrm{Info}_{\mathrm{Humidity}}(\mathrm{D}) = \frac{7}{14}\,\mathrm{I}(3,4) + \frac{7}{14}\,\mathrm{I}(6,1) = 0.78845$$

$$\mathrm{Gain}(\mathrm{Humidity}) = Info(D) - Info_{\mathrm{Humidity}}(D)$$

= 0.94 - 0.78845

= 0.152

Information Gain

Windy	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Weak	6	2	0.811
strong	3	3	1

$$I\,(6,2) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8} = 0.81128$$

$$I\,(3,3) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6} = 1$$

id	outlook	temperature	humidity	wind	play
1	sunny	hot	high	**weak**	no
8	sunny	mild	high	**weak**	no
5	rainy	cool	normal	**weak**	yes
3	overcast	hot	high	**weak**	yes
13	overcast	hot	normal	**weak**	yes
4	rainy	mild	high	**weak**	yes
10	rainy	mild	normal	**weak**	yes
9	sunny	cool	normal	**weak**	yes
2	sunny	hot	high	**strong**	no
11	sunny	mild	normal	**strong**	yes
12	overcast	mild	high	**strong**	yes
14	rainy	mild	high	**strong**	No
6	rainy	cool	normal	**strong**	no
7	overcast	cool	normal	**strong**	yes

$$\mathrm{Info}_{\mathrm{Windy}}(D) = \frac{8}{14}\,I(6,2) + \frac{6}{14}\,I(3,3) = 0.892$$

$$\mathrm{Gain(Windy)} = Info(D) - InfoWi_{ndy}(D)$$

$= 0.94 - 0.892$

$= 0.048$

Information Gain

Outlook	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Sunny	1	3	0.673
Overcast	2	0	0
Rainy	3	1	0.673
		Gain(Outlook)	**0.17**

Temperature	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Hot	2	2	1
mild	4	2	0.9183
Cool	3	1	0.811278
		Gain (**Temperature)**	**0.029**

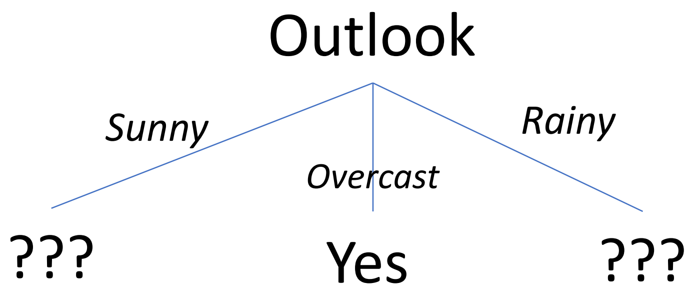
Humidity	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
High	3	4	0.985
Normal	6	1	0.592
		Gain (Humidity)	**0.152**

Windy	C_{1j}: Yes	C_{2j}: No	$I(C_{1j},C_{2j})$
Weak	6	2	0.811
strong	3	3	1
		Gain (Windy)	**0.048**

- → Choose attribute with the largest information gain as the decision node: **0.17 OutLook**

Information Gain

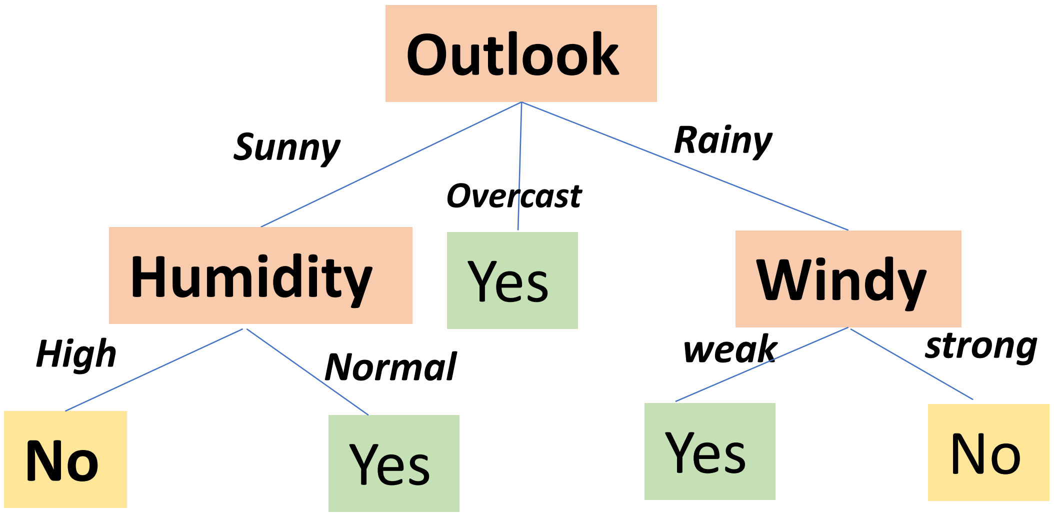
id	outlook	temperature	humidity	wind	play
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
11	sunny	mild	normal	strong	Yes
3	overcast	hot	high	weak	**yes**
7	overcast	cool	normal	strong	**yes**
12	overcast	mild	high	strong	**yes**
13	overcast	hot	normal	weak	**yes**
4	rainy	mild	high	weak	yes
5	rainy	cool	normal	weak	yes
6	rainy	cool	normal	strong	no
10	rainy	mild	normal	weak	yes
14	rainy	mild	high	strong	no



id	temperature	humidity	wind	play
1	hot	**high**	weak	**no**
2	hot	**high**	strong	**no**
8	mild	**high**	weak	**no**
9	cool	**normal**	weak	**yes**
11	mild	**normal**	strong	**Yes**

id	temperature	humidity	wind	play
4	mild	high	weak	yes
5	cool	normal	weak	yes
6	cool	normal	strong	no
10	mild	normal	weak	yes
14	mild	high	strong	no

Information Gain



- R1: if outlook = overcast then yes
- R2: if outlook = Sunny and humidity = High then No
- R3: if outlook = Sunny and humidity = Normal then yes
- R4: if outlook = rainy and windy = weak then yes
- R4: if outlook = rainy and windy = strong then no

Classification - **Decision Tree**

- **Example**: Bike buyer prediction using AdventureWork database
- **<u>Purpose</u>**:
  - Creating a classification model that predicts whether or not a customer will purchase a bike
  - The model should predict bike purchasing for new customers for whom no information about average monthly spend or previous bike purchases is available

https://www.kaggle.com/rahulsah06/bike-buying-prediction-for-adventure-works-cycles?select=AW_BikeBuyer.csv

Classification - **Decision Tree**

- Bike buyer prediction using AdventureWork database
- **Data:**
  - The class were described as either ‘1: Yes’ or ‘0:No’ on the basis of bike buyer or not.
  - The detailed description of the dataset is shown in Table:

