

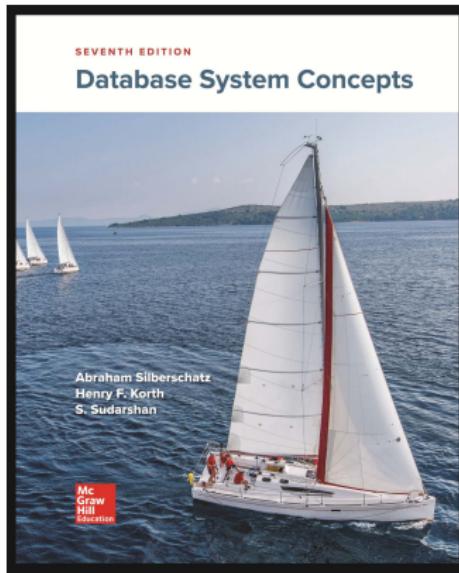
# Database Administration

## Lecture 10: Data Analytics.

Silberschatz, Korth & Sudarshan

21 de abril de 2025

# Database Administration: Data Analytics.



Content has been extracted from *Database System Concepts 7ed (Chapter 11)*, by Abraham Silberschatz, Henry Korth & S. Sudarshan, 2019. Visit <https://db-book.com/>.

# Plan

## Overview

Data Warehousing

OLAP

Data Mining



# Overview

- **Data analytics:** the processing of data to infer patterns, correlations, or models for prediction
- Primarily used to make business decisions
  - Per individual customer
    - E.g. what product to suggest for purchase
  - Across all customers
    - E.g. what products to manufacture/stock, in what quantity
- Critical for businesses today



# Overview (Cont.)

- Common steps in data analytics
  - Gather data from multiple sources into one location
    - Data warehouses also integrated data into common schema
    - Data often needs to be **extracted** from source formats, **transformed** to common schema, and **loaded** into the data warehouse
      - Can be done as **ETL (extract-transform-load)**, or **ELT (extract-load-transform)**
  - Generate aggregates and reports summarizing data
    - Dashboards showing graphical charts/reports
    - **Online analytical processing (OLAP) systems** allow interactive querying
    - Statistical analysis using tools such as R/SAS/SPSS
      - Including extensions for parallel processing of big data
  - Build **predictive models** and use the models for decision making



# Overview (Cont.)

- Predictive models are widely used today
  - E.g. use customer profile features (e.g. income, age, gender, education, employment) and past history of a customer to predict likelihood of default on loan
    - and use prediction to make loan decision
  - E.g. use past history of sales (by season) to predict future sales
    - And use it to decide what/how much to produce/stock
    - And to target customers
- Other examples of business decisions:
  - What items to stock?
  - What insurance premium to change?
  - To whom to send advertisements?



## Overview (Cont.)

- **Machine learning** techniques are key to finding patterns in data and making predictions
- **Data mining** extends techniques developed by machine-learning communities to run them on very large datasets
- The term **business intelligence (BI)** is synonym for data analytics
- The term **decision support** focuses on reporting and aggregation

