Data Analytics: Data Warehousing, Data Mining

Dr. Shrutilipi Bhattacharjee, Assistant Professor, Dept. of IT, NIT Karnataka, India

Other Types of Classifiers

- Neural net classifiers are studied in artificial intelligence and are not covered here
- Bayesian classifiers use **Bayes theorem**, which says

$$p(c_j \mid d) = p(d \mid c_j)\, p(c_j)\, p(d)$$

where

$p(c_j \mid d)$ = probability of instance d being in class c_j,
$p(d \mid c_j)$ = 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 $p(d \mid c_j)$
  - 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

$$p(d \mid c_j) = p(d_1 \mid c_j) * p(d_2 \mid c_j) * \ldots * (p(d_n \mid c_j)$$

  - Each of the $p(d_i \mid c_j)$ 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

Regression

- Regression deals with the prediction of a value, rather than a class
  - Given values for a set of variables, $X_1, X_2, \ldots, X_n$, we wish to predict the value of a variable Y

- One way is to infer coefficients $a_0, a_1, a_2, \ldots, a_n$ such that

$$Y = a_0 + a_1 * X_1 + a_2 * X_2 + \ldots + a_n * X_n$$

- 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:**

  bread $\Rightarrow$ *milk* *DB-Concepts, OS-Concepts* $\Rightarrow$ 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

- 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 is *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

Finding Association Rules

- We are generally only interested in association rules with reasonably high support (e.g., support of X% or greater)

- Naïve algorithm
  - Consider all possible sets of relevant items
  - For each set find its support (i.e., count how many transactions purchase all items in the set)
    - **Large itemsets**: Sets with sufficiently high support

  - Use large itemsets to generate association rules
    - From itemset *A* generate the rule $A - \{b\} \Rightarrow b$ for each $b \in A$
      - Support of rule = support (*A*)
      - Confidence of rule = support (*A*) / support ($A - \{b\}$)

Finding Support

- Determine support of itemsets via a single pass on set of transactions
  - Large itemsets: Sets with a high count at the end of the pass

- If memory not enough to hold all counts for all itemsets use multiple passes, considering only some itemsets in each pass

- Optimization: Once an itemset is eliminated because its count (support) is too small none of its supersets needs to be considered

- The **a priori** technique to find large itemsets:
  - Pass 1:
    - Count support of all sets with just 1 item
    - Eliminate those items with low support
  - Pass *i*:
    - **Candidates**: every set of *i* items such that all its *i-1* item subsets are large
    - Count support of all candidates
    - Stop if there are no candidates

Other Types of Associations

- Basic association rules have several limitations

- Deviations from the expected probability are more interesting
  - E.g., if many people purchase bread, and many people purchase cereal, quite a few would be expected to purchase both
  - We are interested in **positive** as well as **negative correlations** between sets of items
    - Positive correlation: Co-occurrence is higher than predicted
    - Negative correlation: Co-occurrence is lower than predicted

- Sequence associations / correlations
  - E.g., whenever bonds go up, stock prices go down in 2 days

- Deviations from temporal patterns
  - E.g., deviation from a steady growth
  - E.g., sales of winter wear go down in summer
    - Not surprising, part of a known pattern
    - Look for deviation from value predicted using past patterns

Clustering

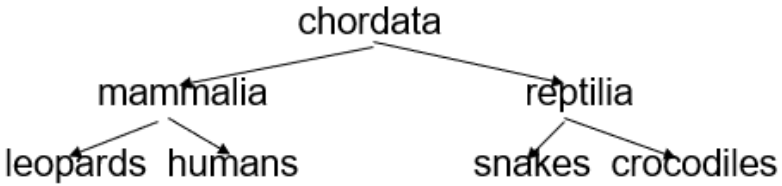
- Clustering: Intuitively, finding clusters of points in the given data such that similar points lie in the same cluster

- 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

- Has been studied extensively in statistics, but on small data sets
  - Data mining systems aim at clustering techniques that can handle very large data sets
  - E.g., the Birch clustering algorithm (more shortly)

Hierarchical Clustering

- Example from biological classification
  - (the word classification here does not mean a prediction mechanism)



- Other examples: Internet directory systems (e.g., Yahoo)

- **Agglomerative clustering algorithms**
  - Build small clusters, then cluster small clusters into bigger clusters, and so on

- **Divisive clustering algorithms**
  - Start with all items in a single cluster, repeatedly refine (break) clusters into smaller ones

Clustering Algorithms

- Clustering algorithms have been designed to handle very large datasets

- E.g., the **Birch algorithm**
  - Main idea: Use an in-memory R-tree to store points that are being clustered
  - Insert points one at a time into the R-tree, merging a new point with an existing cluster if is less than some δ distance away
  - If there are more leaf nodes than fit in memory, merge existing clusters that are close to each other
  - At the end of first pass we get a large number of clusters at the leaves of the R-tree
    - Merge clusters to reduce the number of clusters

Collaborative Filtering

- Goal: Predict what movies/books/… a person may be interested in, on the basis of
  - Past preferences of the person
  - Other people with similar past preferences
  - The preferences of such people for a new movie/book/…

- One approach based on repeated clustering
  - Cluster people on the basis of 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

- Above problem is an instance of **collaborative filtering**, where users collaborate in the task of filtering information to find information of interest

Other Types of Mining

- **Text mining**: Application of data mining to textual documents
  - Cluster web pages to find related pages
  - Cluster pages a user has visited to organize their visit history
  - Classify web pages automatically into a web directory

- **Data visualization** systems help users examine large volumes of data and detect patterns visually
  - Can visually encode large amounts of information on a single screen
  - Humans are very good a detecting visual patterns

Current Trends in Database System

- Top Emerging Trends in Database Management System
  - Cloud Database
  - AI in Database Management System
  - Graph Database

Next Lecture

Storage and File Structure

Thank you for your attention...

Any question?

Contact:
Department of Information Technology, NITK Surathkal, India
6th Floor, Room: 13
Phone: +91-9477678768
E-mail: shrutilipi@nitk.edu.in