Target

	CustomerKey	MaritalStatus	Gender	YearlyIncome	EnglishOccupation	HouseOwnerFlag	CommuteDistance	Age	BikeBuyer
9	11008	S	F	60000.00	Professional	1	10+ Miles	52	1
10	11009	S	M	70000.00	Professional	0	5-10 Miles	52	1
11	11010	S	F	70000.00	Professional	0	5-10 Miles	52	1
12	11011	M	M	60000.00	Professional	1	10+ Miles	52	1
13	11012	M	F	100000.00	Management	1	1-2 Miles	48	0
14	11013	M	M	100000.00	Management	1	0-1 Miles	48	0
15	11014	S	F	100000.00	Management	0	1-2 Miles	48	0
16	11015	S	F	30000.00	Skilled Manual	0	5-10 Miles	37	1
17	11016	M	M	30000.00	Skilled Manual	1	5-10 Miles	37	1
18	11017	S	F	20000.00	Skilled Manual	1	5-10 Miles	72	1
19	11018	S	M	30000.00	Clerical	1	5-10 Miles	71	1

Cluster Analysis for Data Mining

- Clustering techniques apply when there is no class to be predicted but the instances are to be divided into natural groups
- Clustering is dividing data points into homogeneous classes or clusters:
  - Points in the same group are as similar as possible
  - Points in different group are as dissimilar as possible
  - Used for automatic identification of natural groupings of things
  - Part of the machine-learning family
  - Employ unsupervised learning
  - Learns the clusters of things from past data, then assigns new instances

Cluster Analysis for Data Mining

- Clustering results may be used to
  - Identify natural groupings of customers
  - Identify rules for assigning new cases to classes for targeting/diagnostic purposes
  - Provide characterization, definition, labeling of populations
  - Decrease the size and complexity of problems for other data mining methods
  - Identify outliers in a specific domain (e.g., rare-event detection)

K-mean

- *k*-Means Clustering Algorithm
  - *k* : pre-determined number of clusters
  - Algorithm (Step 0: determine value of *k*)

  Step 1: Randomly generate *k* random points as initial cluster centers.

  Step 2: Assign each point to the nearest cluster center.

  Step 3: Re-compute the new cluster centers.

  Repetition step: Repeat steps 3 and 4 until some convergence criterion is met (usually that the assignment of points to clusters becomes stable).

K-mean

The decision of merging two clusters is taken on the basis of closeness of these clusters. There are multiple metrics for deciding the closeness of two clusters :

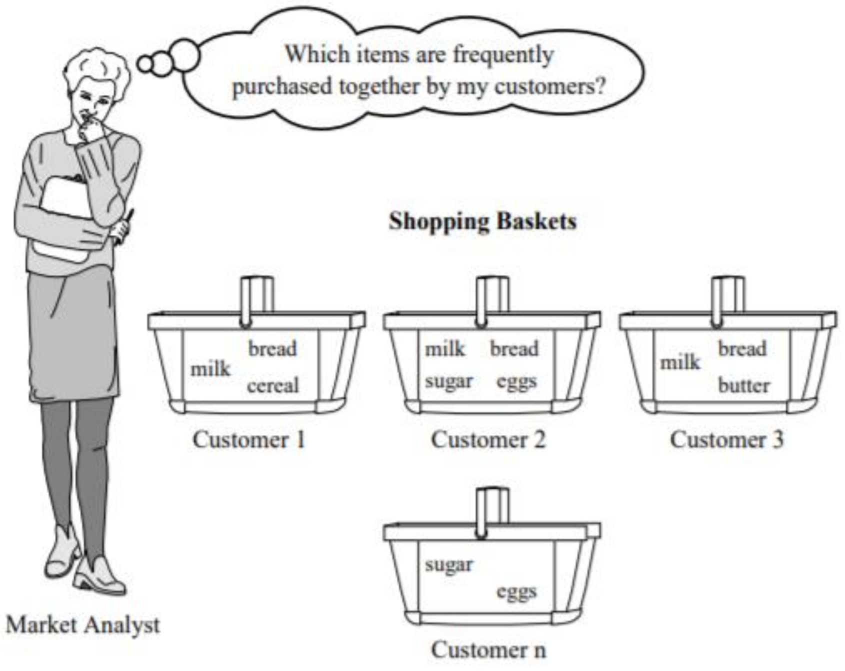
- Euclidean distance: $||a-b||_2 = \sqrt{(\Sigma(a_i-b_i))}$
- Squared Euclidean distance: $||a-b||_2^2 = \Sigma((a_i-b_i)^2)$
- Manhattan distance: $||a-b||_1 = \Sigma|a_i-b_i|$
- Pearson correlation distance
- Spearman correlation distance:
- …

Association Rule Mining

- is used when you want to **find an association between** different **objects in a set**, **find frequent patterns** in a transaction database, relational databases or any other information repository.
- The applications of Association Rule Mining are found in Marketing, Basket Data Analysis
  - **"Frequently Bought Together" → Association**
  - **"Customers who bought this item also bought" → Recommendation**

The market basket model

- Market Basket Analysis takes data at transaction level, which lists all items bought by a customer in a single purchase.
- The technique determines relationships of what products were purchased with which other product(s)



- These relationships are then used to build profiles containing If-Then rules of the items purchased.
- The rules could be written as: **If {X} Then {Y}**

Association Rule Mining

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- Item = products purchased in a basket/transaction
- An *itemset* is a set of item
- An itemset that contains k items is a k-itemset
  - Ex: {beer, diaper} : 2-itemset
- Let **D = {t_1,t_2...,t_m}** be the set of all transactions called the *dataset*.
  - Ex: t1 = {Milk, bread, eggs}
- Let **I = {i_1,i2,...,i_n}** be the set of all item in a market basket data
- Each transaction $\mathbf{t_i}$ contains a subset of items chosen from **I**
- An association *rule* is defined as an implication of the form: X ➔ Y, where X, Y $\subseteq$ I and X$\cap$Y = $\varnothing$
  - Ex: Diaper → beer (Buying diapers may likely lead to buying beers)
- **<u>Problem</u>: Find sets of items that appear together "frequently" in basket**

Association Rule Mining

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

Tid	Beer	Nuts	Diaper	Coffee	Egg	Milk
10	1	1	1	0	0	0
20	1	0	1	1	0	0
30	1	0	1	0	1	0
40	0	1	0	0	1	1
50	0	1	1	1	1	1

This presentation is a very simplistic view of market basket data

Association Rule Mining

10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- **Support count**: is the number of elements in a set.
- **Frequent itemsets**: a set of items that appears in many baskets is said to be "frequent."
  - Given a number **minsupp**, called support threshold. If support (X) >= minsup then X is frequent.
- **Association rule discovery**
  - Given a set of transaction T, find all the rules having support >= *minsup*, and confidence >= *minconf*, where minsup, minconf are the corresponding support and confidence thresholds