# Plan

Overview

Data Warehousing

OLAP

Data Mining

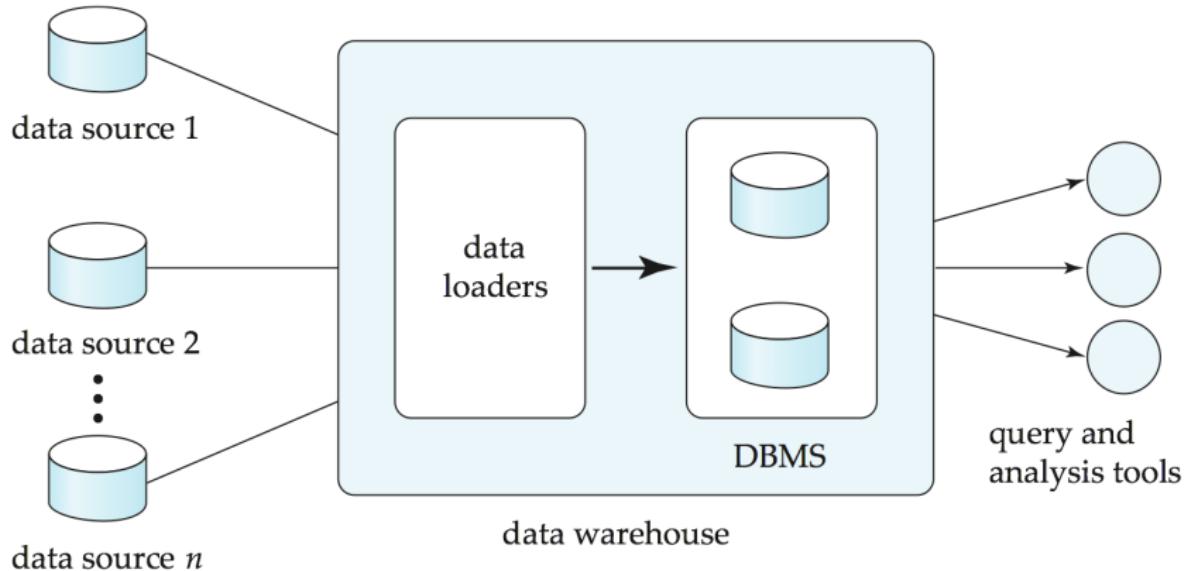


# Data Warehousing

- Data sources often store only current data, not historical data
- Corporate decision making requires a unified view of all organizational data, including historical data
- A **data warehouse** is a repository (archive) of information gathered from multiple sources, stored under a unified schema, at a single site
  - Greatly simplifies querying, permits study of historical trends
  - Shifts decision support query load away from transaction processing systems



# Data Warehousing





# Design Issues

- *When and how to gather data*
  - **Source driven architecture:** data sources transmit new information to warehouse
    - either continuously or periodically (e.g. at night)
  - **Destination driven architecture:** warehouse periodically requests new information from data sources
  - **Synchronous vs asynchronous replication**
    - Keeping warehouse exactly synchronized with data sources (e.g. using two-phase commit) is often too expensive
    - Usually OK to have slightly out-of-date data at warehouse
    - Data/updates are periodically downloaded from online transaction processing (OLTP) systems.
- *What schema to use*
  - Schema integration



# More Warehouse Design Issues

- **Data transformation** and **data cleansing**
  - E.g. correct mistakes in addresses (misspellings, zip code errors)
  - Merge address lists from different sources and **purge** duplicates
- *How to propagate updates*
  - Warehouse schema may be a (materialized) view of schema from data sources
    - View maintenance
- *What data to summarize*
  - Raw data may be too large to store on-line
  - Aggregate values (totals/subtotals) often suffice
  - Queries on raw data can often be transformed by query optimizer to use aggregate values

