Statistical Data Analysis

N. Sairam

sairam@cse.sastra.edu

Unit-V

Predictive Analytics

- A bit of an umbrella term (others might say: marketing term) for various tasks that share the intent of deriving predictive information directly from data
- Different specific application areas
  1. Classification or supervised learning
     - Assign each record to exactly one of a set of predefined classes
     - For example, classify credit card transactions as "valid" or "fraudulent"
     - Spam filtering is another example
     - Classification is considered "supervised", because the classes are known ahead of time and dont need to be inferred from the data
     - Algorithms are judged on their ability to assign records to the correct class

Application Areas

2. Clustering or unsupervised learning
   - Group records into clusters, where the size and shape-and often even the number-of clusters is unknown
   - Clustering is considered "unsupervised", because no information about the clusters is available ahead of the clustering procedure
3. Recommendation
   - Recommend a suitable item based on past interest or behavior
   - Recommendation can be seen as a form of clustering, where you start with an anchor and then try to find items that are similar or related to it
4. Time Series Forecasting

Some Classification Terminology

- A data set have multiple elements, records, or instances
- Each instance consists of several attributes or features
- One of the special features: class label
- Each record belongs to exactly one class

Some Classification Technology

- A large number of classification problems are binary, consisting only of two classes (valid or fraudulent, spam or not spam); however, multiclass scenarios do also occur
- Many classification algorithms can deal only with binary problems, but this is not a real limitation because any multiclass problem can be treated as a set of binary problems
- A classifier takes a record (i.e., a set of attribute values) and produces a class label for this record
- Building and using a classifier generally follows a three-step process of training, testing, and actual application

Some Classification Technology

- We first split the existing data set into a training set and a test set
- In the training phase, we present each record from the training set to the classification algorithm
- Next we compare the class label produced by the algorithm to the true class label of the record in question; then we adjust the algorithms "parameters" to achieve the greatest possible accuracy or, equivalently, the lowest possible error rate
- The results can be summarized in a so-called confusion matrix whose entries are the number of records in each category

Confusion Matrix

	Predicted: A	**Predicted: B**
Actual: A	Correct	Incorrect
Actual: B	Incorrect	Correct

Some Classification Technology

- Training Error: The error rate derived from the training set
- Generalization Error: The error rate obtained when a classifier operate on the elements of the test set to see how well it classifies them and is a much more reliable indicator of the accuracy of the classifier
- Underfitting:If it cannot represent the desired behavior very well, and both its training and generalization error will be poor
- Overfitting:If we make the classifier too complex, then it will perform very well on the training set (low training error) but will not generalize well to unknown data points (high generalization error); this is known as overfitting
- Once a classifier has been developed and tested, it can be used to classify truly new and unknown data pointsthat is, data points for which the correct class label is not known

Algorithms for Classification

- K Nearest Neighbour Classification
  - Determine the parameter K=Number of nearest neighbours
  - Calculate the distance between the query instance and all the training samples
  - Sort the distance and determine nearest neighbours based on the K-th Minimum distance
  - Gather the category Y of the nearest neighbours
  - Use simple majority of the category of nearest neighbours as the prediction value of the query instance

Example

- We have data from the questionnaires survey (to ask people opinion) and objective testing with two attributes (acid durability and strength) to classify whether a special paper tissue is good or not. Here is four training samples. Now the factory produces a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7. Without another expensive survey, can we guess what the classification of this new tissue is using KNN?

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Classification
7	7	Bad
7	4	Bad
3	4	Good
1	4	Good

Solution

- Let K=3
- Calculate the square of the distance between query instance and the given instances

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	**Square Distance to query instance (3, 7)**
7	7	$(7-3)^2+(7-7)^2=16$
7	4	$(7-3)^2+(4-7)^2=25$
3	4	$(3-3)^2+(4-7)^2=9$
1	4	$(1-3)^2+(4-7)^2=13$

- Sort the distances

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Square Distance to query instance (3, 7)	**Rank minimum distance**	**Is it included in 3-Nearest neighbors?**
7	7	$(7-3)^2+(7-7)^2=16$	**3**	**Yes**
7	4	$(7-3)^2+(4-7)^2=25$	**4**	**No**
3	4	$(3-3)^2+(4-7)^2=9$	**1**	**Yes**
1	4	$(1-3)^2+(4-7)^2=13$	**2**	**Yes**

Solution

- Gather the category Y of the nearest neighbours

X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Square Distance to query instance (3, 7)	Rank minimum distance	Is it included in 3-Nearest neighbors?	**Y = Category of nearest Neighbor**
7	7	$(7-3)^2+(7-7)^2=16$	3	Yes	**Bad**
7	4	$(7-3)^2+(4-7)^2=25$	4	No	**-**
3	4	$(3-3)^2+(4-7)^2=9$	1	Yes	**Good**
1	4	$(1-3)^2+(4-7)^2=13$	2	Yes	**Good**