Association Rule Mining

10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- **Supp(X) = count(X) / | D |**
  - Ex: supp(beer} = 3/5 = 60%;
  - Ex: supp(diaper} = 4/5 = 80%
  - Ex: sup(beer, diaper} = 3/5 = 60%
- **The strength of an association rule can be measure as**:
  - **Support s(X→Y)**: The percentage of transaction in the database that contains $X \cup Y$
    - **supp ($X \Rightarrow Y$) = supp($X \cup Y$) = count ($X \cup Y$)/|D|**
  - **Confidence, c(X→Y)**: The conditional probability that a transaction containing X also contains Y
    - **Conf ($X \Rightarrow Y$) = supp($X \cup Y$) / supp(X)**
    - Ex. c = supp{Diaper, Beer}/supp{Diaper} = ¾ = 0.75

Association Rule Mining

- Let **minsupp = 50%**
- All the frequent 1-itemsets:
  - Beer : 3/5 (60%)
  - Nut : 3/5 (60%)
  - Diaper: 4/5 (80%)
  - Egg: 3/5 (60%)
  - Coffee: 2/5 (40%) < minsupp
  - Milk: 2/5 (40%) < minsupp

Beer : 3/5 (60%)
Nut : 3/5 (60%)
Diaper: 4/5 (80%)
Egg: 3/5 (60%)

- All the frequent 2-itemsets: {Beer, Diaper}: 3/5 (60%)
- All the frequent 3-itemsets: None.

Apriori Algorithm

Association rule mining can be viewed as a two-step process:

1. Finds subsets that are common to at least a minimum number of the itemsets
2. Uses a bottom-up approach
   - frequent subsets are extended one item at a time (the size of frequent subsets increases from one-item subsets to two-item subsets, then three-item subsets, and so on), and
   - groups of candidates at each level are tested against the data for minimum support.

The Apriori Algorithm—An Example

minsup = 2

Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

1st scan

C_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

F_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

C_2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

2nd scan

C_2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

F_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

C_3

Itemset
{B, C, E}

3rd scan

F_3

Itemset	sup
{B, C, E}	2

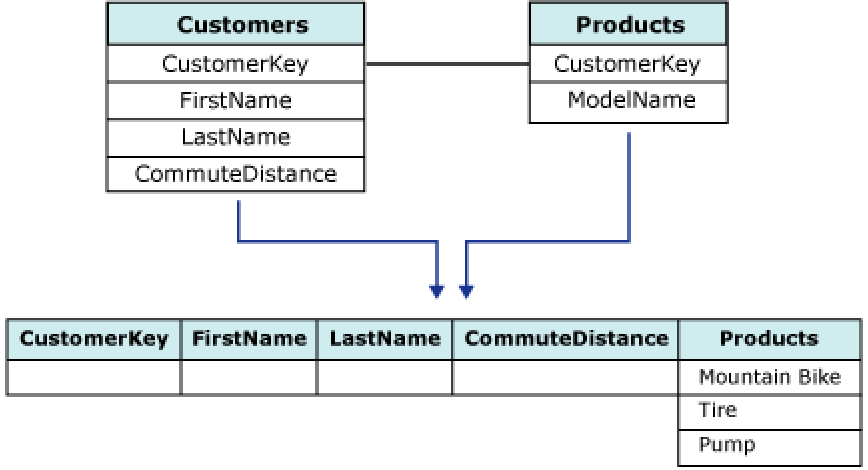
http://hanj.cs.illinois.edu/cs412/bk3_slides/06FPBasic.pdf

Association rule with SSAS

- Data source: **AdventureWorksDW**
- **Problem:** The objective of the Association Rule Mining is to find out what models are selling together.

https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-association-algorithm?view=asallproducts-allversions

Association rule with SSAS



1. Data preparation:

- The requirements for an association rules model are as follows:
- **A single key column**: one numeric or text column that uniquely identifies each record. compound keys not permitted.
- **A single predictable column**: The values must be discrete or discretized.
- **Input columns:** The input columns must be discrete. The input data for an association model often is contained in two tables:
  - one table might contain customer information while another table contains customer purchases
  - **vAssocSeqLineItems: nested table**
  - **vAssocSeqOrders: case table**

To add a data source view

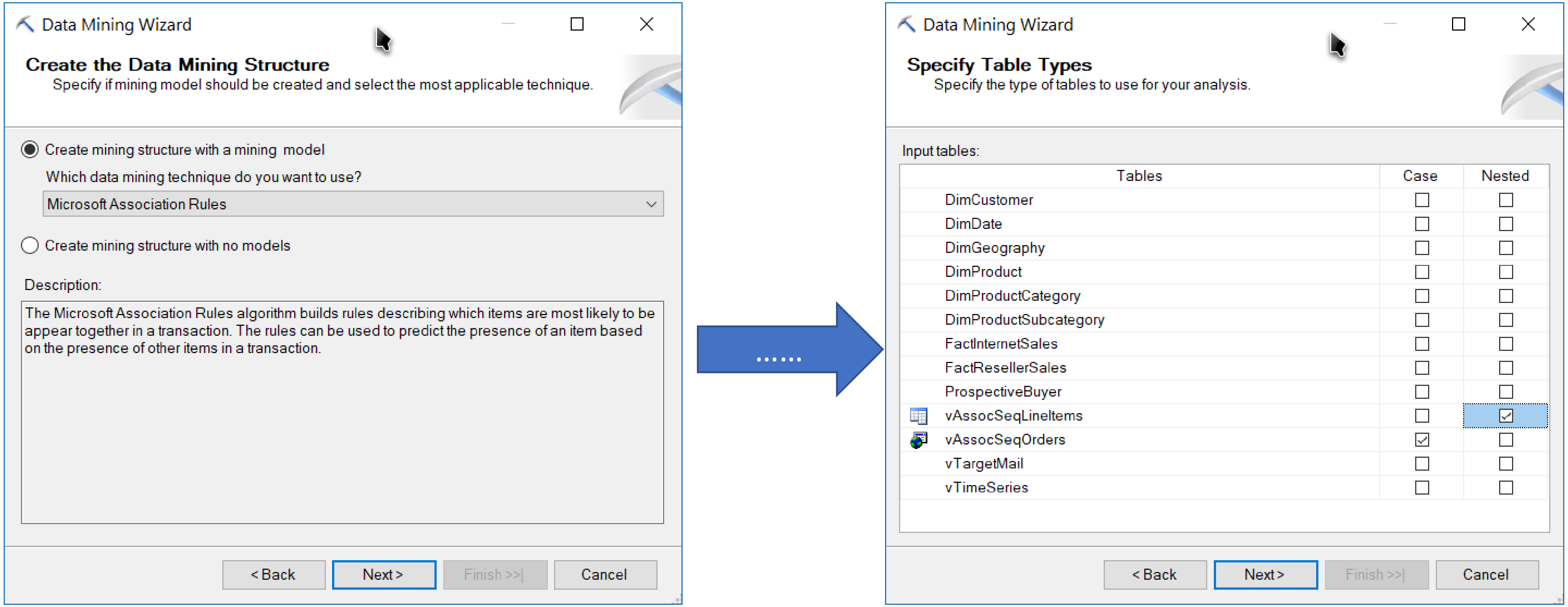
1. In Solution Explorer, right-click **Data Source Views**, and then select **New Data Source View**.

   The Data Source View Wizard opens.