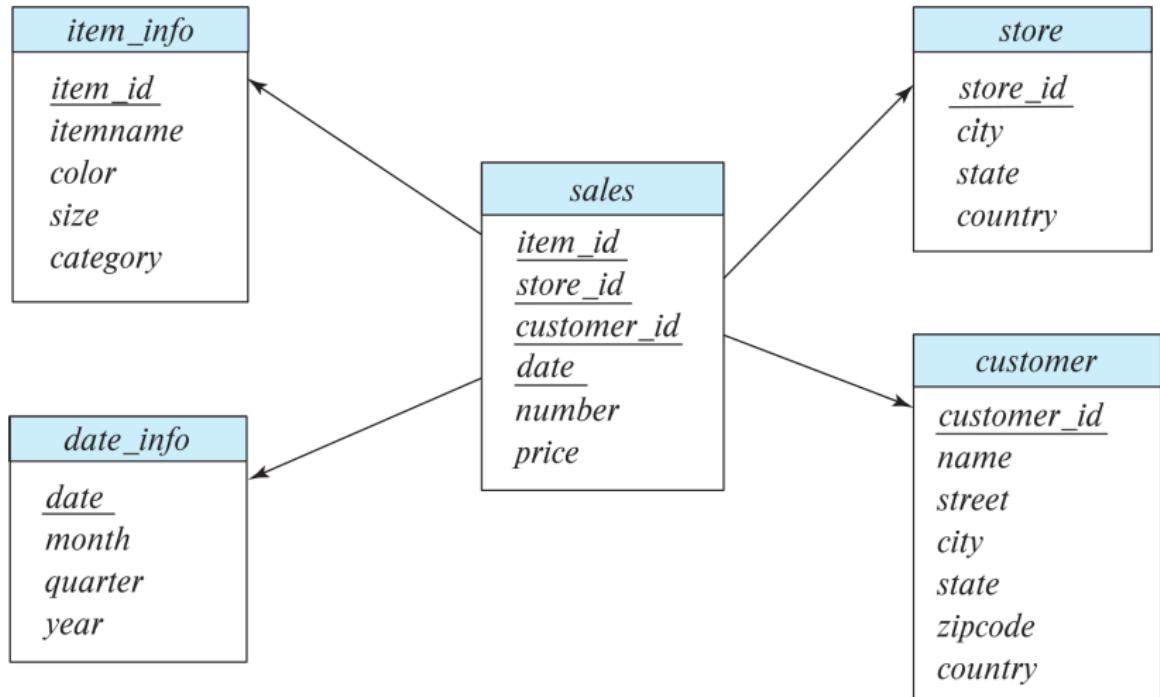


# Multidimensional Data and Warehouse Schemas

- Data in warehouses can usually be divided into
  - **Fact tables**, which are large
    - E.g *sales(item\_id, store\_id, customer\_id, date, number, price)*
  - **Dimension tables**, which are relatively small
    - Store extra information about stores, items, etc
- Attributes of fact tables can be usually viewed as
  - **Measure attributes**
    - measure some value, and can be aggregated upon
    - e.g., the attributes *number* or *price* of the *sales* relation
  - **Dimension attributes**
    - dimensions on which measure attributes are viewed
    - e.g., attributes *item\_id*, *color*, and *size* of the *sales* relation
    - Usually small ids that are foreign keys to dimension tables



# Data Warehouse Schema





# Multidimensional Data and Warehouse Schemas

- Resultant schema is called a **star schema**
  - More complicated schema structures
    - **Snowflake schema**: multiple levels of dimension tables
    - May have multiple fact tables
- Typically
  - fact table joined with dimension tables and then
  - group-by on dimension table attributes, and then
  - aggregation on measure attributes of fact table
- Some applications do not find it worthwhile to bring data to a common schema
  - **Data lakes** are repositories which allow data to be stored in multiple formats, without schema integration
  - Less upfront effort, but more effort during querying



# Database Support for Data Warehouses

- Data in warehouses usually append only, not updated
  - Can avoid concurrency control overheads
- Data warehouses often use **column-oriented storage**
  - E.g. a sequence of *sales* tuples is stored as follows
    - Values of item\_id attribute are stored as an array
    - Values of store\_id attribute are stored as an array,
    - And so on
  - Arrays are compressed, reducing storage, IO and memory costs significantly
  - Queries can fetch only attributes that they care about, reducing IO and memory cost
  - More details in Section 13.6
- Data warehouses often use parallel storage and query processing infrastructure
  - Distributed file systems, Map-Reduce, Hive, ...

# Plan

Overview

Data Warehousing

OLAP

Data Mining



# Data Analysis and OLAP

- **Online Analytical Processing (OLAP)**
  - Interactive analysis of data, allowing data to be summarized and viewed in different ways in an online fashion (with negligible delay)
- We use the following relation to illustrate OLAP concepts
  - *sales (item\_name, color, clothes\_size, quantity)*

This is a simplified version of the *sales* fact table joined with the dimension tables, and many attributes removed (and some renamed)



# Example sales relation

item_name	color	clothes_size	quantity
dress	dark	small	2
dress	dark	medium	6
dress	dark	large	12
dress	pastel	small	4
dress	pastel	medium	3
dress	pastel	large	3
dress	white	small	2
dress	white	medium	3
dress	white	large	0
pants	dark	small	14
pants	dark	medium	6
pants	dark	large	0
pants	pastel	small	1
pants	pastel	medium	0
pants	pastel	large	1
pants	white	small	3
pants	white	medium	0
pants	white	large	2
shirt	dark	small	2
shirt	dark	medium	6
shirt	dark	large	6
shirt	pastel	small	4
shirt	pastel	medium	1
shirt	pastel	large	2
shirt	white	small	17
shirt	white	medium	1
shirt	white	large	10
skirt	dark	small	2
skirt	dark	medium	5
...	...	...	...
...	...	...	...