- Using the simple majority policy, the given query instance is "good"

R Code

```
x<-c(7,7,3,1)
y<-c(7,4,4,4)
train<-data.frame(x,y)
x1<-c(3)
y1<-c(7)
test<-data.frame(x1,y1)
cl <- factor(c(rep("Bad",2), rep("Good",2)))
knn(train, test, cl, k = 1, l = 0, prob = FALSE,
 use.all = TRUE)
[1] Good
Levels: Bad Good
```

Naive Bayesian Classifier

- The Bayes Naive classifier selects the most likely classification V_{nb} given the attribute values a_1;a_2;...a_n. This results in:

$$V_{nb} = argmax_{v_j \in V} P(V_j) \prod P(a_i|v_j) \quad (1)$$

- We generally estimate P(a_i | v_j) using m-estimates:

$$P(a_i|v_j) = \frac{n_c + mp}{n + m} \quad (2)$$

- where n= number of training samples for which v=v_j
- n_c=number of examples for which v=v_j and a=a_i
- p=a priori estimate for P(a_i | v_j)
- m= the equivalent sample size

Example

- Car Theft Example
  - Attributes are Color , Type , Origin, and the subject, stolen can be either yes or no
- Data Set:

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

The Process

- Some standard methods commonly used to enhance accuracy-especially for the important case when the most "interesting" type of class occurs much less frequently than the other types
- Ensemble Methods: Bagging and Boosting
  - The term ensemble methods refers to a set of techniques for improving accuracy by combining the results of individual or "base" classifiers
  - When performing some experiment or measurement multiple times and then averaging the results: as long as the experimental runs are independent, we can expect that errors will cancel and that the average will be more accurate than any individual trial

Ensemble Methods

- The same logic applies to classification techniques: as long as the individual base classifiers are independent, combining their results will lead to cancellation of errors and the end result will have greater accuracy than the individual contributions

Bagging

- Bagging is an application of the bootstrap idea to classification
- We generate additional training sets by sampling with replacement from the original training set
- Each of these training sets is then used to train a separate classifier instance
- During production, we let each of these instances provide a separate assessment for each item we want to classify
- The final class label is then assigned based on a majority vote or similar technique

Boosting

- Boosting is another technique to generate additional training sets using a bootstrap approach
- In contrast to bagging, boosting is an iterative process that assigns higher weights to instances misclassified in previous rounds
- As the iteration progresses, higher emphasis is placed on training instances that have proven hard to classify correctly
- The final result consists of the aggregate result of all base classifiers generated during the iteration
- A popular variant of this technique is known as "AdaBoost."

Business Intelligence

- Data Warehouses
  - Comprehensive data stores in which data is represented in a denormalized schema that is intended to be more general than the schema of the transactional databases and also easier to use for non technical users
  - Data is imported into the data warehouse from the transactional databases using so-called ETL (extraction, transformation, and load) processes
- Business Intelligence
  - It is an accessibility layer sitting on top of a data warehouse or similar data store, trying to make the underlying data more useful through better reporting, improved support for ad hoc data analysis, and even some attempts at canned predictive analytics

Reporting

- The primary means by which data is used for "analysis" purposes in an enterprise environment is via reports
- Much of "business intelligence revolves around reporting, and "reporting" is usually a big part of what companies do with their data

Types of Reports

1. Representative reports
   - Intended for external users
   - Quarterly filings certainly fall into this category, as do reports the company may provide to its customers on various metrics
   - In short, anything that gets published
2. Operational Reports
   - Used by managers within the company to actually run the business
   - Such reports include information on the the number of orders shipped today, the size of the backlog, or the CPU loads of various servers

Corporate Metrics and Dashboards

- Metrics Programs
  - The goals of a metrics program are to define those quantities that are most relevant and should be tracked and to design and develop the infrastructure required to collect the appropriate data and make it accessible
- Dash Board
  - A dashboard might be the visible outcome of a metrics program
  - The purpose of a dashboard is to provide a high-level view of all relevant metrics in a single report (rather than a collection of individual, more detailed reports)
  - Dashboards often include information on whether any given metric is within its desired range
  - Dashboard implementations can be arbitrarily fancy, with various forms of graphical displays for individual quantities

Data Quality Issues

- All reporting and metrics efforts depend on the availability and quality of the underlying data
- If the required data is improperly captured (or not captured at all), there is nothing to work with
- Two problems in particular occur frequently when one is trying to prepare reports or metrics: data may not be available or it may not be consistent
- Data Availability
  - If data is not available, this does not necessarily mean that it is not being collected
  - Data may be collected but not at the required level of granularity. Or it is collected but immediately aggregated in a way that loses the details required for later analysis

Data Quality Issues

- Data Consistency
  - Problems of data consistency (as opposed to data availability) occur in every company of sufficient size, and they are simply an expression of the complexity of the underlying business