2. On the **Welcome to the Data Source View Wizard** page, click **Next**.
3. On the **Select a Data Source** page, under **Relational data sources**, select the Adventure Works DW Multidimensional 2012 data source that you created in the Basic Data Mining Tutorial. Click **Next**.
4. On the **Select Tables and Views** page, select the following tables, and then click the right arrow to include them in the new data source view:
   - **vAssocSeqOrders**
   - **vAssocSeqLineItems**
5. Click **Next**.
6. On the **Completing the Wizard** page, by default the data source view is named Adventure Works DW Multidimensional 2012 . Change the name to **Orders**, and then click **Finish**.

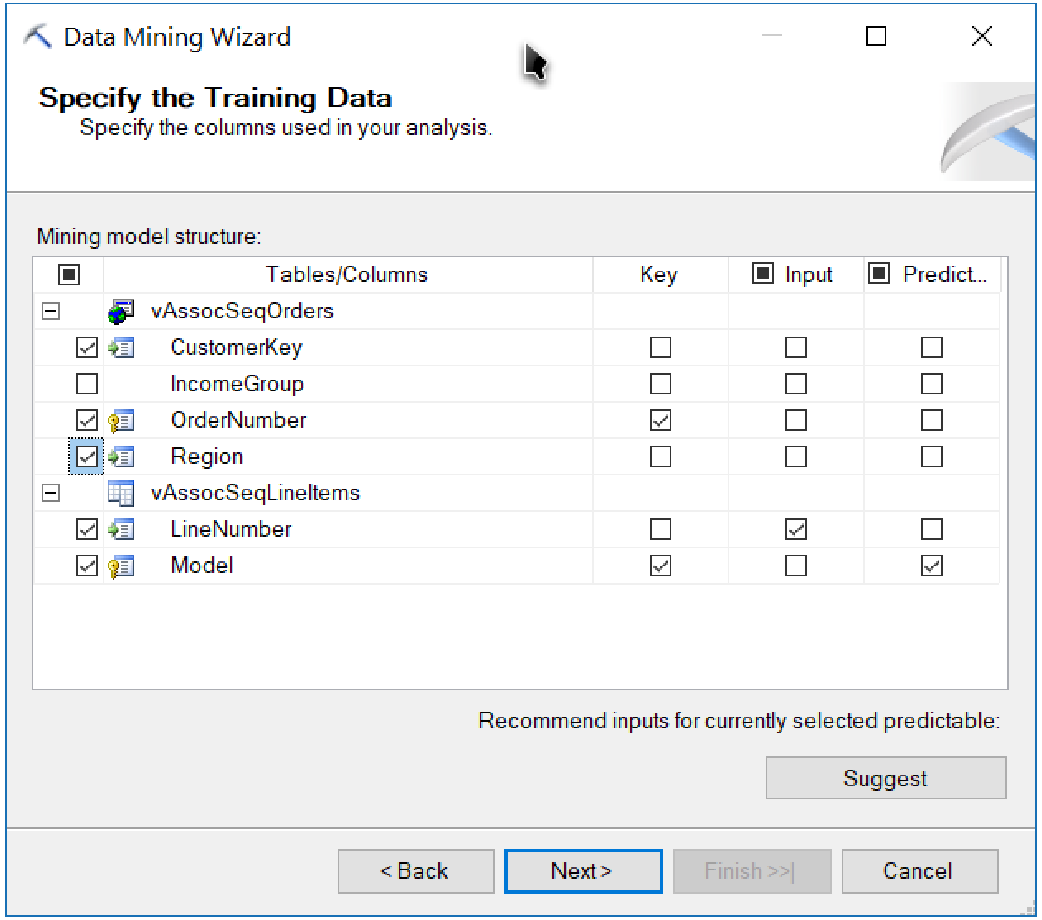
   Data Source View Designer opens and the **Orders** data source view appears.

Association rule with SSAS



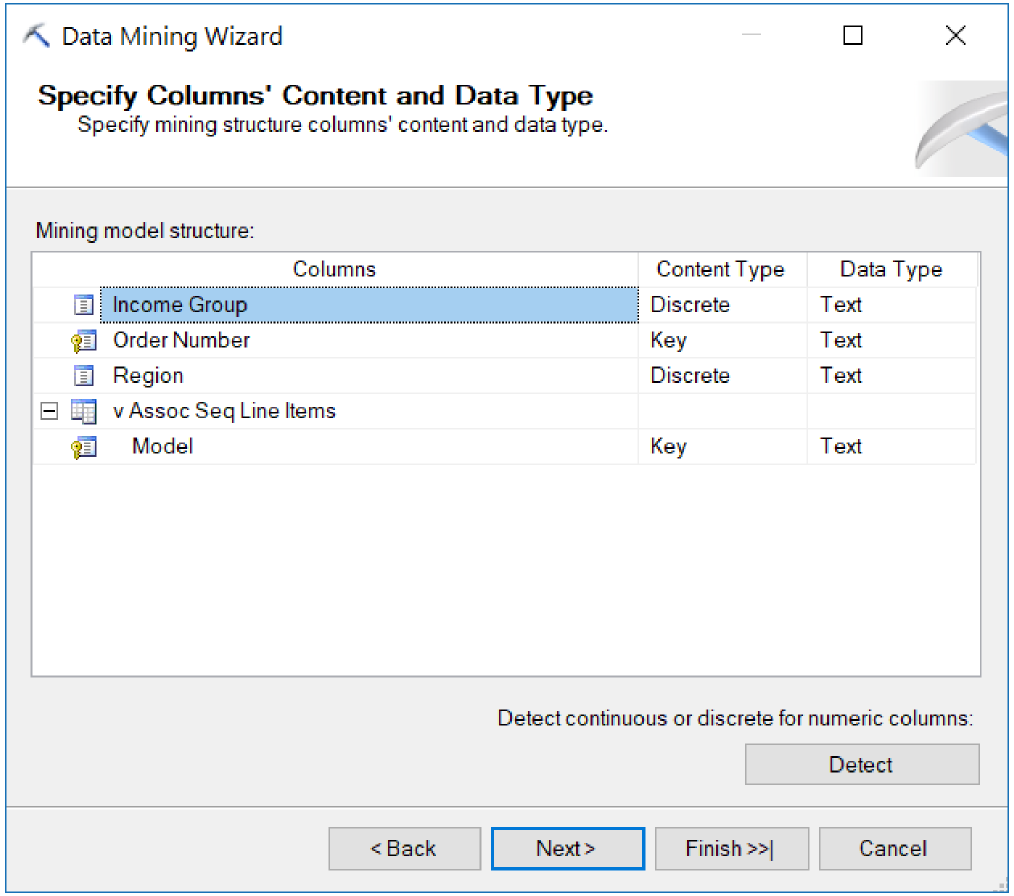
Association rule with SSAS

- After the Association Rule configuration is completed, then the model can be processed. Then users can review the prediction model and perform the predictions.