## Cross Tabulation of sales by *item\_name* and *color*

*clothes\_size* all

		color		
		dark	pastel	white
<i>item_name</i>	skirt	8	35	10
	dress	20	10	5
	shirt	14	7	28
	pants	20	2	5
	total	62	54	48
		total		164

- The table above is an example of a **cross-tabulation (cross-tab)**, also referred to as a **pivot-table**.
  - Values for one of the dimension attributes form the row headers
  - Values for another dimension attribute form the column headers
  - Other dimension attributes are listed on top
  - Values in individual cells are (aggregates of) the values of the dimension attributes that specify the cell.



# Data Cube

- A **data cube** is a multidimensional generalization of a cross-tab
- Can have n dimensions; we show 3 below
- Cross-tabs can be used as views on a data cube

		item_name					clothes_size			
		skirt	dress	shirt	pants	all	all	large	medium	small
color		dark	8	20	14	20	62	4	16	34
pastel		35	10	7	2	54	21	9	18	45
white		10	5	28	5	48	77	42	45	42
all		53	35	49	27	164	all	large	medium	small



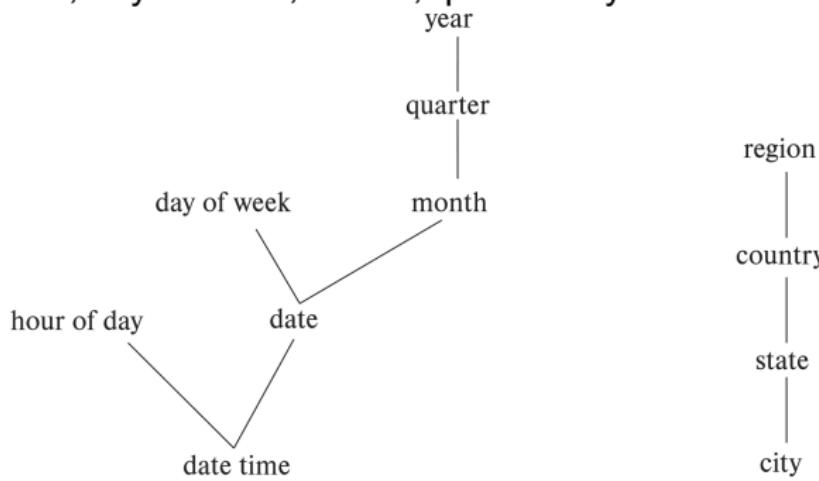
# Online Analytical Processing Operations

- **Pivoting:** changing the dimensions used in a cross-tab
  - E.g. moving colors to column names
- **Slicing:** creating a cross-tab for fixed values only
  - E.g fixing color to white and size to small
  - Sometimes called **dicing**, particularly when values for multiple dimensions are fixed.
- **Rollup:** moving from finer-granularity data to a coarser granularity
  - E.g. aggregating away an attribute
  - E.g. moving from aggregates by day to aggregates by month or year
- **Drill down:** The opposite operation - that of moving from coarser-granularity data to finer-granularity data



# Hierarchies on Dimensions

- **Hierarchy** on dimension attributes: lets dimensions be viewed at different levels of detail
- E.g., the dimension *datetime* can be used to aggregate by hour of day, date, day of week, month, quarter or year



(a) time hierarchy

(b) location hierarchy



# Cross Tabulation With Hierarchy

- Cross-tabs can be easily extended to deal with hierarchies
- Can drill down or roll up on a hierarchy
- E.g. hierarchy: *item\_name* → *category*