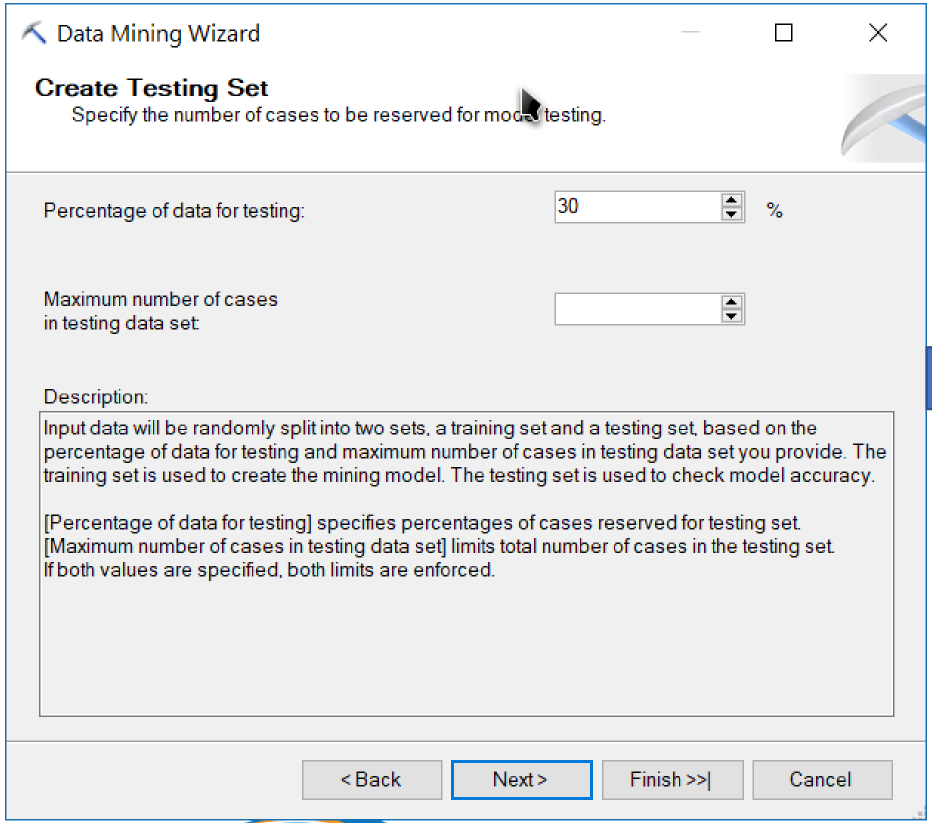


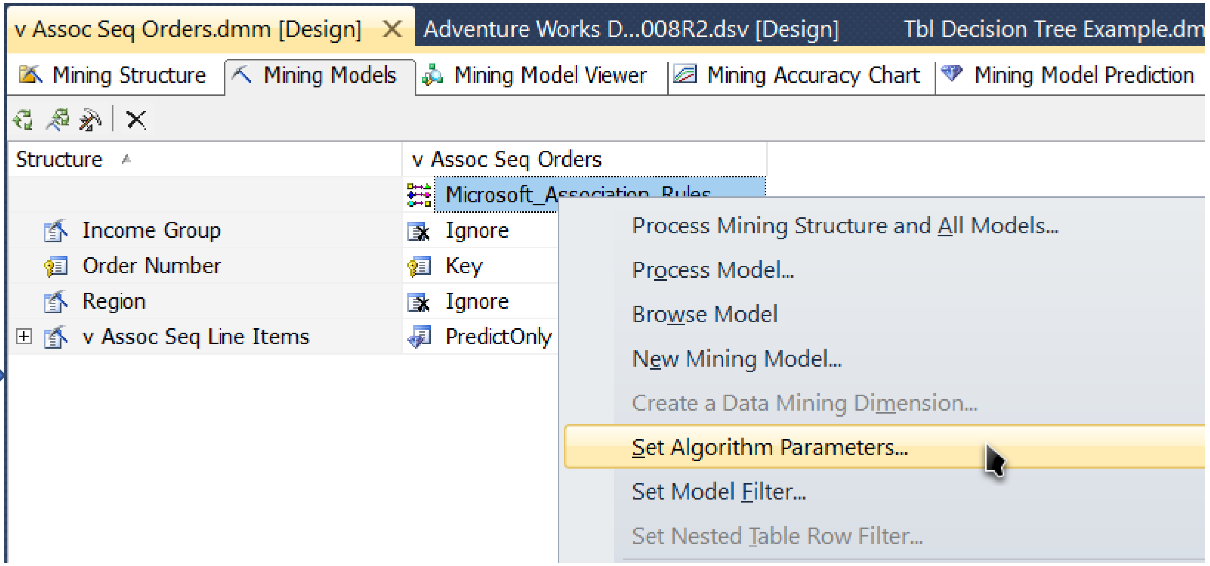
Association rule with SSAS

- **Input columns:** The input columns must be discrete
- **A single predictable column**: The values must be discrete or discretized

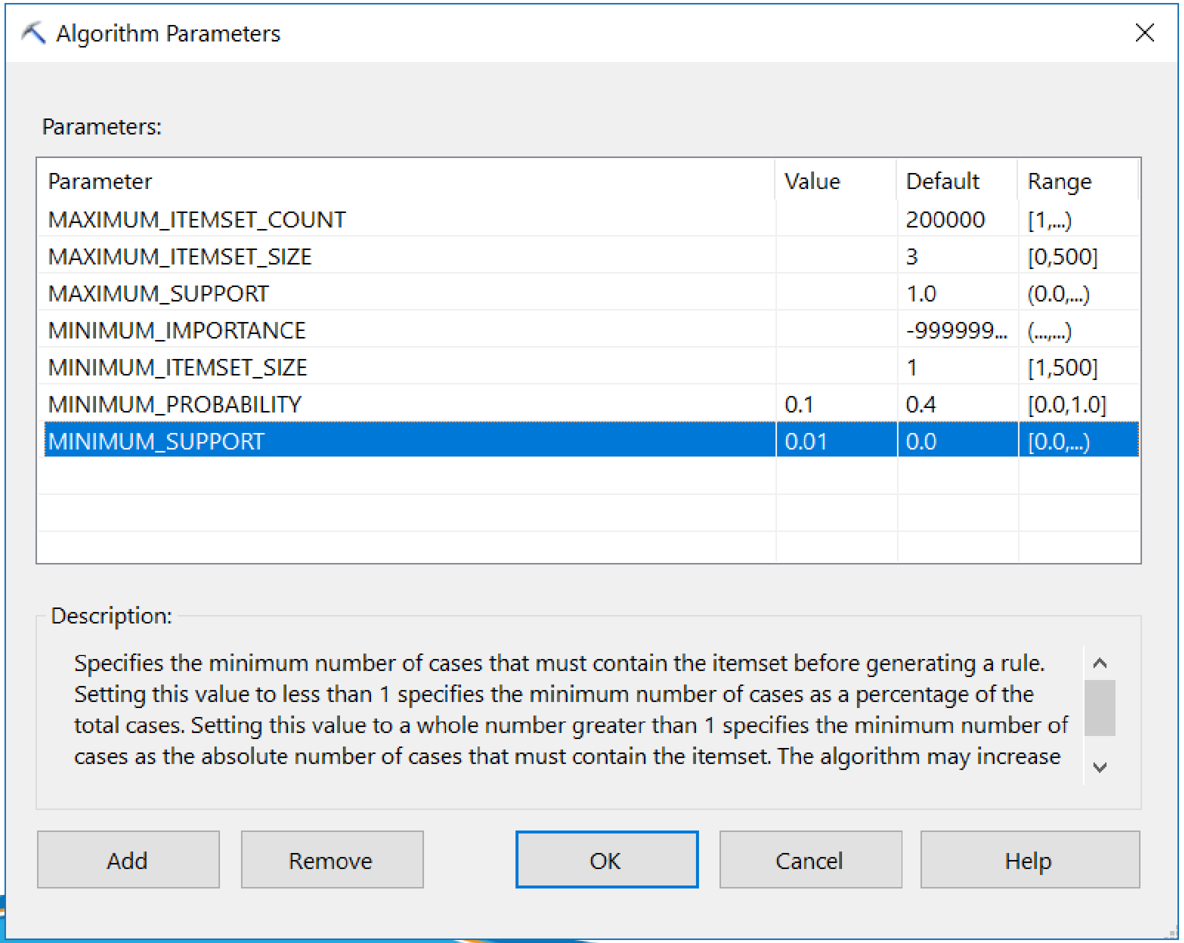


Modifying and Processing the Market Basket Model



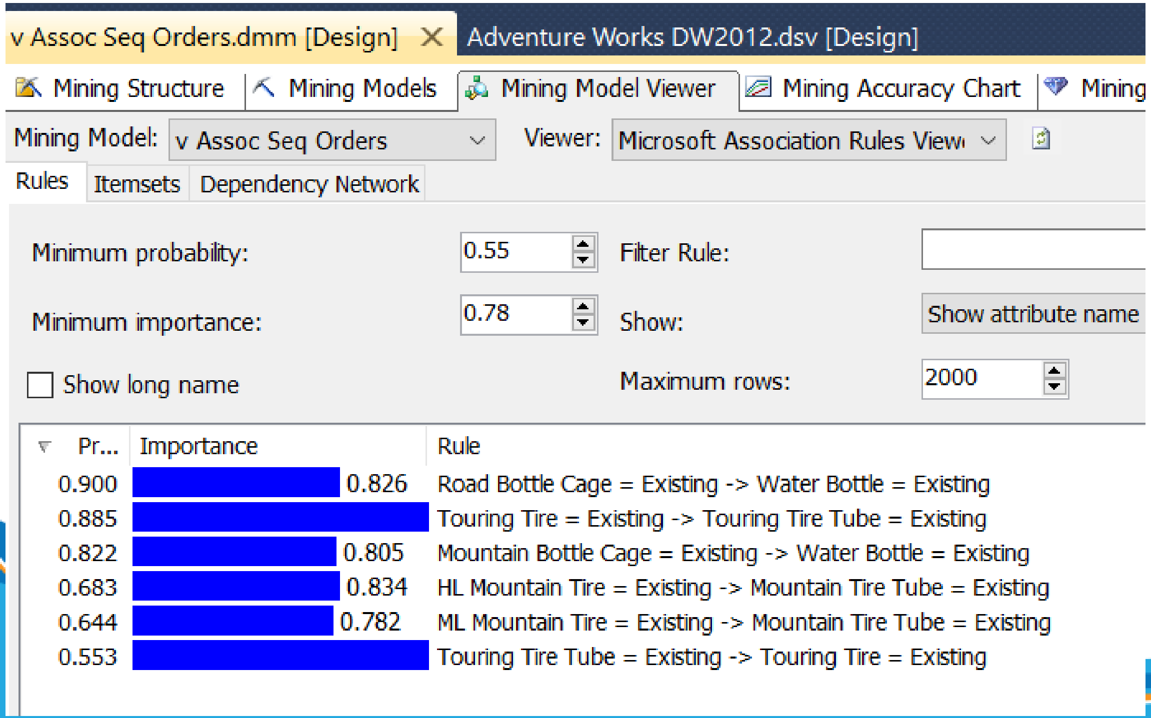


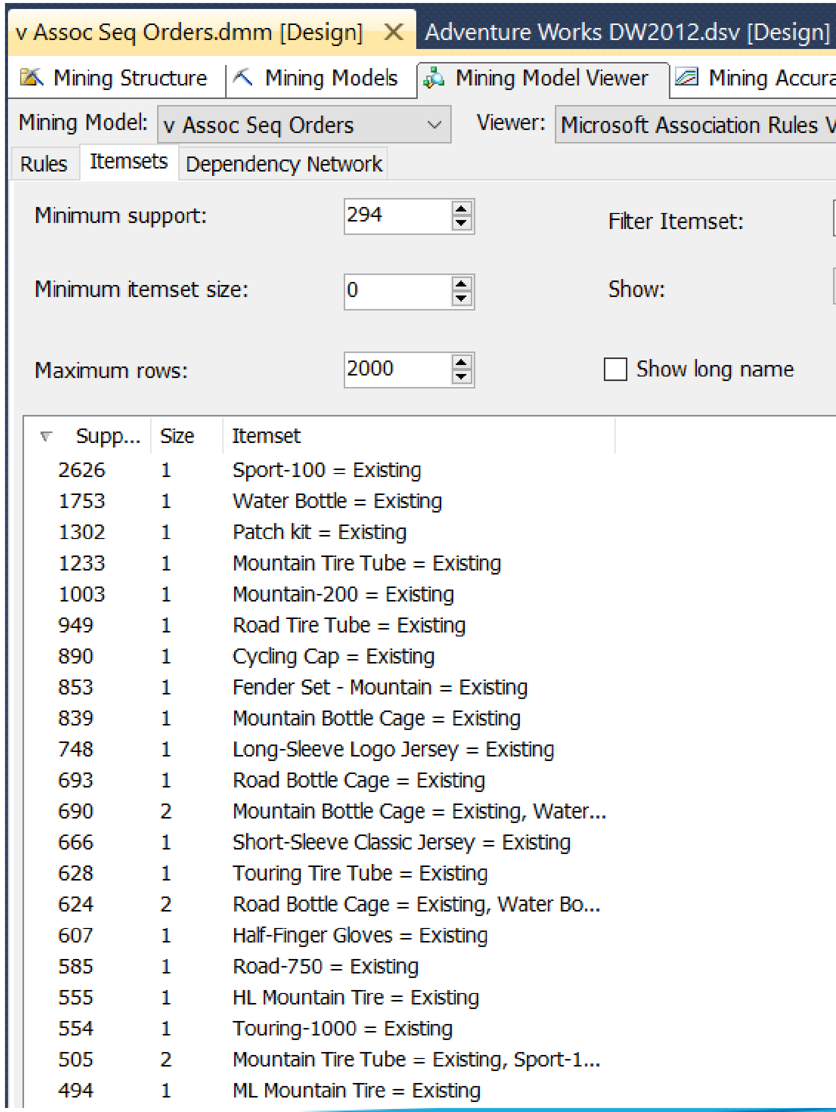
- Before you process the association mining model that you created, you must change the default values of two of the parameters: **Support and Probability.**
- **MINIMUM_PROBABILITY = 0.5**
- **MINIMUM_SUPPORT = 0.03**