*clothes\_size:* all

	<i>category</i>	<i>item_name</i>	<i>color</i>			
			dark	pastel	white	total
womenswear	skirt	8	8	10	53	88
	dress	20	20	5	35	
	subtotal	28	28	15		
menswear	pants	14	14	28	49	76
	shirt	20	20	5	27	
	subtotal	34	34	33		
total		62	62	48		164



# Relational Representation of Cross-tabs

- Cross-tabs can be represented as relations
- We use the value **all** to represent aggregates.
- The SQL standard actually uses *null* values in place of **all**
  - Works with any data type
  - But can cause confusion with regular null values.

item_name	color	clothes_size	quantity
skirt	dark	all	8
skirt	pastel	all	35
skirt	white	all	10
skirt	all	all	53
dress	dark	all	20
dress	pastel	all	10
dress	white	all	5
dress	all	all	35
shirt	dark	all	14
shirt	pastel	all	7
shirt	white	all	28
shirt	all	all	49
pants	dark	all	20
pants	pastel	all	2
pants	white	all	5
pants	all	all	27
all	dark	all	62
all	pastel	all	54
all	white	all	48
all	all	all	164

# Plan

Overview

Data Warehousing

OLAP

Data Mining



# Data Mining

- **Data mining** is the process of semi-automatically analyzing large databases to find useful patterns
  - Similar goals to machine learning, but on very large volumes of data
- Part of the larger area of **knowledge discovery in databases (KDD)**
- Some types of knowledge can be represented as rules
- More generally, knowledge is discovered by applying machine learning techniques on past instances of data, to form a **model**
  - Model is then used to make predictions for new instances



# Types of Data Mining Tasks

- **Prediction** based on past history
  - Predict if a credit card applicant poses a good credit risk, based on some attributes (income, job type, age, ..) and past history
  - Predict if a pattern of phone calling card usage is likely to be fraudulent
- Some examples of prediction mechanisms:
  - **Classification**
    - Items (with associated attributes) belong to one of several classes
    - **Training instances** have attribute values and classes provided
    - Given a new item whose class is unknown, predict to which class it belongs based on its attribute values
  - **Regression** formulae
    - Given a set of mappings for an unknown function, predict the function result for a new parameter value



# Data Mining (Cont.)

- **Descriptive Patterns**

- **Associations**

- Find books that are often bought by “similar” customers. If a new such customer buys one such book, suggest the others too.

- Associations may be used as a first step in detecting **causation**

- E.g. association between exposure to chemical X and cancer,

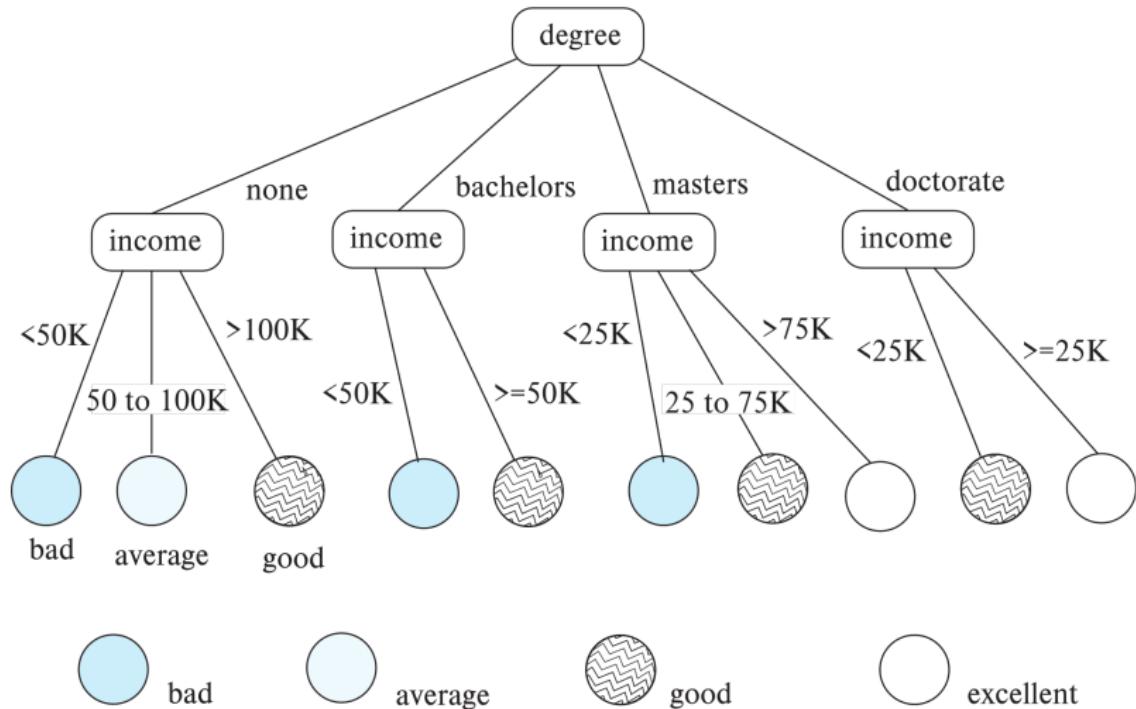
- **Clusters**

- E.g. typhoid cases were clustered in an area surrounding a contaminated well

- Detection of clusters remains important in detecting epidemics



# Decision Tree Classifiers





# Decision Trees

- Each internal node of the tree partitions the data into groups based on a **partitioning attribute**, and a **partitioning condition** for the node
- Leaf node:
  - all (or most) of the items at the node belong to the same class, or
  - all attributes have been considered, and no further partitioning is possible.
- Traverse tree from top to make a prediction
- Number of techniques for constructing decision tree classifiers
  - We omit details



# Bayesian Classifiers

- Bayesian classifiers use **Bayes theorem**, which says

$$p(c_j | d) = \frac{p(d | c_j) p(c_j)}{p(d)}$$

where

$p(c_j | d)$  = probability of instance  $d$  being in class  $c_j$ ,

$p(d | c_j)$  = probability of generating instance  $d$  given class  $c_j$ ,

$p(c_j)$  = probability of occurrence of class  $c_j$ , and

$p(d)$  = probability of instance  $d$  occurring



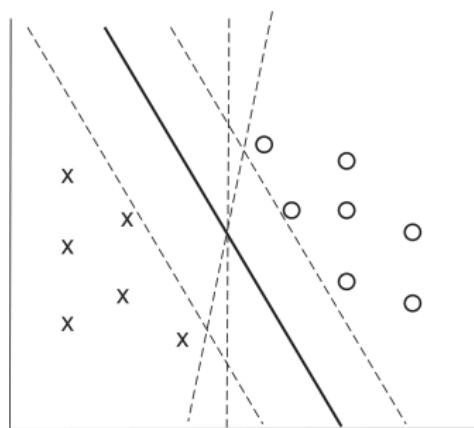
# Naïve Bayesian Classifiers

- Bayesian classifiers require
  - computation of  $p(d | c_j)$
  - precomputation of  $p(c_j)$
  - $p(d)$  can be ignored since it is the same for all classes
- To simplify the task, **naïve Bayesian classifiers** assume attributes have independent distributions, and thereby estimate
$$p(d | c_j) = p(d_1 | c_j) * p(d_2 | c_j) * \dots * (p(d_n | c_j))$$
  - Each of the  $p(d_i | c_j)$  can be estimated from a histogram on  $d_i$  values for each class  $c_j$ 
    - the histogram is computed from the training instances
  - Histograms on multiple attributes are more expensive to compute and store



# Support Vector Machine Classifiers

- Simple 2-dimensional example:
  - Points are in two classes
  - Find a line (**maximum margin line**) s.t. line divides two classes, and distance from nearest point in either class is maximum