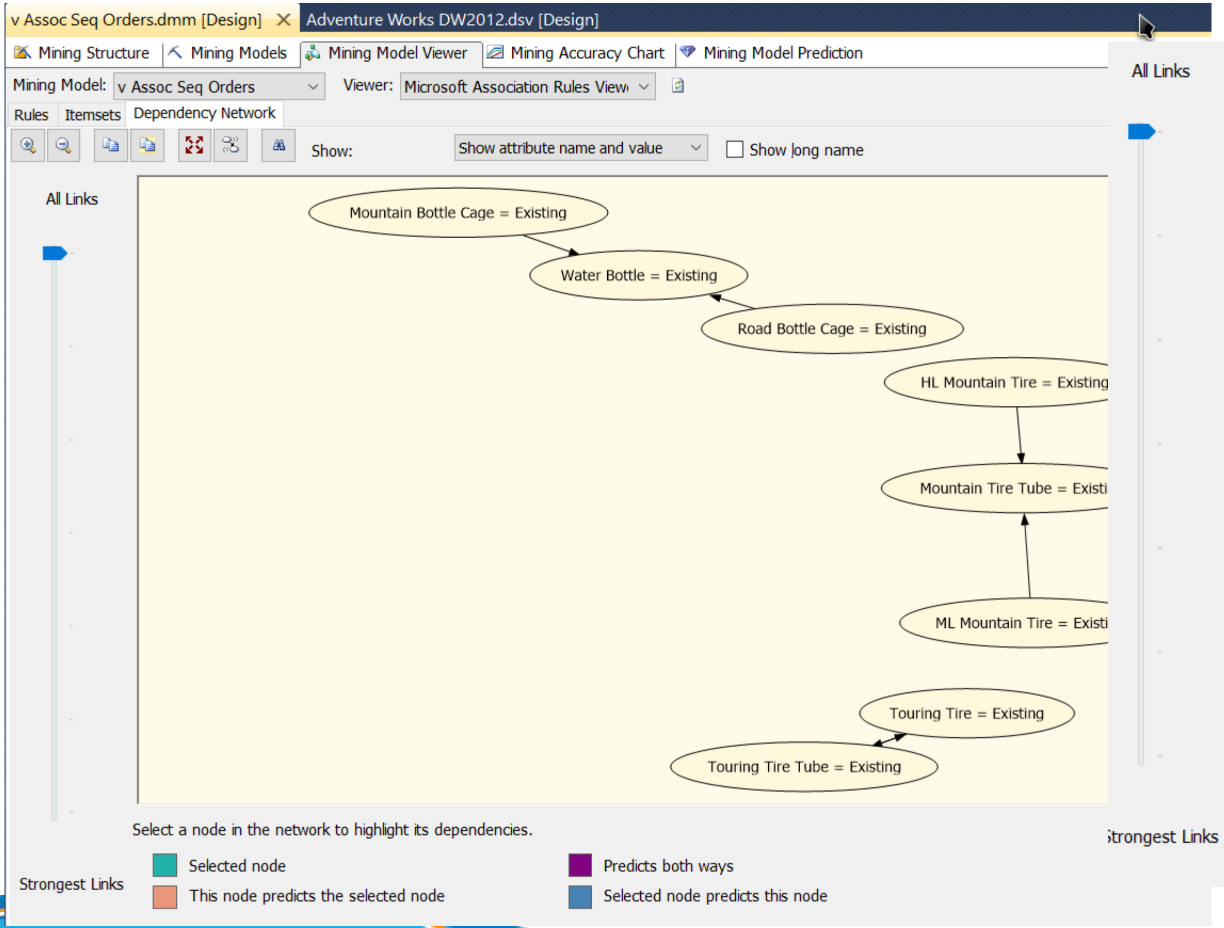


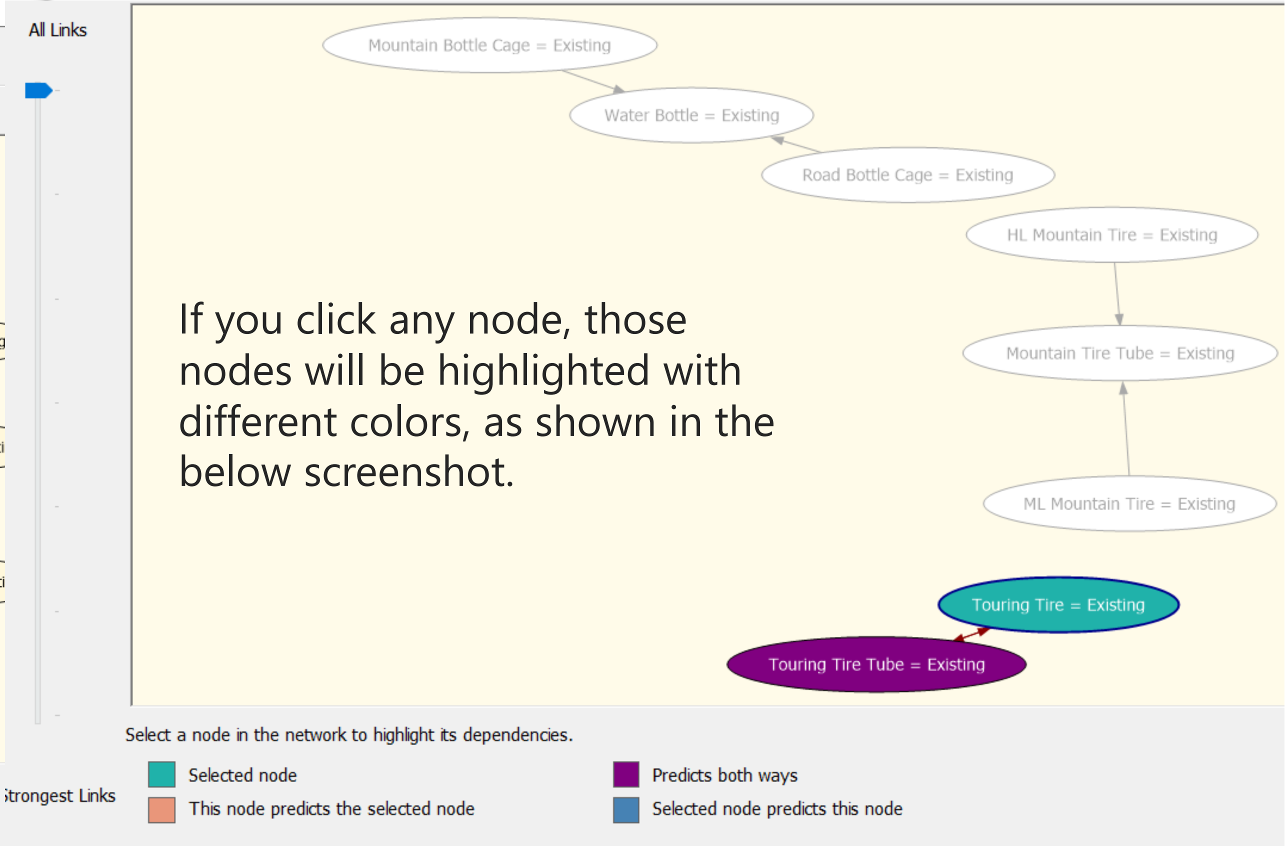
- support :
  - MINIMUM_SUPPORT value of 0.03, it means that at least 3% of the total cases in the data set must contain this item or itemset for inclusion in the model.
- *Confidence:*
  - INIMUM_PROBABILITY: 0.5
  - means that no rule with less than fifty percent probability can be generated