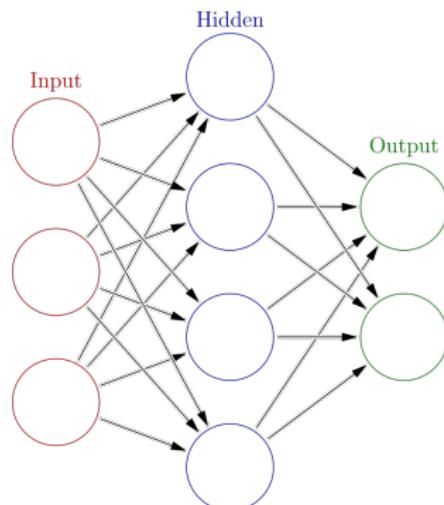
# Support Vector Machine

- In n-dimensions points are divided by a plane, instead of a line
- SVMs can be used separators that are curve, not necessarily linear, by transforming points before classification
  - Transformation functions may be non-linear and are called kernel functions
  - Separator is a plane in the transformed space, but maps to curve in original space
- There may not be an exact planar separator for a given set of points
  - Choose plane that best separates points
- N-ary classification can be done by N binary classifications
  - In class  $i$  vs. not in class  $i$ .



# Neural Network Classifiers

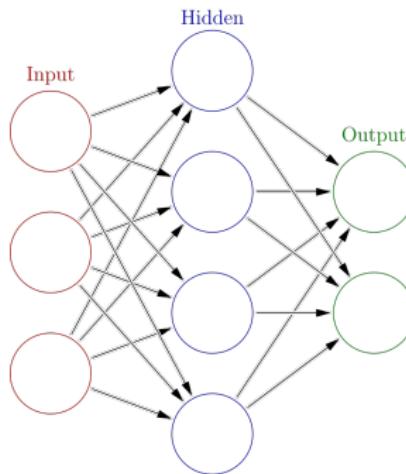
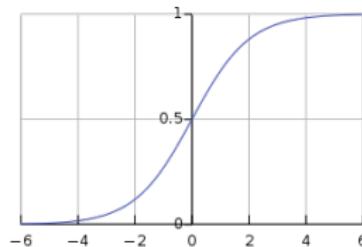
- Neural network has multiple layers
  - Each layer acts as input to next later
- First layer has input nodes, which are assigned values from input attributes
- Each node combines values of its inputs using some weight function to compute its value
  - Weights are associated with edges
- For classification, each output value indicates likelihood of the input instance belonging to that class
  - Pick class with maximum likelihood
- Weights of edges are key to classification
- Edge weights are learnt during training phase





# Neural Network Classifiers

- Value of a node may be linear combination of inputs, or may be a non-linear function
  - E.g. sigmoid function
- **Backpropagation algorithm** works as follows
  - Weights are set randomly initially
  - Training instances are processed one at a time
    - Output is computed using current weights
    - If classification is wrong, weights are tweaked to get a higher score for the correct class





# Neural Networks (Cont.)

- **Deep neural networks** have a large number of layers with large number of nodes in each layer
- **Deep learning** refers to training of deep neural network on very large numbers of training instances
- Each layer may be connected to previous layers in different ways
  - Convolutional networks used for image processing
  - More complex architectures used for text processing, and machine translation, speech recognition, etc.
- Neural networks are a large area in themselves
  - Further details beyond scope of this chapter



# Regression

- Regression deals with the prediction of a value, rather than a class.
  - Given values for a set of variables,  $X_1, X_2, \dots, X_n$ , we wish to predict the value of a variable  $Y$ .
- One way is to infer coefficients  $a_0, a_1, a_2, \dots, a_n$  such that
$$Y = a_0 + a_1 * X_1 + a_2 * X_2 + \dots + a_n * X_n$$
- Finding such a linear polynomial is called **linear regression**.
  - In general, the process of finding a curve that fits the data is also called **curve fitting**.
- The fit may only be approximate
  - because of noise in the data, or
  - because the relationship is not exactly a polynomial
- Regression aims to find coefficients that give the best possible fit.



# Association Rules

- Retail shops are often interested in associations between different items that people buy.
  - Someone who buys bread is quite likely also to buy milk
  - A person who bought the book *Database System Concepts* is quite likely also to buy the book *Operating System Concepts*.
- Associations information can be used in several ways.
  - E.g. when a customer buys a particular book, an online shop may suggest associated books.
- **Association rules:**

$\text{bread} \Rightarrow \text{milk}$        $\text{DB-Concepts, OS-Concepts} \Rightarrow \text{Networks}$

  - Left hand side: **antecedent**, right hand side: **consequent**
  - An association rule must have an associated **population**; the population consists of a set of **instances**
    - E.g. each transaction (sale) at a shop is an instance, and the set of all transactions is the population



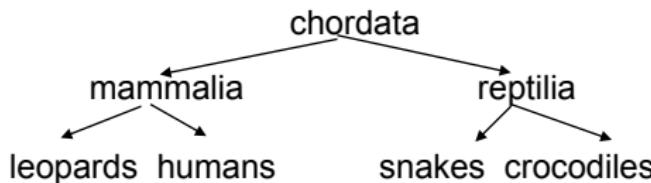
## Association Rules (Cont.)

- Rules have an associated support, as well as an associated confidence.
- **Support** is a measure of what fraction of the population satisfies both the antecedent and the consequent of the rule.
  - E.g. suppose only 0.001 percent of all purchases include milk and screwdrivers. The support for the rule  $milk \Rightarrow screwdrivers$  is low.
- **Confidence** is a measure of how often the consequent is true when the antecedent is true.
  - E.g. the rule  $bread \Rightarrow milk$  has a confidence of 80 percent if 80 percent of the purchases that include bread also include milk.
- We omit further details, such as how to efficiently infer association rules



# Clustering

- **Clustering:** Intuitively, finding clusters of points in the given data such that similar points lie in the same cluster
- Can be formalized using distance metrics in several ways
  - Group points into  $k$  sets (for a given  $k$ ) such that the average distance of points from the centroid of their assigned group is minimized
    - Centroid: point defined by taking average of coordinates in each dimension.
  - Another metric: minimize average distance between every pair of points in a cluster
- **Hierarchical clustering:** example from biological classification
  - (the word classification here does not mean a prediction mechanism)





# Clustering and Collaborative Filtering

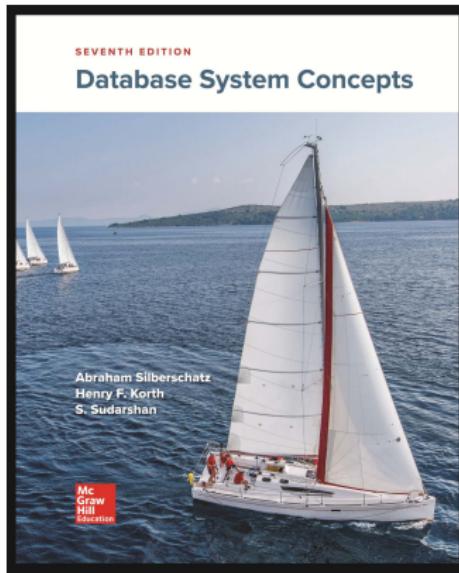
- Goal: predict what movies/books/... a person may be interested in, on the basis of
  - Past preferences of the person
  - Preferences of other people
- One approach based on repeated clustering
  - Cluster people based on their preferences for movies
  - Then cluster movies on the basis of being liked by the same clusters of people
  - Again cluster people based on their preferences for (the newly created clusters of) movies
  - Repeat above till equilibrium
  - Given new user
    - Find most similar cluster of existing users and
    - Predict movies in movie clusters popular with that user cluster
- Above problem is an instance of **collaborative filtering**



# Other Types of Mining

- **Text mining**: application of data mining to textual documents
- **Sentiment analysis**
  - E.g. learn to predict if a user review is positive or negative about a product
- **Information extraction**
  - Create structured information from unstructured textual description or semi-structured data such as tabular displays
- **Entity recognition and disambiguation**
  - E.g. given text with name “Michael Jordan” does the name refer to the famous basketball player or the famous ML expert
- **Knowledge graph** (see Section 8.4)
  - Can be constructed by information extraction from different sources, such as Wikipedia

# Database Administration: Data Analytics.



Content has been extracted from *Database System Concepts 7ed (Chapter 11)*, by Abraham Silberschatz, Henry Korth & S. Sudarshan, 2019. Visit <https://db-book.com/>.