- In the Mining Model viewer, there are three tabs to view the data patterns. In the **Rules** tab, it will show the rules that can be derived fro the Association Rule Mining model in the sample set.

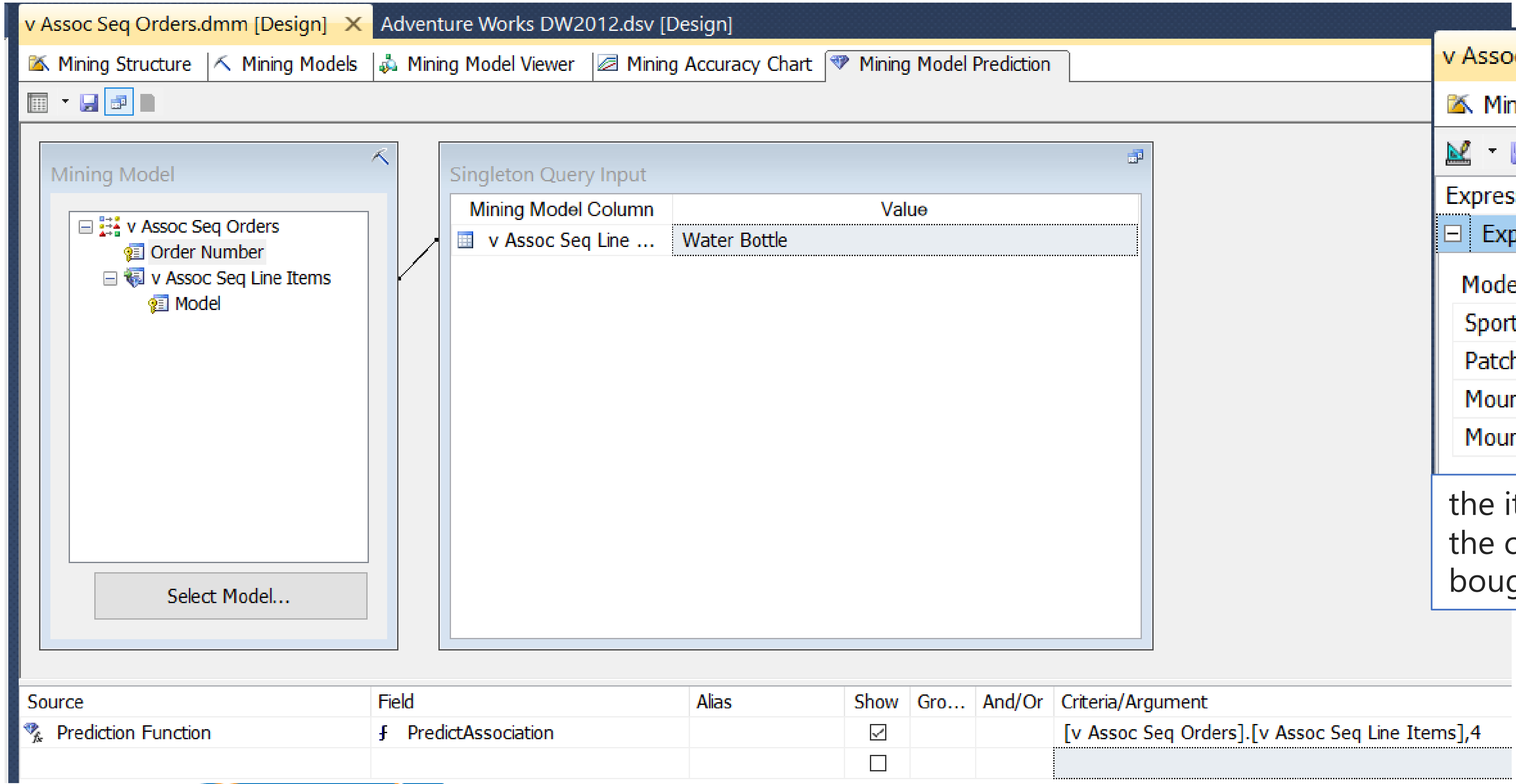


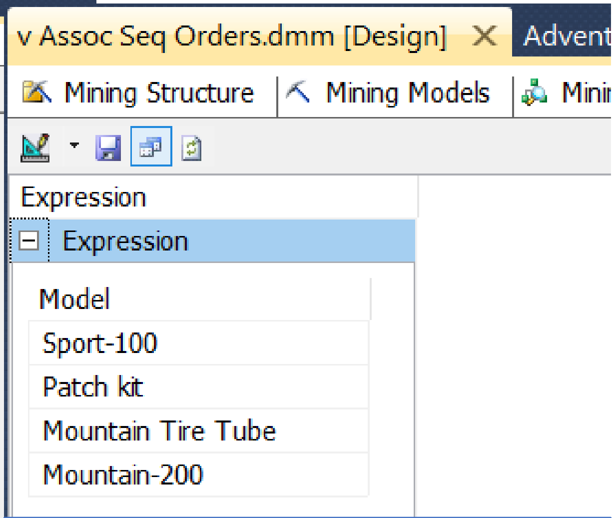






If you click any node, those nodes will be highlighted with different colors, as shown in the below screenshot.





the items which will be bought by the customers who had bought *Water Bottle*.

References

- J Han, J Pei, M Kamber - Data mining: concepts and techniques
- Ian H. Witten, Eibe Frank, Mark A. Hall - Data Mining Practical Machine Learning Tools and Techniques
- https://medium.com/analytics-vidhya/entropy-calculation-information-gain-decision-tree-learning-771325d16f
- https://www.saedsayad.com/decision_tree.htm
- http://hanj.cs.illinois.edu/cs412/bk3/06.pdf

Bài tập

- Mindmap nội dung đã học
- Mô tả kiểu dữ liệu cho dữ liệu input, output của thuật toán:
  - Cây quyết định
  - Gom cụm
  - Luật kết hợp
- Dự đoán doanh thu, lợi nhuận theo thời gian.
  - https://www.sqlshack.com/microsoft-time-series-in-sql-server/